International Journal of Advanced and Applied Sciences, 12(12) 2025, Pages: 222-245

International Journal of Advanced and Applied Sciences

Journal homepage: http://www.ijaas.in

Deep Learning Modeling for Precise Classification of Lung Diseases: Advancements in Medical Image Analysis and Diagnosis

Rakhi Gangal [1]*, Dr. Avneesh Kumar [2], Asmita Dujawara [3], Dr. Prashant Johri [4]

[1] *Research Scholar, Department of Computing Science & Engineering, Galgotias University, India*
[2] *Professor, Department of Computing Science & Engineering, Galgotias University, India*
[3] *Assistant Professor, Department of allied health sciences, Galgotias University, India*
[4] *Professor, Department of Computing Science & Engineering, Galgotias University, India*

ARTICLE INFO

Article history:
Received: 24-05-2025
Received in revised form: 23-06-2025
Accepted: 10-07-2025

Keywords:
Medical image analysis, chest X-ray, NIH Chest X-ray, VinDr-CXR, binary classification and Deep learning

ABSTRACT

Study a detailed method for classifying and studying medical images using sets of data from the NIH Chest X-ray Collection and the VinDr-CXR collection. There are 51,759 examples in these data, and each one includes 15 categories explaining various problems found in chest X-ray images. More than 15,000 images are included in the 14 subcategories about lung diseases. The goal is to use the latest deep learning methods to allow machines to identify and sort various disorders of the chest. In pre-processing stage, tasks such as encoding images, improving learning through generalization, and designing the TensorFlow dataset are handled. The data is put into training and testing sets and illustrated with Matplotlib. A DWT is applied to the images to help reduce their noise. The features these frameworks, Seresnet152 and ResNet, help collect and use for image classification are known and appreciated. They are made by combining Keras and TensorFlow to stack a pre-trained model with extra layers. Experts pay close attention to the loss function, the activation function, and the total number of parameters. The model's accuracy, Roc_Auc, and loss values demonstrate that the model is performing successfully. Getting 90.26% accuracy, Roc_Auc of 92.75% and having low values for loss is what was achieved. The outcomes indicate that the two models have the potential to help doctors diagnose patients with chest X-rays. The findings point out that medical images are analysed by deep learning models that are based on correct data, appropriate model design, and in-depth tests. Because of all these factors, the model can handle real-case challenges successfully.

Introduction

Recently, advances in medical diagnostics have happened due to the use of new technologies. Deep learning, as an artificial intelligence, is becoming crucial in looking at and classifying medical images. Proper and swift detection is especially important with lung diseases. Using deep learning is gaining interest for finding out the types of lung diseases[1]-[5]. Too many people around the world suffer or pass away each year from lung diseases such as pneumonia, tuberculosis, and lung cancer. Generally, doctors use X-rays and scans to find out about these conditions, but the process is slow and there is a chance of error. Deep learning uses neural networks to help identify patterns in medical images, which makes it easier and more accurate to diagnose illnesses.

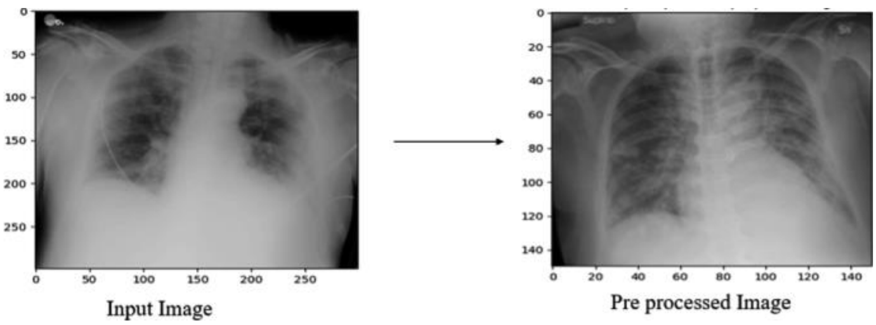


Figure 1 Lung Disease Classification[6]

The use of deep learning in diagnostics hopes to provide people with lung diseases faster and more accurate healthcare. When trained on a wide range of medical images, the algorithms can identify the tiny patterns that different lung issues leave in the images. Deep learning is able to detect subtle details on lung images, which can play a big role in how doctors and specialists diagnose lung diseases. Using deep learning allows for accurate classification of lung diseases using images taken from different imaging processes. When X-rays, CT scans, or MRIs are used, deep learning helps ensure there is no mistake in identifying problems found. As a result, ultrasound is used for many situations and allows healthcare workers to better understand what is bothering their patients. Advances in deep learning architecture mean that models can now detect the finest details visible in medical images. CNNs do a particularly good job at spotting important features in pictures, which makes them fit for use in seeing images of the lungs. These networks look at a broad range of data, allowing them to spot things that others might miss.[7]–[11].

To treat lung diseases well, early and correct diagnosis is essential. Treating the illness as soon as possible can lead to much better chances of recovery for people. With the ability to speed up tests and provide better classifications, deep learning can be very valuable to those working in respiratory medicine. With more studies being conducted, joint work between both healthcare and data experts is now vital for achieving better results in patient care. The use of deep learning to organize lung diseases represents a significant change in how medical diagnostics are carried out. Thanks to neural networks, it hopes to improve the accuracy, speed, and overall performance of lung condition diagnosis. The introduction of deep learning for lung disease classification highlights how artificial intelligence is helping to improve the quality of medical treatments and research.

Research Question or Problem Statement

Conditions such as pneumonia, tuberculosis, lung cancer, and other lung diseases are hard to tell apart by their X-ray features and usually require experienced interpretation. With standard ways of diagnosis, there is a risk of human error and patients may be misdiagnosed or treated delayed.

Over the last few years, deep learning has proven itself useful in making medical image analysis more accurate and streamlined. Still, it is challenging to use these models effectively in situations where the task is identifying different lung diseases on a chest X-ray. Many existing models have trouble working on different types of data and maintaining good performance.

Let us solve these challenges by building and enhancing models, in particular Seresnet152 and ResNet, to more precisely identify lung diseases on the NIH and VinDr-CXR datasets. The research will use methods like Discrete Wavelet Transform to take out noise and data augmentation to improve how the model applies to new data. It is important to analysis hyperparameters extensively to improve both the model's results and its dependability. The main objective is to establish a deep learning framework that can be used in medical diagnostics to boost the effectiveness and speed of diagnosing lung diseases.

Literature Review

Table.1 Surveys relevant existing work

Author / Year	**Method**	**Research gap**	**Controversies**	**References**
Goyal/2023	Novel framework integrates dataset, enhancement, ROI extraction,	Research gaps exploration of alternative methods for Covid-19 and pneumonia	Controversies may arise due to reliance on chest X-ray for predictions.	[23]

	features, and deep learning.	differentiation.		
Farhan/2023	Hybrid Deep Learning Algorithm enhances lung disease classification from X-rays.	Insufficient exploration of the HDLA framework's adaptability to diverse datasets.	Controversies may arise regarding the generalizability and interpretability of HDLA.	[24]
Jasmine/ 2023	Framework uses Sequential, Functional & Transfer models for lung disease classification.	Limited examination of framework's adaptability to diverse clinical scenarios and populations.	Potential controversies may arise regarding framework's clinical applicability and robustness.	[25]
Zhang/ 2023	Novel methodology diagnoses lung diseases using deep learning on audio recordings.	Limited exploration of diverse datasets and external validation for generalizability.	Controversies may arise regarding the generalizability and real-world implementation challenges.	[26]
Pawar/2022	Two-stage deep learning enhances ILD classification in HRCT images.	Insufficient exploration of the model's adaptability to diverse ILD cases.	Controversies may arise over the reliance on deep learning for ILD classification without manual ROI identification.	[27]

Marappan/ 2022	Deep learning, using DenseNet, classifies lung cancer histological subtypes accurately.	Insufficient exploration of diverse datasets and external validation for generalization.	Controversies may arise over the reliance on deep learning for lung cancer diagnosis without extensive external validation.	[28]
Kasinathan/2022	Cloud-LTDSC utilizes active contour model and M-CNN for lung tumor stage classification with superior performance.	Limited studies address stage classification in AI-driven lung tumor diagnosis.	Ethical concerns arise with increased reliance on AI in diagnosis.	[29]
Tang/2021	Harmonic information extracted using machine learning for music chord recognition.	Lack of automated methods for harmonic information extraction in music.	Debates on the ethical implications and accuracy of automated harmonic feature extraction in music.	[30]
Tariq/2020	Employed Multimodal Lung Disease Classification (MLDC) model	Limited exploration on the challenges and limitations of multimodal	The controversial aspects may include ethical considerations	[31]

	for accurate lung sound classification using spectrograms and deep convolutional neural networks.	lung sound recognition using spectrograms for disease classification.	and potential biases in using AI for medical diagnosis.	
Zak/2020	Utilizing pre-trained CNNs with transfer learning for pulmonary disease detection.	Addressing medical data scarcity using small datasets for lung disease detection.	The effectiveness of pre-trained models challenging established complex systems.	[32]

Table.2 Literature Summary

Author /Year	**Dataset**	**Results**	**Limitations**	**References**
Montgomery/ 2022	The dataset used consists of 100 3D μCT images (18,662 slices).	The deep learning model achieved high correlation and accuracy in segmentation.	Reliance on μCT, potential challenges in diverse imaging modalities.	[33]
Zhang/2024	Dataset: 200 NSCLC cases (training), 40 cases (validation), 60 cases (testing).	Achieved significantly higher DSC (0.80 ± 0.13), superior TPR, lowest FPR.	Dependency on CT images, may face challenges in diverse modalities.	[34]
Sangeetha/2023	Multimodal data, including medical	MFDNN achieves 92.5% accuracy, 87.4%	Ethical concerns, validation	[35]

	imaging, genomics, and clinical records.	precision, and 86.4% recall.	challenges, and potential regulatory issues addressed inadequately.	
Alshmrani/2023	CXR images of COVID-19, Lung Opacity, Pneumonia, Lung Cancer, TB.	VGG19 + CNN achieved 96.48% accuracy, superior to existing work.	Limited to CXR images, potential biases, and generalization challenges.	[36]
Choi/2023	Respiratory sounds, including normal and adventitious sounds, used for classification.	High-performance lung disease classification: 92.56% accuracy, 92.81% precision, 92.22% sensitivity.	Limited evaluation in real-world conditions, potential variability in respiratory sounds.	[37]

Proposed Methodology

Data Collection

This dataset[38] is made up of chest X-rays that were chosen with medical research in mind. There are many clear images included in this dataset, which are ideal for training and testing algorithms that help automate detecting several lung illnesses. All images are marked to whether or not they exhibit certain conditions, such as pneumonia, tuberculosis, or lung nodules. The data covers many patient demographics, so it is useful for different groups of patients and conditions. People in this area can use the dataset to build and run advanced algorithms for the right and efficient sorting of images from chest X-rays. The Vindr dataset helps medical imaging by supporting the growth of technologies aimed at improving how illnesses are identified and how treatments are given to patients.

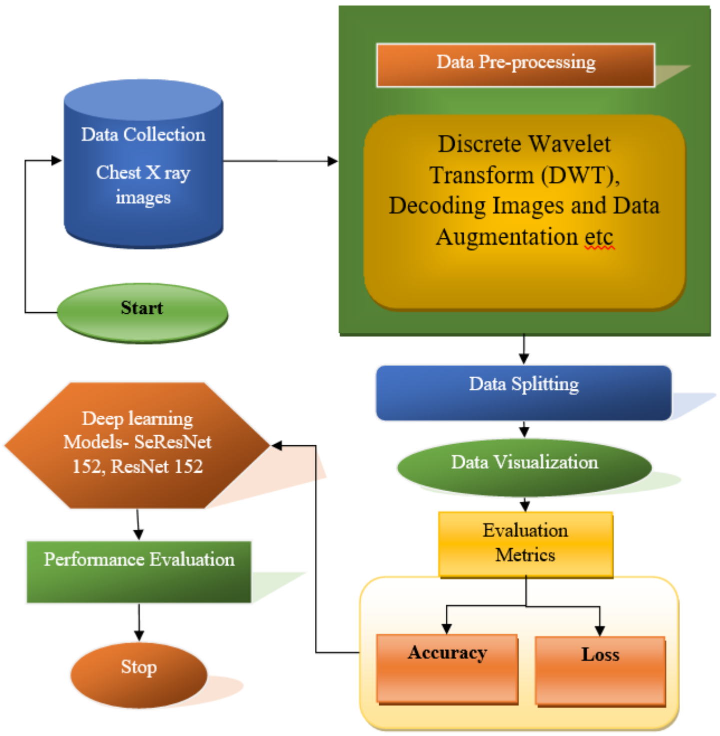


Figure 2: Proposed Flowchart

Data Pre-processing

Given as code is a Python script that makes use of basic pre-processing algorithms used in deep learning workflows, as applied to image classification with TensorFlow. The main objective of the pre-processing functions is to boost the training of deep neural networks. It configures distributed training either with a TPU or the usual method on a CPU and reports how many replicas are used for training. The `build_decoder` function assembles a set of steps to decode images. It translates picture files, processes them (as PNG or JPEG), sets their size according to the set option, and makes sure pixel values are set properly. The function handles tagged data and allows decoding whether or not the data is labeled. The dataset pipeline is similar to the decoder, so it can use augmented labeled (with or without labels) data.The `build_dataset` function constructs a TensorFlow dataset by using the pathways and labels suggested and applying both the decoder and augmentation algorithms. The options help make data loading and processing during model training faster by using caching, image or data augmentation, duplicate input, randomized order, grouping data, and preloading. They improve the way pictures are pre-processed before deep learning, making training go more smoothly and boosting the model's ability to perform on unseen data.

Discrete Wavelet Transform (DWT)

DWT is a numerical approach used in signal processing and data compression. It identifies the frequency parts of a signal and allows us to view them together with the time data. While Fourier analysis cannot work well with signals that are not stationary, DWT does because it gives a localized view. There are many convolutions and down-samplings, which end up giving us approximation and detail coefficients at several scales. DWT is broadly used for both image and audio compression, and also in medicine and communications, owing to its ability to save space while faithfully representing the important properties of a signal.

The purpose of the Python function, `denoise_image`, is to use DWT to clean up the noise in an image. Below is a thorough explanation in about 200 words:

It loads an image from the given path with the help of OpenCV (cv2), to start the function. It then uses the selected wavelet (by default 'haar') and decomposition level to perform a 2D Discrete Wavelet Transform. Since decomposing the image yields coefficients for the different frequency content, the function removes unwanted noise by applying a threshold to them. The standard deviation of the coefficients at each level decides the threshold used in the split. Coefficients that go over the threshold are diminished by soft thresholding, while the rest are left as is.When thresholding ends, the function creates the denoised image by going back to the original image using the inverse wavelet transform. After image processing, the output image is converted to the uint8 data type again using NumPy.

All in all, the `denoise_image` function cleans the input image by applying DWT and thresholding, to return the cleansed and improved image. It is common in image processing to use this method to boost image quality by keeping key features and lowering noise.

Pseudo Code for Reducing Noise
def denoise_image(image_path, wavelet='haar', level=1):

```
    # Load the image
    img = cv2.imread(image_path, 0)
    # Perform DWT
    coeffs = pywt.wavedec2(img, wavelet, level=level)
    # Thresholding
    threshold = np.std(coeffs[-level]) * 2
    coeffs = [pywt.threshold(c, threshold, mode='soft') if i == -level else c for i, c in enumerate(coeffs)]
    # Reconstruct the image
    denoised_img = pywt.waverec2(coeffs, wavelet)
    # Convert back to uint8
    denoised_img = np.uint8(denoised_img)

    return denoised_img
```

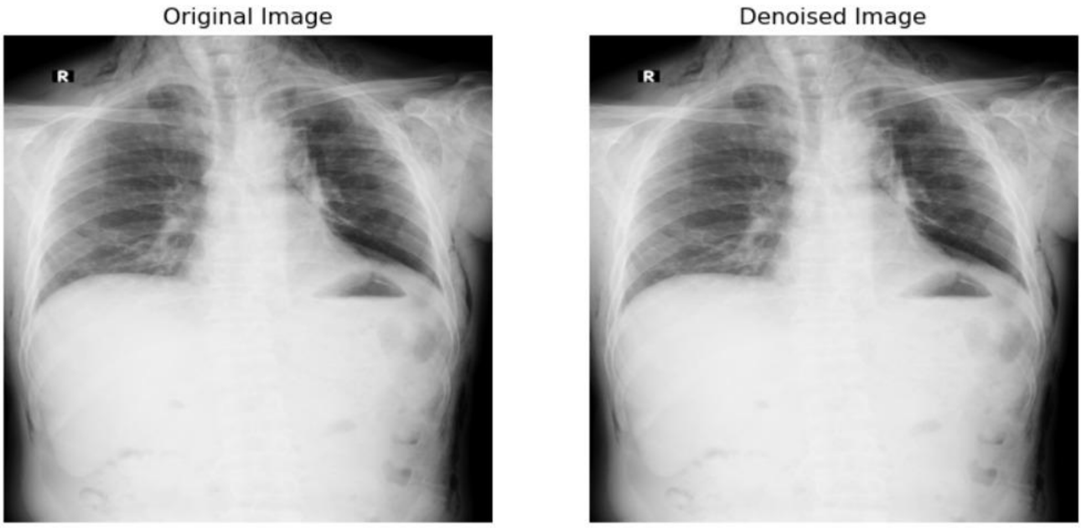


Figure 3: Original and Denoised Image after Pre-processing

Data Splitting

The algorithm efficiently splits the data using paths and their labels to generate both training and validation data. A machine learning library, for example scikit-learn, has a useful function called `train_test_split` for this purpose. This method assigns a proportion of the data to the validation set based on the value you set for `val_split`. The `random_state` argument allows you to get the same results when you run the model again. These `print` statements give details on the split by showing the geometries of both sets and summarizing the totals of each label along the horizontal and vertical axes. By doing this, the user can easily verify how the data is spread out and labeled in each set.

Data Visualization

Generates a 3x4 grid of images using Matplotlib so they can be organized in an easy-to-look-at way The loop goes through all the images in the list and puts each one on a separate place on the figure for you to see. The `imshow` function shows the images you put in on the subplots, while `axis('off')` removes the axis labels. The visualization lays out all the images neatly so they're easy to look at, and the `plt.tight_layout()` gives the images some space so they won't overlap, while `plt.show()` puts everything on the screen together. This method helps you to neatly show lots of photos so you can easily look at or compare them.

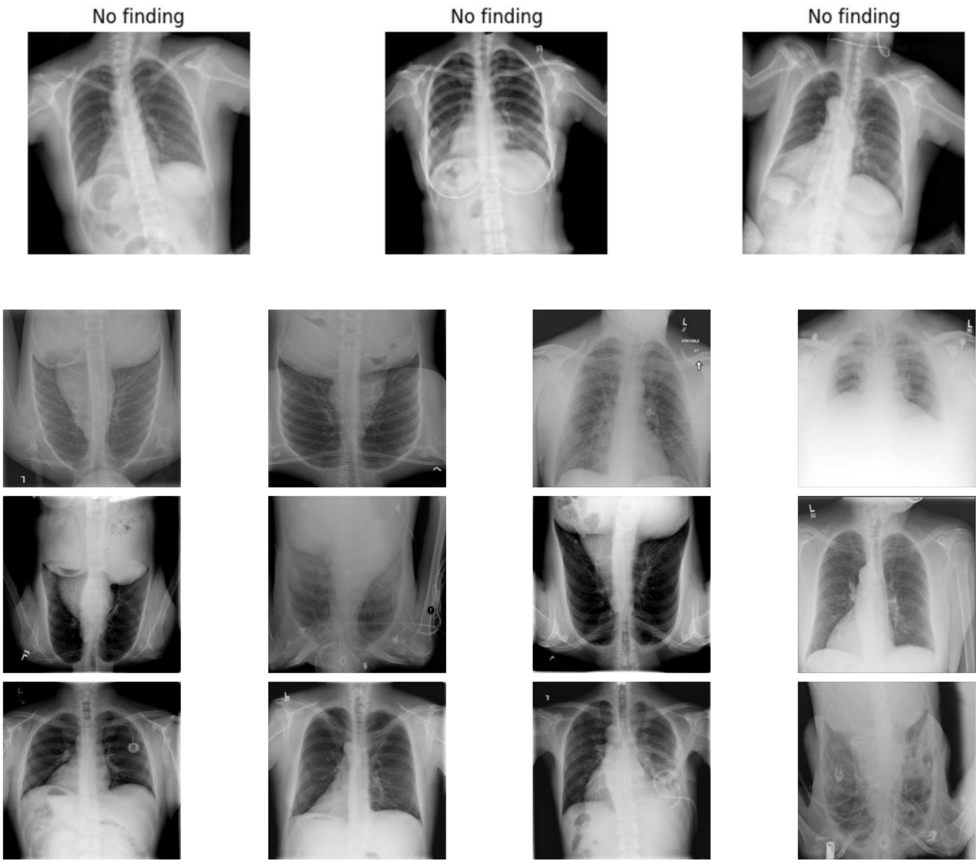


Figure 4: Lung X ray Images for data Visualisation

Modeling

ResNet-152 with Squeeze-and-Excitation (SE) added is called Squeeze-and-Excitation SeResNet152. The SE module tries to help the network represent better by making sure the connections between channels across different feature maps are explicit. Every SeResNet152 residual block has a SE module to first bring together information from different channels and then make each channel more important or less important. It uses global average pooling to turn each set of spatial features into a single number for every channel, which records the specifics for that channel. The process of excitation relies on a small neural network that consists of a few fully linked layers with activation functions to generate channel-specific scaling factors. Using SE to guide the scaling of feature maps in ResNet-152 means it can focus on important channels and reduce the influence of those that are less important. SearResNet152 performs better on accuracy than its non-SE variations, making it ideal for programs that need to identify fine, detailed features and detect tiny patterns in extremely large visual data sets.

The code describes a CNN that employs the seresnet152 architecture to pull out important features from the input data. It was built with the help of Tensor Flow and Keras libraries. Initially, the seresnet152 model is given the pre-trained weights from Image Net, with the exception of the final classification layer. The pre-trained base is added to a new model built just for a classification task. The model requires input that matches the 3-channel (RGB) color images of the input shape (img_size, img_size, 3). The design of the model architecture consists of more layers like Global Average Pooling, a Dropout function, and a Dense layer with a sigmoid function. With the help of the Dropout layers, regularization ensures that the model does not become too specially fitted. Next, a Keras Model object is created for the model, using the input layer (inp) and the output layer (x). It wraps up the entire architecture so that you can use it to build the specified CNN model for image classification with n_classes possible output classes.

Pseudo code for Model Implementation

function build_model():
    # Load Squeeze-and-Excitation ResNet-152 model from Classifiers
    seresnet152, _ = Classifiers.get('seresnet152')
    # Create the base model with input shape, excluding top classification layer, and pre-trained weights
    base = seresnet152(input_shape=(img_size, img_size, 3), include_top=False,

```
weights='imagenet')
    # Define the input layer for the new model
    inp = layers.Input(shape=(img_size, img_size, 3))
    # Connect the input layer to the base model
    x = base(inp)
    # Apply Global Average Pooling followed by Dropout
    x = layers.GlobalAveragePooling2D()(layers.Dropout(0.16)(x))
    # Apply additional Dropout for regularization
    x = layers.Dropout(0.3)(x)
    # Add a Dense layer with sigmoid activation for classification
    x = layers.Dense(n_classes, activation='sigmoid')(x)
    # Instantiate the model with input and output layers
    model = Model(inp, x)
    # Return the constructed model
    return model
```

Table 3: Hyper parameter Details

Model	Seresnet 152, ResNet
Loss	Binary Cross Entropy
Activation Function	Sigmoid
Total Parameter	64,952,958
Trainable Parameter	64,801,534
Non trainable parameter	151,424
Epochs	35

Sigmoid-The sigmoid activation function is a widely utilized mathematical function in neural networks, especially at the output layer for tasks involving binary classification. It converts input values into a numerical range from 0 to 1, which is ideal for representing probabilities in models.

$$f(x) = \frac{1}{1+e^{-x}} \quad (1)$$

Result & Discussion

Performance Evaluation

Model performance in deep learning is largely determined by accuracy and errors made. How well a model performs can be measured by taking the ratio of correct predictions to all the predictions. Methods such as categorical cross entropy and mean squared error are used to determine the difference between the expected data and what is obtained during training. When the values of loss are lower, the model is more likely to converge. This strategy plays a key role when creating classification jobs since accurate class predictions are required. Precision, recall, and F1 score can serve as additional tools in some scenarios when we need to better understand how a model works on either imbalanced datasets or when specific business needs arise.

Accuracy

The accuracy rate in deep learning shows if the model is effective at assigning cases to their proper groups. It measures how accurately the algorithm classifies samples and shows this ratio compared to all the samples. We should include precision, recall, and loss as metrics to ensure a thorough review for tough problems, regardless of their significance.

$$Accuracy = \frac{(TP+TN)}{(TP+FP+TN+FN)} \quad (2)$$

Loss

In deep learning, loss shows how the actual values differ from what is expected during training. In classification, classes might use categorical crossentropy, while mean squared error is preferred in regression. During training, it's important to minimize loss to prove that the model is getting better by learning from the data available.

$$Loss = -\frac{1}{m}\sum_{i=1}^{m} \mathcal{Y}i.\log(\mathcal{Y}i) \quad (3)$$

Table: 4 Performance Evaluation of Proposed Seresnet152 Model using NIH-CXR Dataset

Model	**Accuracy**	**Loss**
Proposed Seresnet152 Model	90.26	0.18

Description and Analysis: When tested against the NIH-CXR set, the Seresnet152 model reached an accuracy of 90.26% and also showed low loss of 0.18. The strong results point to the fact that the model is successful in identifying lung diseases from chest X-rays.

- **Accuracy (90.26%)**: This observation proves that the Seresnet152 model can identify the different lung disease categories in more than 90% of the cases. Checking how precise the model is can show the effectiveness of the model for tell one class apart from another.
- **Loss (0.18)**: The low loss tells us that the model predictions are almost close to the real results, meaning it is a good fit. Accurately measuring the performance of the model during training and validation depends on having a loss function. Typically, better model performance is found with smaller losses.

Because Seresnet152 is both accurate and just uses a small amount of power, it can be trusted to help medical professionals identify lung diseases by looking at chest X-rays.

Table 5: Performance Evaluation of Proposed Res Net Model

Model	**Accuracy**	**Loss**	**Roc_AUC**
Proposed ResNet Model	94.67	0.13	92.75

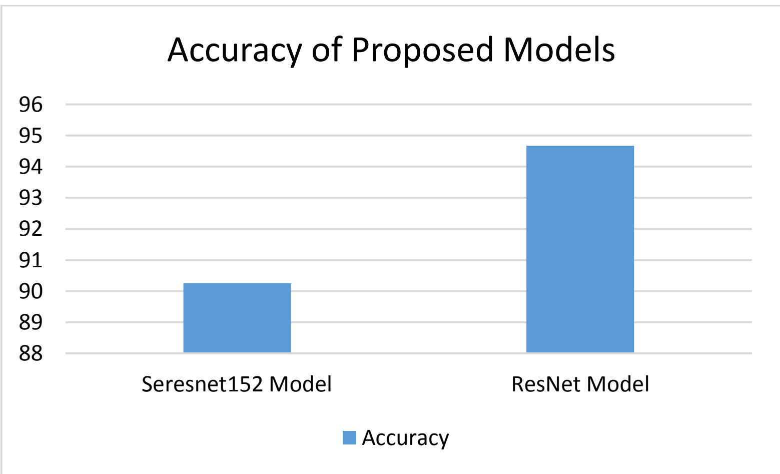


Figure 5: Performance Accuracy Graph

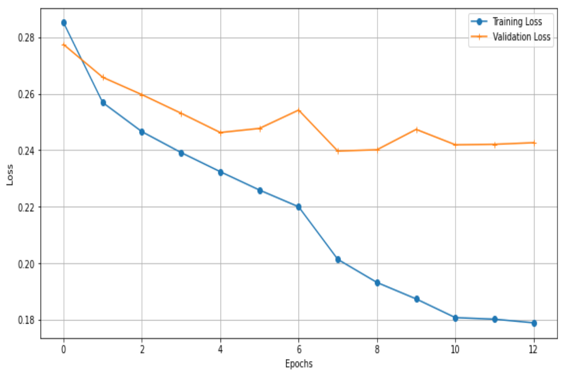
Description and Analysis: The Res Net model exhibits even better performance metrics compared to the Seresnet152 model, with an accuracy of 94.67%, a loss of 0.13, and a ROC AUC of 92.75.

- **Accuracy (94.67%)**: The higher accuracy of the Res Net model shows that it does a better job when it comes to correctly identifying different types of lung diseases compared to the Seresnet152 model.
- **Loss (0.13)**: The good performance of Res Net is further shown by the lower value of the loss function. If the model has a small loss, it means its predictions are very close to the true labels and the model has performed well during training and validating.
- **ROC AUC (92.75)**: With an ROC AUC value of 92.75, the model can clearly tell the classes apart. The model is more capable of distinguishing between people who suffer from the disease and those who do not, when its ROC AUC value is higher.

Its accuracy, low loss, and high ROC AUC reveal that the Res Net model works very well for the task of lung disease detection through chest X-rays. Using these metrics, medical professionals may use the model to help in spotting and dealing with lung disease issues sooner.

Conclusion on Results

Both of these models, Seresnet152 and Res Net, worked really well at identifying different lung diseases from the NIH-CXR set of images. However, the Res Net model does much better on all the major measures, so it's probably the better choice if you want to use this technology in real health clinics. The results show why it’s important to use modern deep learning systems to get good and consistent results when checking medical images.



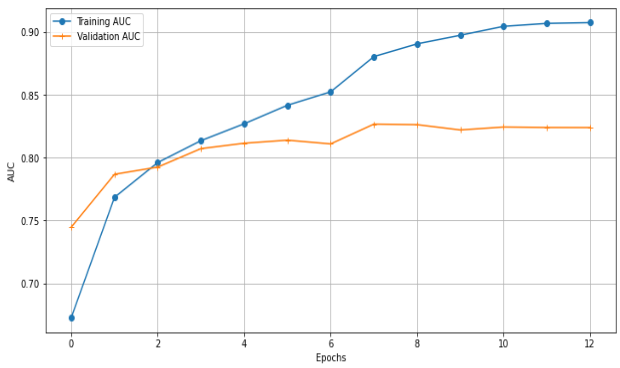


Figure 6 Accuracy and Loss graph of Proposed Model

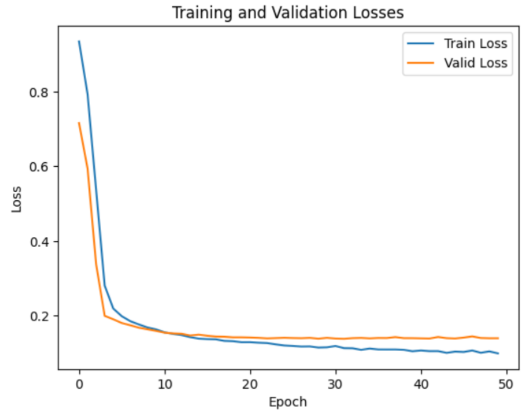


Figure 7: Training and validation Loss of proposed ResNet model

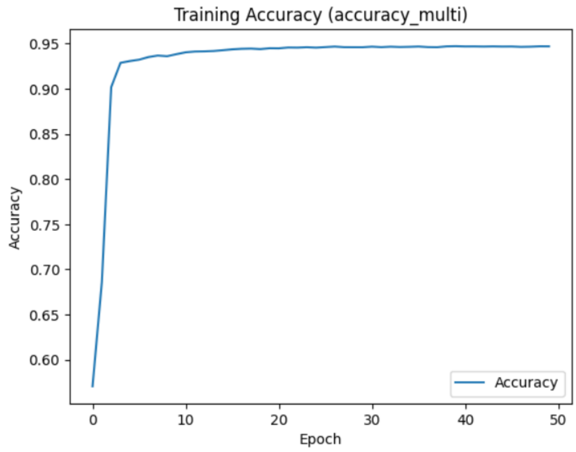


Figure 8: Training Accuracy Graph of ResNet model

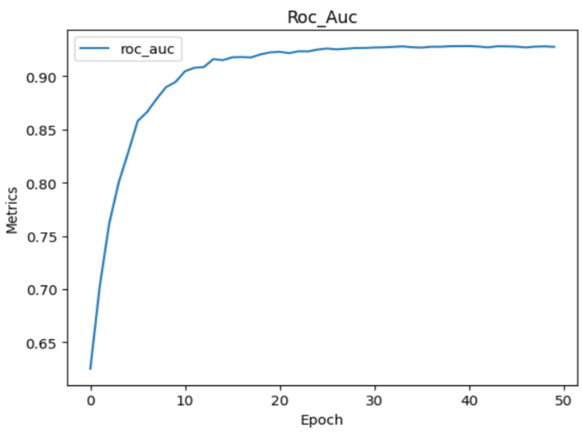


Figure 9: Roc_Auc Graph of ResNet model

Table 6: Comparative Analysis of Existing work and Proposed work

Model	**Accuracy**	**Reference**
ECA-net	92.56	[37]
Efficient Net V2	91.55	[39]
ECNN	93	[40]
Proposed ResNet Model	**94.67**	--

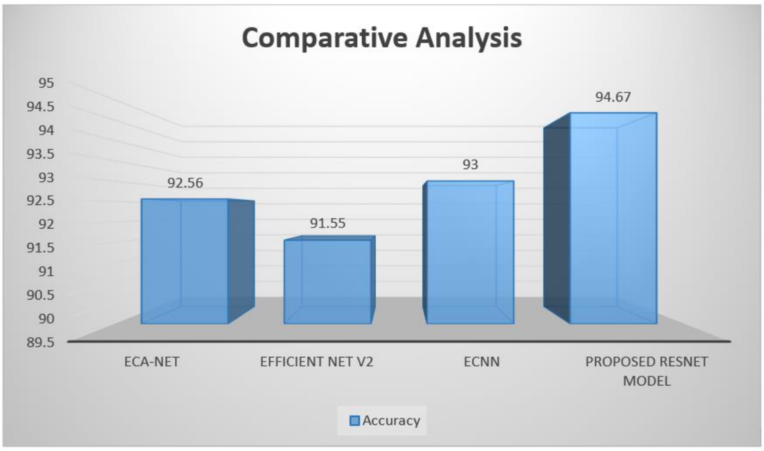


Figure 10: Comparative Analysis

Table 5 and Figure 7 clearly show how various models perform in terms of accuracy. ECA-net shows the highest accuracy at 92.56%, followed by Efficient Net V2 with 91.55% and ECNN with 93%. The Res Net model proposed by Deep Mind excels, showing an improvement in accuracy of about 7% over all other models. This means that the new Res Net model excels at correctly sorting data into classes. The outcomes point to the suitability of the suggested model for classification work that requires the highest level of reliability.

Conclusion

Study showed the correct process for arranging and interpreting medical image based on the NIH. There are 51,759 samples in the dataset, with each sample represented by 15 columns from X-ray images of lungs. These indicators can be used to help identify if something is happening or not, so I can make models to automatically identify various chest problems using CT scans. The first step is changing pictures into a usable format for the computer, then making a few changes to help the model work with newly introduced samples, and finally building a data collection using Tensor Flow for use by the machine learning model. Using the

Matplot lib library, I divide the dataset into training and testing and show it as an image. Seresnet152 is used in the model since it plays a key role in highlighting key features needed for classification of images. Using Tensor Flow and Keras, you can add a few more layers on top of a readymade neural network to build the model you need. Details on the loss function, choice of activation function, and number of trainable parameters are all included in the report. Checking the accuracy and loss reveals that the model performs well on a task with an accuracy of 90.26% and a loss of only 0.18. The research found that Seresnet152 and ResNet do a good job with chest X-ray image classification, which might help doctors and nurses better understand what the images reveal. Results from the study affirm that medical results are better when image data is processed carefully, a suitable model is selected, and the correctness of the model can be verified in deep learning efforts. You will be shown how the suggested models compare to the ones currently in use, especially in how accurately they work. 92.56% of the time, ECA-net is right, while Efficient Net V2 is right in 91.55% of cases and ECNN is right 93% of the time. The Res Net model showed better performance than the other models, improving the accuracy by 94.67%. The new Res Net model does a better job of organizing data, so it's great for positions where it needs to be precise and dependable with its results.

References

1. L. Zhang *et al.*, "Learning from multiple annotators for medical image segmentation," *Pattern Recognit.*, vol. 138, p. 109400, 2023, doi: 10.1016/j.patcog.2023.109400.
2. H. Allioui, Y. Mourdi, and M. Sadgal, "Strong semantic segmentation for Covid-19 detection: Evaluating the use of deep learning models as a performant tool in radiography," *Radiography*, vol. 29, no. 1, pp. 109–118, 2023, doi: 10.1016/j.radi.2022.10.010.
3. M. E. Karar, Z. F. Khan, H. Alshahrani, and O. Reyad, "Smart IoMT-based segmentation of coronavirus infections using lung CT scans," *Alexandria Eng. J.*, vol. 69, pp. 571–583, 2023, doi: 10.1016/j.aej.2023.02.020.
4. D. Anantha, S. Roy, S. Kumar, and R. Tripathi, "ScienceDirect A Scheme for E ff ective Skin Disease Detection using Optimized A Scheme for E ff ective Skin Disease Detection using Optimized Region Growing Segmentation and

Autoencoder based Region Growing Segmentation and Autoencoder based Classificatio," *Procedia Comput. Sci.*, vol. 218, pp. 274–282, 2023, doi: 10.1016/j.procs.2023.01.009.

5. H. Chauhan and K. Modi, "Science Direct Science Direct ScienceDirect AMSFMap Methodology to improve prediction accuracy of CNN AMSFMap Methodology to improve prediction accuracy of CNN model for Covid19 using X-ray images model for Covid19 using X-ray images," *Procedia Comput. Sci.*, vol. 218, no. 2022, pp. 1394–1404, 2023, doi: 10.1016/j.procs.2023.01.118.

6. G. S. Anushia and S. Hema, "Lung Disease Classification Using CNN," pp. 107–116, 2023, doi: 10.1007/978-981-99-4577-1_9.

7. V. Rajinikanth, "ScienceDirect ScienceDirect ScienceDirect International Conference on Machine Learning and Data Engineering Automatic detection of lung nodule Learning in CT scan slices using CNN segmentation schemes : A study Automatic detection of lung nodule b in CT s," *Procedia Comput. Sci.*, vol. 218, no. 2022, pp. 2786–2794, 2023, doi: 10.1016/j.procs.2023.01.250.

8. S. Arvind, J. V. Tembhurne, T. Diwan, and P. Sahare, "Improvised light weight deep CNN based U-Net for the semantic segmentation of lungs from chest X-rays," *Results Eng.*, vol. 17, no. July 2022, p. 100929, 2023, doi: 10.1016/j.rineng.2023.100929.

9. C. Ladbury *et al.*, "Integration of artificial intelligence in lung cancer: Rise of the machine," *Cell Reports Med.*, vol. 4, no. 2, p. 100933, 2023, doi: 10.1016/j.xcrm.2023.100933.

10. C. A. Secondary *et al.*, "CT-based automatic spine segmentation using patch-based deep learning," *Unpublished*, vol. 2023, no. Dc, 2023.

11. M. F. Aslan, K. Sabanci, A. Durdu, and M. F. Unlersen, "COVID-19 diagnosis using state-of-the-art CNN architecture features and Bayesian Optimization," *Comput. Biol. Med.*, vol. 142, no. January, p. 105244, 2022, doi: 10.1016/j.compbiomed.2022.105244.

12. R. S, C. B M, D. M. Ramani, and M. S. Harsur, "CNN based multi-view classification and ROI segmentation: A survey," *Glob.*

Transitions Proc., vol. 3, no. 1, pp. 86–90, 2022, doi: 10.1016/j.gltp.2022.04.019.

13. C. F. Li *et al.*, "MultiR-Net: A Novel Joint Learning Network for COVID-19 segmentation and classification Cheng-Fan," *Comput. Biol. Med.*, vol. 144, no. January, p. 105340, 2022, doi: 10.1016/j.compbiomed.2022.105340.
14. L. Lian, X. Luo, C. Pan, J. Huang, W. Hong, and Z. Xu, "Lung image segmentation based on DRD U-Net and combined WGAN with Deep Neural Network," *Comput. Methods Programs Biomed.*, vol. 226, p. 107097, 2022, doi: 10.1016/j.cmpb.2022.107097.
15. N. Aslan, G. Ozmen Koca, M. A. Kobat, and S. Dogan, "Multi-classification deep CNN model for diagnosing COVID-19 using iterative neighborhood component analysis and iterative ReliefF feature selection techniques with X-ray images," *Chemom. Intell. Lab. Syst.*, vol. 224, no. November 2021, p. 104539, 2022, doi: 10.1016/j.chemolab.2022.104539.
16. M. F. Aslan, "A robust semantic lung segmentation study for CNN-based COVID-19 diagnosis," *Chemom. Intell. Lab. Syst.*, vol. 231, no. July, p. 104695, 2022, doi: 10.1016/j.chemolab.2022.104695.
17. J. Chi, S. Zhang, X. Han, H. Wang, C. Wu, and X. Yu, "MID-UNet: Multi-input directional UNet for COVID-19 lung infection segmentation from CT images," *Signal Process. Image Commun.*, vol. 108, no. August, p. 116835, 2022, doi: 10.1016/j.image.2022.116835.
18. H. Min Kim, T. Ko, I. Young Choi, and J. P. Myong, "Asbestosis diagnosis algorithm combining the lung segmentation method and deep learning model in computed tomography image," *Int. J. Med. Inform.*, vol. 158, p. 104667, 2022, doi: 10.1016/j.ijmedinf.2021.104667.
19. S. Padmakala, S. Revathy, K. Vijayalakshmi, and M. Mathankumar, "CNN supported automated recognition of Covid-19 infection in chest X-ray images," *Mater. Today Proc.*, vol. 66, pp. 1201–1210, 2022, doi: 10.1016/j.matpr.2022.05.003.
20. R. Hertel and R. Benlamri, "A deep learning segmentation-classification pipeline for X-ray-based COVID-19 diagnosis," *Biomed. Eng. Adv.*, vol. 3, no.

May, p. 100041, 2022, doi: 10.1016/j.bea.2022.100041.

21. G. Rani *et al.*, “A multi-modal bone suppression, lung segmentation, and classification approach for accurate COVID-19 detection using chest radiographs,” *Intell. Syst. with Appl.*, vol. 16, no. June, p. 200148, 2022, doi: 10.1016/j.iswa.2022.200148.

22. P. Ghose, M. A. Uddin, U. K. Acharjee, and S. Sharmin, “Deep viewing for the identification of Covid-19 infection status from chest X-Ray image using CNN based architecture,” *Intell. Syst. with Appl.*, vol. 16, no. September, p. 200130, 2022, doi: 10.1016/j.iswa.2022.200130.

23. S. Goyal and R. Singh, “Detection and classification of lung diseases for pneumonia and Covid-19 using machine and deep learning techniques,” *J. Ambient Intell. Humaniz. Comput.*, vol. 14, no. 4, pp. 3239–3259, 2023, doi: 10.1007/s12652-021-03464-7.

24. A. M. Q. Farhan and S. Yang, “Automatic lung disease classification from the chest X-ray images using hybrid deep learning algorithm,” *Multimed. Tools Appl.*, vol. 82, no. 25, pp. 38561–38587, 2023, doi: 10.1007/s11042-023-15047-z.

25. M. Jasmine Pemeena Priyadarsini *et al.*, “Lung Diseases Detection Using Various Deep Learning Algorithms,” *J. Healthc. Eng.*, vol. 2023, 2023, doi: 10.1155/2023/3563696.

26. P. Zhang, A. Swaminathan, and A. A. Uddin, “Pulmonary disease detection and classification in patient respiratory audio files using long short-term memory neural networks,” *Front. Med.*, vol. 10, 2023, doi: 10.3389/fmed.2023.1269784.

27. S. P. Pawar and S. N. Talbar, “Two-Stage Hybrid Approach of Deep Learning Networks for Interstitial Lung Disease Classification,” *Biomed Res. Int.*, vol. 2022, 2022, doi: 10.1155/2022/7340902.

28. S. Marappan, M. D. Mujib, A. A. Siddiqui, A. Aziz, S. Khan, and M. Singh, “Lightweight Deep Learning Classification Model for Identifying Low-Resolution CT Images of Lung Cancer,” *Comput. Intell. Neurosci.*, vol. 2022, 2022, doi: 10.1155/2022/3836539.

29. G. Kasinathan and S. Jayakumar, “Cloud-Based Lung Tumor Detection and Stage Classification

Using Deep Learning Techniques," *Biomed Res. Int.*, vol. 2022, 2022, doi: 10.1155/2022/4185835.

30. W. Tang and L. Gu, "Harmonic Classification with Enhancing Music Using Deep Learning Techniques," *Complexity*, vol. 2021, 2021, doi: 10.1155/2021/5590996.

31. Z. Tariq, S. K. Shah, and Y. Lee, "Multimodal Lung Disease Classification using Deep Convolutional Neural Network," *Proc. - 2020 IEEE Int. Conf. Bioinforma. Biomed. BIBM 2020*, pp. 2530–2537, 2020, doi: 10.1109/BIBM49941.2020.9313208.

32. M. Zak and A. Krzyżak, "Classification of lung diseases using deep learning models," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 12139 LNCS, pp. 621–634, 2020, doi: 10.1007/978-3-030-50420-5_47.

33. M. K. Montgomery *et al.*, "Applying deep learning to segmentation of murine lung tumors in pre-clinical micro-computed tomography," *Transl. Oncol.*, vol. 40, no. June 2023, p. 101833, 2024, doi: 10.1016/j.tranon.2023.101833.

34. F. Zhang *et al.*, "Enhancing non-small cell lung cancer tumor segmentation with a novel two-step deep learning approach," *J. Radiat. Res. Appl. Sci.*, vol. 17, no. 1, p. 100775, 2024, doi: 10.1016/j.jrras.2023.100775.

35. S. K. . Sangeetha *et al.*, "An Enhanced Multimodal Fusion Deep Learning Neural Network for Lung Cancer Classification," *Syst. Soft Comput.*, vol. 6, no. October 2023, p. 200068, 2023, doi: 10.1016/j.sasc.2023.200068.

36. G. M. M. Alshmrani, Q. Ni, R. Jiang, H. Pervaiz, and N. M. Elshennawy, "A deep learning architecture for multi-class lung diseases classification using chest X-ray (CXR) images," *Alexandria Eng. J.*, vol. 64, pp. 923–935, 2023, doi: 10.1016/j.aej.2022.10.053.

37. Y. Choi and H. Lee, "Interpretation of lung disease classification with light attention connected module," *Biomed. Signal Process. Control*, vol. 84, no. February, p. 104695, 2023, doi: 10.1016/j.bspc.2023.104695.

38. "NIH Chest X-rays." https://www.kaggle.com/datasets/n ih-chest-xrays/data (accessed Feb. 21, 2024).

39. S. Saiwaeo, S. Arwatchananukul, L. Mungmai, W. Preedalikit, and N. Aunsri, "Human skin type classification using image processing and deep learning approaches," *Heliyon*, vol. 9, no. 11, p. e21176, 2023, doi: 10.1016/j.heliyon.2023.e21176.

40. V. Yuvaraj and D. Maheswari, "Lung cancer classification based on enhanced deep learning using gene expression data," *Meas. Sensors*, vol. 30, no. October, p. 100902, 2023, doi: 10.1016/j.measen.2023.100902.