CERTIFICATE

This is to certify that the project entitled **"Lung Disease Classification"** is a bonafide work carried out by **Balla Pradeep,** student of VII Semester B.Tech. (AI), Department of Information Technology, National Institute of Technology Karnataka, Surathkal, during the **summer** term of the academic year 2024-2025. It is submitted to the Department in partial fulfillment of the requirements for the award of the degree of Bachelor of Technology in Information Technology. I certify that he has carried out the work in his own capacity and has successfully completed the work assigned to him. The aforementioned work was carried out during the period **24th April** to **12th July.**

Place: NITK Surathkal

Date: ~~28 August~~ 2024
04 October

(Signature of the Guide/Mentor)

Name: Dr. Ananthanarayana V.S.

Designation: Professor

Organization: NITK Surathkal

Address: NH 66, Srinivasnagar,

Surathkal, Mangaluru,

Karnataka 575025

Phone: +91 8242473550

A Practical Training Report (IT447)

on

Lung Disease Classification

undergone at

Department Of Information Technology

under the guidance of

Dr. Ananthanarayana V. S.

Submitted by

Balla Pradeep

211AI009

VII Semester B.Tech (AI)

in partial fulfillment for the award of the degree of

BACHELOR OF TECHNOLOGY

in

INFORMATION TECHNOLOGY

Department Of Information Technology
National Institute of Technology Karnataka, Surathkal.

August 2024

DECLARATION BY THE STUDENT

I, **Balla Pradeep,** hereby declare that the Practical Training work entitled "**Lung Disease Classification**", was carried out by me from 24th April to 12th July **of duration 12 weeks** term of the academic year 2024-2025. I declare that this is my original work and has been completed successfully under the guidance of Dr. Ananthanarayana V.S , Professor(HAG Scale) at **Department of Information Technology a**nd as per the specifications of NITK Surathkal.

Place: Surathkal

Date: 28th July 2024

(Signature of the Student)

ACKNOWLEDGEMENTS

I would like to express my sincere gratitude to my project supervisor, Dr. Ananthanarayana V.S for their invaluable guidance, support, and encouragement throughout the project. Their expertise and insights were crucial in shaping the direction and execution of the project, and their feedback helped me overcome various challenges. Thank you!

ABSTRACT

Deep learning has significantly advanced the field of medical image analysis, particularly in the classification of lung diseases using chest X-ray (CXR) images. This report presents a comprehensive study where multiple convolutional neural network (CNN) models, including VGG19, DenseNet121, EfficientNetB0, VGG16, Xception, and ResNet50, were employed to detect various lung conditions such as Viral Pneumonia, Lung Opacity, and COVID-19. Using the COVID-19 Radiology Database, we selected and preprocessed 1,300 images per class across four distinct classes, followed by a test-train-validation split.

Among the models evaluated, the Xception model achieved the highest performance with a training accuracy of 99.6% and a test accuracy of 96.2%. To enhance model performance, an ensemble model combining ResNet50, Xception, and VGG16 was developed. The stacking ensemble model, in particular, achieved an impressive accuracy of 97.60%. We further explored various ensemble techniques, including stacking, boosting, and voting, and compared their results. Hyperparameter tuning was performed using Grid Search Optimization, and data augmentation techniques were investigated to improve model robustness.

Additionally, the ensemble model was trained and tested on a second dataset, with results compared against existing literature. This comprehensive analysis not only demonstrates the effectiveness of deep learning models in lung disease classification but also highlights the potential of ensemble methods in improving diagnostic accuracy. The ensemble model

CONTENTS

LIST OF FIGURES

1. INTRODUCTION

1.1. Challenges

Research studies on lung disease identification and treatment have generated significant interest globally. Most scientific studies have found that various lung disease symptoms is seen in X-Ray and CT images of the lungs. The availability of X-Ray and CT scans has made these two imaging methods suitable for early detection of lung diseases. To speed up image analysis, many efforts have been made to implement artificial intelligence (AI) to improve the healthcare system. The major successes of deep learning (DL) approaches in detecting certain irregularities in medical images have motivated researchers to explore deep CNN architectures further for lung disease classification in both X-Ray and CT scans.

The coronavirus disease 2019, known as COVID-19, is a contagious illness caused by a virus called the severe acute respiratory syndrome coronavirus 2 (SARS-CoV-2). The symptoms of COVID-19 can vary, but commonly include fever, cough, headache, fatigue, difficulty breathing, loss of smell, and loss of taste. These symptoms may start appearing between one to fourteen days after being exposed to the virus. Interestingly, at least a third of infected people don't experience noticeable symptoms. For those who do develop noticeable symptoms, most (81%) have mild to moderate symptoms, like mild pneumonia, while 14% have severe symptoms like breathing problems, low oxygen levels, or extensive lung involvement. Only 5% of patients develop critical symptoms such as respiratory failure, shock, or issues with multiple organs.

1.2. Motivation for the work

Doctors check CXR images to find lung diseases. They do this because it's easy and doesn't hurt. CXR images show what's happening in the chest. They also show if a disease changes over time. But sometimes doctors make mistakes reading CXR images. The chest has a complicated structure, so it's hard to understand the images. That's why computer programs help doctors make better decisions. The programs diagnose diseases more accurately than doctors alone. Many methods use CXR images to make these programs work better

Deep learning really stands out from normal machine learning when analyzing medical images. It shines for classification, detection, and segmentation tasks. This technique works great with different data types: 1D signals, 2D images, and even 3D scans. Unlike natural RGB images, chest X-rays are grayscale with one channel of info. These radiographs capture X-ray radiation passing through the body. Calcium-rich areas like bones block radiation, appearing white. But soft tissues like lungs let radiation through, showing up as gray or black.

Some research taught a DL model to read one-channel grayscale CXR images from scratch. Yet gathering and labeling medical data is a time-consuming, expensive task requiring experts. Although CXR images differ from regular RGB pictures, a model trained on RGB may identify general features useful for lung disease. Deep learning models initially struggled when trained solely on converted three-channel grayscale CXR data. But the latest solution leverages transfer learning to apply knowledge from natural RGB image models, providing a more effective starting point for identifying relevant patterns in the medical scans.

2. LITERATURE SURVEY

2.1. Introduction to Literature Survey

The rapid advancement of medical imaging technologies, particularly chest X-ray (CXR) imaging, has revolutionized the field of pulmonary disease diagnosis. The integration of deep learning and artificial intelligence (AI) into medical imaging has led to significant improvements in the accuracy and speed of lung disease classification, which is critical for early diagnosis and treatment. This literature survey aims to provide a comprehensive overview of recent research efforts in lung disease classification using CXR images, focusing on the application of deep learning models, ensemble techniques, and explainable AI.

This literature survey provides an in-depth analysis of these studies, highlighting the strengths and limitations of various methodologies. By examining the latest advancements in lung disease classification, this survey aims to contribute to the ongoing efforts to develop more accurate, efficient, and explainable AI-based diagnostic tools.

2.2. Related Work

1. K. Vij, S. Thakur, A. Sharma and R. Mohana, "Lung Disease Classification using X-Ray Imaging with Ensemble Learning," *2023* In this paper, they have proposed an ensemble model using two popular pretrained models Res-Net and VGG19 achieved Validation Accuracy of 97.62% using 85,000 frontal chest X-ray images in their dataset.

2. Tanzina Taher Ifty , et.al, “Explainable Lung Disease Classification from Chest X-Ray Images Utilizing Deep Learning and XAI”, 2024 In their research they used various methodologies like hyperparameter tuning, stratified k-fold cross-validation, and transfer

learning with fine-tuning and achieved an accuracy of 96.21%. They have also explored the explainable AI methodologies in their research.

3. Apostolopoulos, I.D., Mpesiana, T.A. “Covid-19: automatic detection from X-ray images utilizing transfer learning with convolutional neural networks”2023. In this paper, they have utilized the transfer learning techniques and achieved an accuracy of 98.75% (using 2-class)with the VGG19 model. However they have only used 227 xray images in their research.

4. P. Mahajan, N. Karlupia, P. Abrol and P. K. Lehana, "Identifying COVID-19 Pneumonia using Chest Radiography using Deep Convolutional Neural Networks," *2021.* In this paper, the authors aim to extract those regions rapidly that may contain features of COVID-19 from chest X-ray images and further classify the possible existence of Covid disease. Their proposed approach can efficiently contribute to the detection of COVID-19 disease with 91.08% accuracy with a minimum loss of 0.0846 for 3-class based classification.

5. A. Gokarn, K. Patni, Y. Purohit and R. Sonkusare, "COVID-19 Radiography Using ConvNets," *2022.* In this study, CNN based model have been proposed for the detection of coronavirus pneumonia infected patients using chest X-ray radiography and gives a classification accuracy of 93.77% (training accuracy of 99.81% and validation accuracy of 95.45%).

6. A. C. Mazari and H. Kheddar, "Deep Learning-and Transfer Learning-based Models for COVID-19 Detection using Radiography Images," *2023.* In this study, They used two Transfer Learning models, namely RestNet50 and MobileNetV2 using the pretrained model of the ImageNet database, experimented on the new dataset (COVID-QU-Ex Dataset 2022) offered by the University of Qatar. The results achieved by CNN (Acc =95.97%), ResNet50 (Acc =95.53%) and MobileNetV2 (Acc =97.32%) using a 2-class classification.

7. Goram Mufarah M. Alshmrani , et.al “A deep learning architecture for multi-class lung diseases classification using chest X-ray (CXR) images” 2023. In their proposed model,The experimental results revealed that thei proposed VGG19 + CNN outperformed other existing work with 96.48 % accuracy.

2.3. Outcome of Literature Review

The literature survey on lung disease classification using chest X-ray images has revealed significant advancements in the application of deep learning and ensemble learning techniques. The survey highlights that models such as ResNet, VGG19, and CNN have consistently demonstrated high classification accuracies, often exceeding 90%, with some approaches reaching up to 98.75%. The integration of transfer learning has proven to be particularly effective, allowing models to achieve superior performance even with limited data, as evidenced by multiple studies.

Ensemble learning approaches, like the combination of ResNet and VGG19, have shown promise in enhancing model robustness and accuracy. Additionally, the exploration of explainable AI (XAI) methodologies is a notable trend, addressing the critical need for transparency and interpretability in AI-driven medical diagnoses. However, the survey also identifies challenges such as the need for larger and more diverse datasets to improve model generalization and the necessity for further research into multi-class classification tasks.

A comparison of various authors and the deep learning models they used for classifying lung diseases from different chest X-ray datasets, along with the achieved accuracy is provided in the table. It highlights the use of models like VGG19, ResNet50, and InceptionV3, with accuracies ranging from 91.06% to 98.75%.

Author	Models Used	Dataset	Accuracy
K. Vij [1]	VGG19+ResNet50	Chest X-rays	97.60%
Tanzina Taher Ifty [2]	InceptionV3+Xception	COVID-19 Detection X-Ray Dataset	96.21%
Apostolopoulos [3]	VGG19	Github Repository	98.75%
P. Mahajan [4]	InceptionV3	Covid-19 Radiography Dataset	91.06%
A. Gokarn [5]	VGG19	Covid-19 Radiography Dataset	95.45%
A. C. Mazari [6]	MobileNetV2	Covid-19 Radiography Dataset	97.32%
Goram Mufarah [7]	VGG19 + CNN	Multiple Public Datasets	96.48%

Table 1 *Outcome of the existing literature survey*

2.4. Problem Statement

To develop an automated system that accurately classifies lung diseases such as COVID-19, Viral Pneumonia, and Lung Opacity from chest X-ray images, providing fast, reliable, and scalable solutions for healthcare professionals.

2.5. Research Objectives

The primary goal of this research project is to improve the existing architecture by utilizing different techniques. To achieve this, the project focuses on the following objectives:

1. **Develop and Evaluate CNN Models**: Train and evaluate multiple convolutional neural network (CNN) models, including VGG19, DenseNet121, EfficientNetB0,

VGG16, Xception, and ResNet50, on a large COVID-19 Radiology Database to classify lung diseases such as COVID-19, Viral Pneumonia, and Lung Opacity.

2. **Ensemble Model Development**: Create and optimize an ensemble model combining the strengths of individual CNN models (ResNet50, Xception, and VGG16) to improve classification accuracy.
3. **Explore Ensemble Techniques**: Investigate various ensemble techniques, including stacking, boosting, and voting, to identify the most effective method for improving model performance.
4. **Comparison with Existing Literature**: Compare the performance of the proposed models and ensemble techniques against existing models and approaches documented in the literature.
5. **Apply Models to a Second Dataset**: Test the developed ensemble model on a second dataset to evaluate its generalization capability and compare the results with the original dataset.

3. METHODOLOGY AND FRAMEWORK

3.1. System Architecture

The system architecture for this lung disease classification project is built on a series of components that work together to process chest X-rays and classify lung diseases. Below

is a detailed explanation of each step and the system architecture for lung disease classification, is shown in **Figure 1**:

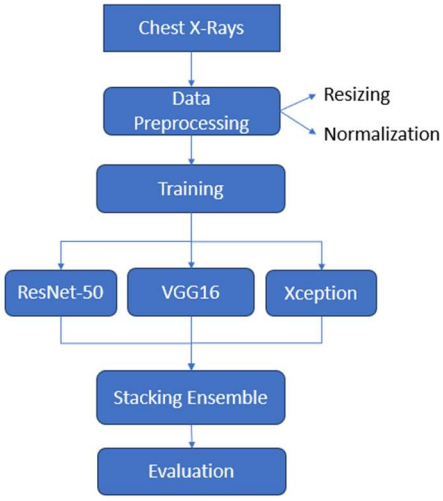

Figure-*1* *Flow Chart*

3.1.1. Chest X-Rays Dataset

The system takes raw chest X-ray images as input. These images are collected from a medical dataset and serve as the starting point for the entire classification process.

3.1.2 Data Preprocessing

Preprocessing plays a crucial role in preparing the raw images for effective model training. The following operations are applied:

Resizing: Images are resized to 299x299 pixels to match the input requirements of the deep learning models.

Normalization: The pixel values are normalized to a range of 0 to 1, which aids in faster convergence during training.

3.1.3 Training

Once the images are preprocessed, they are passed into multiple pre-trained deep learning models for training. In this case, ResNet-50, VGG16, VGG19, Densenet121, MobileNetV2 and Xception are used. These models are fine-tuned on the chest X-ray dataset, learning to extract meaningful features to distinguish between lung diseases.

3.1.4. Stacking Ensemble

After training the models, their predictions are combined using a stacking ensemble method:

Base Learners: ResNet-50, VGG16, and Xception each make their own predictions based on the features extracted from the chest X-rays.

Meta-Model: Logistic regression is used as the meta-model, taking the output predictions from the three models and making the final decision. This ensemble strategy improves overall classification performance by leveraging the complementary strengths of the base learners.

3.1.5. Evaluation

The final step is to evaluate the performance of the ensemble model. The metrics like Accuracy, Recall, F1-Score are used to assess its performance.

3.2. Algorithms, Techniques, and Tools

The project utilizes a variety of algorithms, techniques, and tools to implement the system architecture.

3.2.1. Datasets Used

In this Project we used chest x-ray images of 2 datasets were used these two datasets are collected from various sources.

a. **Dataset-1:** COVID-19 Radiology Database

A team of researchers from Qatar University, Doha, Qatar, and the University of Dhaka, Bangladesh along with their collaborators from Pakistan and Malaysia in collaboration with medical doctors have created a database of chest X-ray images for COVID-19 positive cases along with Normal, Lung Opacity and Viral Pneumonia images.

The dataset contains chest x-ray images of COVID-19, Normal, Lung Opacity, Viral Pneumonia labels.

• Covid-19 : 3616 images

• Normal : 10,192 images

• Viral Pneumonia : 1345 images

• Lung Opacity : 6012 images

All the images are in 299*299 pixels resolution. In this Project
I have used 1300 images each of all the four classes because of resource constraints and time constraints.

b. Dataset-2

Link:https://www.kaggle.com/datasets/omkarmanohardalvi/lungs-disease-dataset-4-types
The Dataset contains chest x-rays images. This Dataset was prepared from various datasets combined accordingly. It has 4 types of Lung diseases and a folder of Normal Lungs. Augmented the dataset with factor 6 so there are basically 10000 images.
Number of Classes: 3 (Covid19, Viral Pneumonia, Normal)
Approximately 1300 images in each class

3.3. Model Architectures

Initially, we tried a CNN model from a research paper. This was our starting point to measure other models. We trained and tested this custom model. We wanted to see how well it did. After testing the custom model, we looked at other pretrained models. We used models like EfficientNetB0, ResNet50, DenseNet121, VGG16, and VGG19. These models were already trained on huge datasets. They are known to work really well for computer vision tasks. We wanted to use their learned skills. We hoped they would do better than our custom model.

3.3.1. VGG-16 Model

VGG-16 excels at tasks like image classification and object recognition. The model learns intricate visual features via its stack of convolutional layers, each followed by max-pooling. This unique architecture allows VGG-16 to make accurate, robust predictions by capturing hierarchical representations of images. While simple in concept, the depth of VGG-16 grants it remarkable capabilities. The VGG16 model has a total of 39.9m parameters in which 25.5m parameters are trainable pparameters.

3.3.2. VGG-19 Model

The VGG19 model is VGGNet-19. It works like VGG16, but has more number of layers. VGGNet-19 model has nineteen weight layers. The numbers sixteen and nineteen stand for convolutional layers. VGG19 has three extra convolutional layers compared to VGG16. VGG19 Model has a total of 45.2m parameters in which a 25m parameters are trainable parameters.

3.3.3. DenseNet121 Model

DenseNet121 is a convolutional neural network. It makes each layer connect to all other layers. This allows feature maps from earlier layers to be used by later ones. DenseNet121 promotes reusing features. This helps reduce information loss. It emphasizes feature reuse through dense connectivity in a feed-forward way. In Traditional CNN layers except the first layer rest all recieves the output from the previous layers and passes to the next convolutional layer. As the number of layers increase the vanishing gradient problem comes into place. This means the information path from input to output layer increases it can cause information to vanish or reduce and can effect the training process. DenseNets resolves the problem of vanishing gradient problem by changing the traditional CNN architecture and simplifies the connection between layers. The model which we used has a total of 8m parameters in which 1m are trainable parameters

3.3.4. Xception Model

Xception is a Convolutional neural network. It was made by Franc¸ois Chollet. Xception is good at computer vision tasks. It works well and doesn't waste time. The network is based on inception. But it uses different convolution blocks. These blocks are called depthwise separable convolutions. They make the network better and faster. We have utilized the Xception model as a feature extractor, which means including the top (classification) layer of the pretrained Xception model. This allows us to use the pretrained Xception model as a feature extractor and add our own classification layers on top.

We have also loaded the weights of the Xception model pretrained on the ImageNet dataset by setting weights to 'imagenet'. This initializes the model with weights learned from the ImageNet dataset, which can be beneficial for transfer learning tasks. we have defined additional layers to be added on top of the Xception base. These include a global average pooling layer (GlobalAveragePooling2D), a dropout layer (Dropout), batch normalization layer (Batch Normalization), and a dense output layer (Dense) with softmax activation function. in Xception Model which we used there are a total of 20.88m parameters

4. WORK DONE

4.1. Development Environment

4.1.1. Hardware Setup

The hardware setup is designed to provide a robust and flexible platform for developing and testing the hardened operating system.

Machine Specifications: These are my system specifications:

- Processor (Intel Core i5)
- RAM (16GB)

- GPU (NVIDIA GTX 1650)
- Storage (512GB SSD)

Google colab's memory resources are displayed in the figure 2.

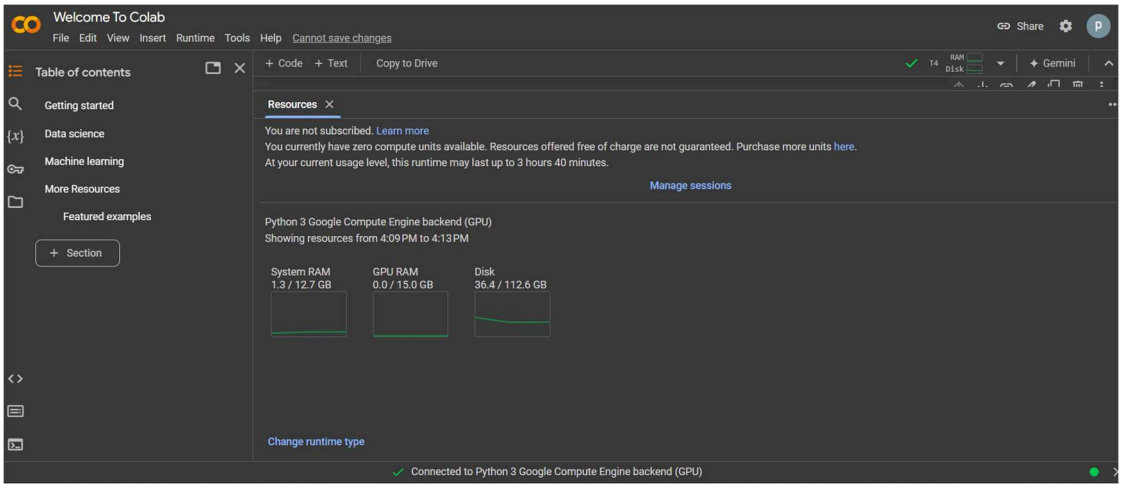

Figure 2 *Google Colab Ram and Disk usage*

4.1.2. Software and Tools

A comprehensive suite of software and tools is utilized to streamline the development process and ensure the effective implementation of Lung disease classification.

Development Tools: Google Colab for writing, editing, implementing code.
Programming Languages Used: Python
Libraries and Frameworks: TensorFlow, Keras, PyTorch, Scikit-learn, OpenCV.
Data Handling: Pandas, NumPy.

4.2. Results and Analysis

4.2.1. Dataset Exploration

Used 1300 images of each class.
All the images are coloured. All the images are resized to 299x299 and normalization is done.

Converted the images into Numpy arrays and applied test-train split

Train data-(3751, 299, 299, 3)

Test data-(1042, 299, 299, 3)

Validation data-(417, 299, 299, 3)

The required packages are to be installed as shown in figure 3:

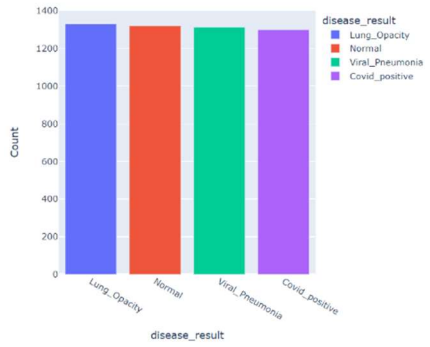

Figure 3 *count of number of images in each category*

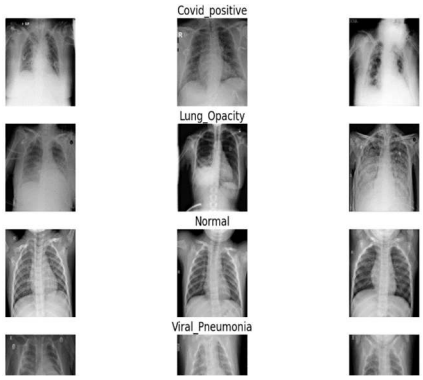

Figure 4 *A random sample of images*

4.2.2. Training

At first the images are processed that is they are converted into NumPy arrays and the resulting array is stored to a CSV file for further process reducing the time to process images every time. Later all the images are resized to 299x299 pixels and normalized. After preparing the data, images and labels are separated and the dataset is divided into training,

testing and validation sets using train test split function. The training set consists of 80 percent and 20 percent into testing and validation sets. Used several pre trained models like ResNet50, Inception V3, MobileNet V2, Xception, DenseNet121, VGG 19, VGG 16. Out of them Resnet50, MobileNet V2, Xception, VGG16 were taken for making an ensemble model.

4.2.2.1 ResNet50 Model

Initial Training: Freeze the base model layers and train only the new layers on top. This adapts the model to the new task without losing the general features learned from the large dataset.

Fine-tuning: Unfreeze some of the deeper layers (closer to the output) and continue training with a lower learning rate. This fine-tunes the model, allowing it to learn more specific features for the new task.

Trained on batch_size of 64 and 20 epochs, early stopping if there is no decrease in validation loss. Figure 5 presents the training and validation accuracy, along with the loss curves, for the ResNet50 model, providing insights into the model's performance during training and evaluation phases.

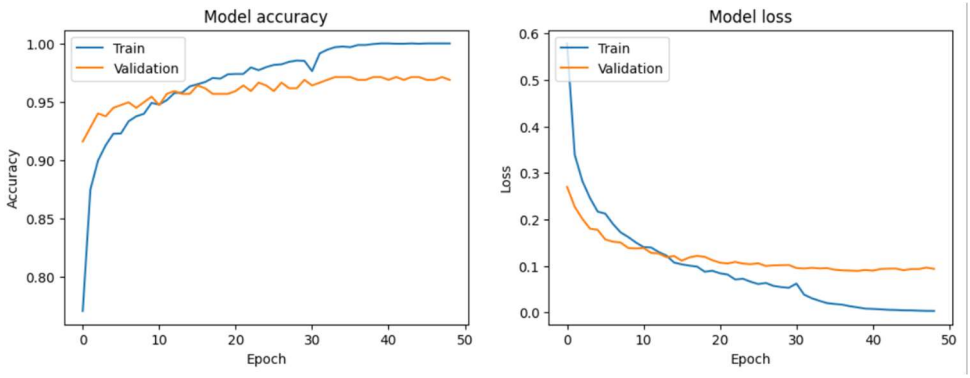

Figure 5 *ResNet50 model train and validation accuracy, loss plots*

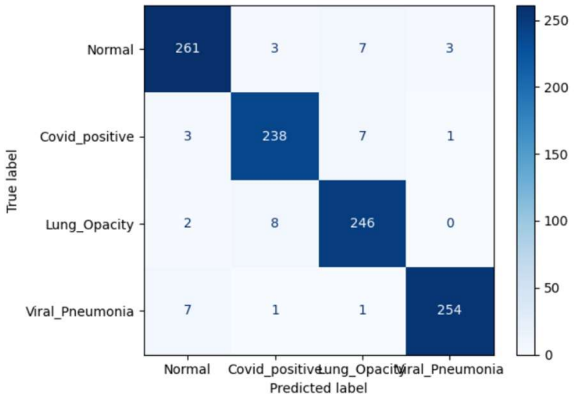

Figure 6 *Confusion Matrix for ResNet50 model*

In the figure 6, shows the confusion matrix for Resnet50 model. Most of the classes are correctly predicted leaving very less outliers in all the classes

4.2.2.2 Xception Model

This model is trained for 20 epochs and batch size of 64 using adam optimizer

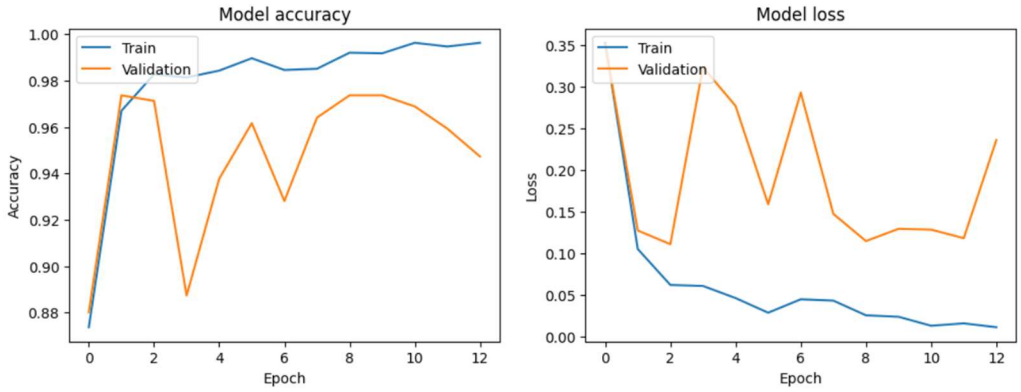

Figure 7 *Xception Model training and validation accuracy, loss plots.*

Figure 6 illustrates the training and validation accuracy, as well as the loss curves, for the Xception model, highlighting its performance throughout the training process. The training accuracy is consistently high across the epochs 1-12, but the validation accuracy is fluctuating suggesting there might be overfitting due to less generalization

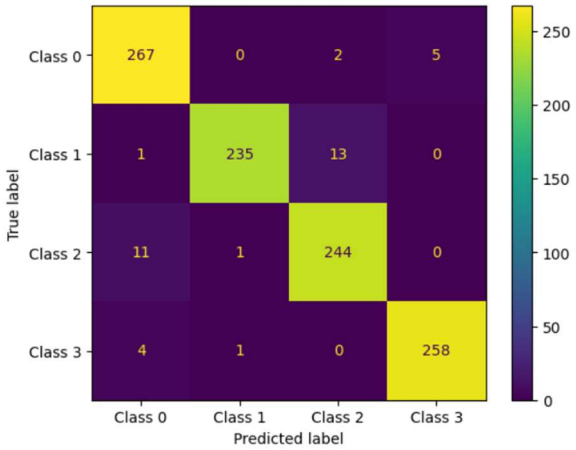

Figure 8 *Confusion Matrix for Xception Model test data*

In figure 8, Lung opacity(Class 2) has the most significant number of misclassifications, being confused with normal cases.

4.2.2.3 MobileNet V2 Model

This model also uses similar fine-tuning technique as resnet 50 model.

Trained this model for 100 epochs of batch size 64 using adam optimizer. With early stopping of patience = 10. **Figure 9** displays the training and validation accuracy along with the loss plots for the MobileNetV2 model, offering a visual representation of its performance during the training and validation phases.

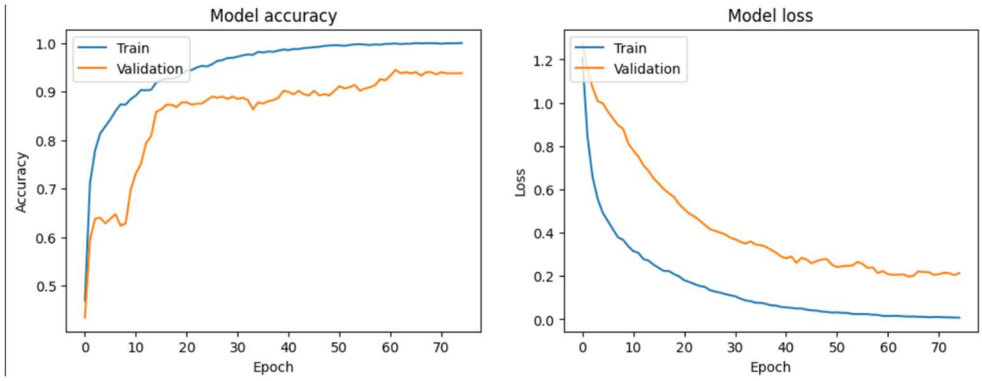

Figure 9 *MobileNetV2 Model training and validation accuracy, loss plots*

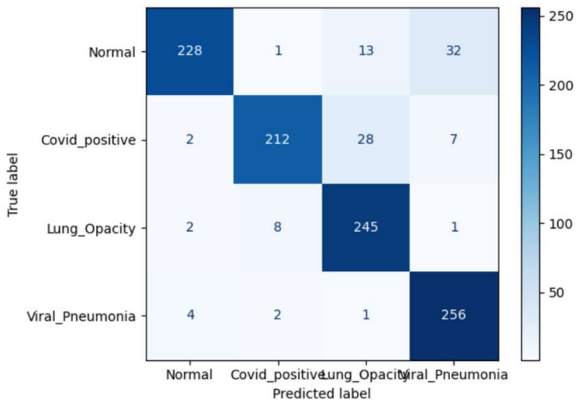

Figure 10 *confusion matrix for MobileNet V2 model test set.*

In figure 10, some of the normal images misclassified as lung opacity and viral pneumonia and some of the covid images are miss classified as lung opacity.

This suggests that the model struggles somewhat for deciding lung opacity class, where normal and covid images are misclassified as lung opacity.

4.2.2.4 Vgg16 Model

VGG16 is known for its deep architecture with 16 layers, which allows it to capture complex patterns and features in images. This model has demonstrated excellent performance on large-scale image recognition tasks, making it a robust choice for transfer learning.

This model was trained on 50 epochs, batch size of 64. Achieved validation accuracy of 97.2% and validation loss of 0.09. **Figure 11** shows the training and validation accuracy, along with the loss plots, for the VGG16 model, providing a detailed overview of its performance throughout the training and validation stages.

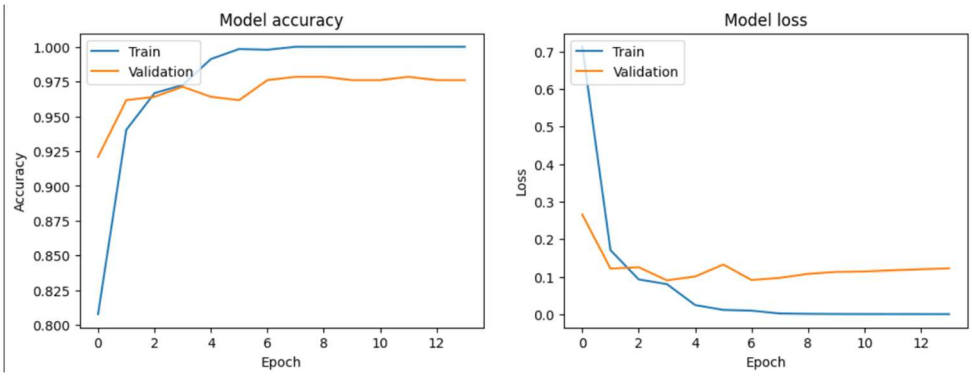

Figure 11 *VGG16 Model training and validation accuracy, loss plots*

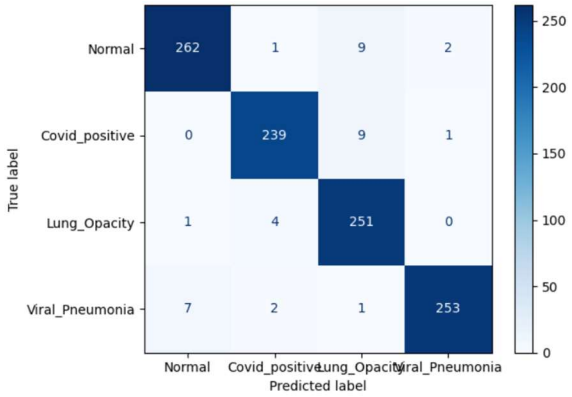

Figure 12 *confusion matrix for vgg16 model*

Figure 12 presents the confusion matrix for the VGG16 model, illustrating the model's classification performance by showing the number of correct and incorrect predictions across the various classes.

I tried using data augmentation for this model. It didn't perform well and it took more time to complete the run.
I also tried Grid Search Optimization in the hyperparameter optimization and got better results.

4.2.3. Hyperparameter Tuning

Grid Search Optimization: Grid search is an optimization technique used to brute force through all possible combinations of a set of variables. It works by creating a grid of all possible combinations of parameter values and testing each combination to find

the best one. **Figure 13** displays the parameters grid, detailing the hyperparameter settings explored during model tuning to optimize performance.

```
param_grid = {
    'learning_rate': [1e-5, 1e-4, 1e-3],
    'dense_units': [512, 1024],
    'trainable_layers': [-4, -8, -12],
    'batch_size': [32, 64],
    'epochs': [20, 50]
}
```

Figure 13 *Parameters grid.*

Learning rate: This controls how much to change the model in response to the estimated error each time the model weights are updated. Lower values make the training slower but can lead to more accurate models, while higher values make the training faster but can potentially skip over optimal solutions.

Dense units: This refers to the number of units or neurons in the dense (fully connected) layers of your model. The choice of dense units affects the model's capacity to learn complex patterns.

Trainable layers: This parameter specifies which layers in the model are trainable. The values likely refer to layers indexed from the end of the model. For example, `-4` would mean the last four layers of the model are trainable.

Example values: `[-4, -8, -12]` indicate that the last 4, 8, or 12 layers of the model will be trained.

Batch size: The batch size defines the number of samples that will be propagated through the network at once. A smaller batch size allows for more updates per epoch but requires more memory, while a larger batch size is more stable but may generalize less well.

Example values: `[32, 64]` represent batch sizes of 32 and 64 samples.

Epochs: The number of epochs defines how many times the learning algorithm will work through the entire training dataset. More epochs can improve model accuracy but also increase the risk of overfitting.

Example values: `[20, 50]` represent training the model for 20 or 50 epochs.

Parameters selected:

Learning rate:0.0001, Dense units:512, Trainable layers: -8, Batch size:32, Epochs: 10

Model	Validation Accuracy	Validation Loss
Normal VGG16	96.8%	0.09
VGG16+ Grid Seach	97.1%	0.07

Table 1 *comparison table for vgg16 with vgg16+ grid search hyperparameter tuning*

Table 1 provides a comparative analysis of the VGG16 model with and without grid search hyperparameter tuning, highlighting the differences in performance metrics achieved through the optimization process.

4.2.4. Ensemble Models

By using resnet, mobilenet, xception, vgg16 I have made an ensemble model.

Stacking ensemble model is a logistic regression model which uses the predictions generated by the trained models on training, validation data.

4.2.4.1 Stacking Ensemble Model

Structure of Meta Classifier:

a. **Predictions of Base Models:** Generate predictions for the training and validation datasets using three pre-trained models:
   i. **MobileNet + ResNet50 + Xception**
   ii. **VGG16 + ResNet50 + Xception**
b. **Combine Predictions:** Concatenate the predictions from the three models for both training and validation sets to form combined prediction sets
c. **Train Meta Classifier:** Train a Logistic Regression meta-classifier using the combined predictions from the training set.

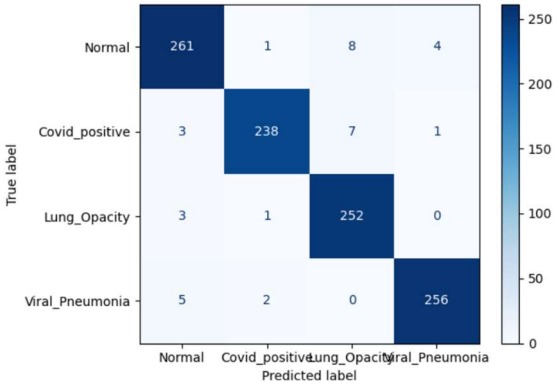

Figure 14 *confusion matrix for stacking ensemble (i) model for test set*

The confusion matrix for the stacking ensemble (i) model applied to the test set, showcasing the model's classification performance by depicting the true positive, true negative, false positive, and false negative predictions across different classes shown in **Figure 14**. This ensemble model has very less misclassification. overall better performing with slight decrease in validation loss compared to resnet50 model

	precision	recall	f1-score	support
Normal	0.99	0.96	0.98	103
Covid_positive	0.97	0.98	0.98	105
Lung_Opacity	0.96	0.97	0.97	115
Viral_Pneumonia	0.99	0.99	0.99	94
accuracy			0.98	417
macro avg	0.98	0.98	0.98	417
weighted avg	0.98	0.98	0.98	417

Figure 15 c*lassification report for ensemble model (ii).*

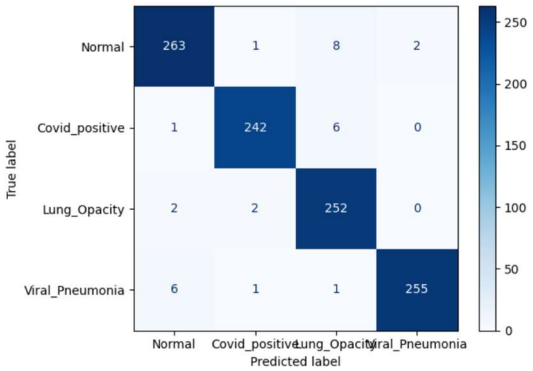

Figure 16 *confusion matrix for stacking ensemble (ii) model for validation set*

Classification report for the ensemble model (ii), summarizing key performance metrics such as precision, recall, and F1-score for each class, thereby providing a comprehensive evaluation of the model's effectiveness in classifying lung diseases shown in the **Figure 15**. The classification report provides a breakdown of the model's performance for each

class: Normal, Covid positive, Lung Opacity, and Viral Pneumonia. The model achieved a high overall accuracy of 98% shown in the **Figure 16**. The highest f1-score of 0.99 was observed for the Viral Pneumonia class.

In the Confusion matrix for the validation dataset there were very less misclassification and the viral pneumonia class has performed the best The second combination of the stacking ensemble model has performed better than the first combination. **Table 2** offers a comparative analysis of two different ensemble combinations, highlighting their respective performance metrics to evaluate which configuration yields better results in classifying lung diseases.

Ensemble Models	Test accuracy (%)
(i) MobileNet + ResNet50 + Xception	96.88
(ii) VGG16 + ResNet50 + Xception	**97.12**

Table 2 *comparison of two ensemble combinations*

4.2.4.2 Voting Ensemble Model

In voting ensemble all the base models are individually trained and each base model makes its own prediction. These predictions are then aggregated using a voting mechanism

Voting Mechanisms:

the final prediction is determined by the majority vote. The class label that receives the most votes from the base models is selected as the final prediction

4.2.4.3 Boosting Ensemble Model

Boosting focuses on correcting the errors made by previous models, aiming to build a strong predictive model incrementally.

1. Sequential Training: Boosting trains models sequentially, where each model attempts to correct the mistakes of the previous one.
2. Weighted Data Points: Initially, all data points are given equal weight. As models are trained, boosting adjusts the weights of misclassified data points, giving more importance to those that were previously misclassified. This means subsequent models focus more on correcting these errors.
3. Model Combination: After training all models, the final prediction is made by combining the predictions of all base models. The combination is usually a weighted sum of the predictions, where the weights are determined by the performance of each model.

Types of Boosting:

1. XGBoost (Extreme Gradient Boosting)
2. LightGBM (Light Gradient Boosting

Ensemble technique	Validation Accuracy (%)
Stacking Ensemble (Meta Model)	**97.60**
Voting Model	97.36
XGBoost	96.88
LightGBM	**97.60**

Table 3 *comparison of different ensemble techniques*

Table 3 presents a comparison of various ensemble techniques, detailing their performance metrics to assess the effectiveness of each method in classifying lung diseases.

Dataset 2 Training Results

Model	Test Accuracy (%)
VGG16	96.15
ResNet50	91.20
Ensemble Model (VGG16 + ResNet50)	96.88
Ensemble Model + GSO	**97.06**

Table 4 *Comparison of different models*

This dataset is used in the paper Tanzina Taher Ifty , et.al [2]. Their model achieved an accuracy of 90% on one fold, 95% on another fold, 98% on another fold, 99% on another fold, and 99% on the fifth fold. The average accuracy across all five folds is 96.20%. the comparision of the proposed model with the existing paper is shown in **Table 5**.

Paper	Model	Test Accuracy (%)
Tanzina Taher [2]	Xception+ 5 fold	96.20
Proposed Ensemble Model	VGG16 + ResNet50	97.06

Table 5 C*omparison of proposed model with existing paper for dataset-2*

Model	Train Accuracy	Test Accuracy	Validation Accuracy
ResNet50	**0.99**	**0.96**	**0.971**
VGG19	0.91	0.87	0.83
VGG16	**0.98**	**0.96**	**0.97**
Xception	**0.98**	**0.96**	**0.97**
EfficientNetB0	0.98	0.89	0.85
MobileNetV2	0.99	0.90	0.93
DenseNet121	0.91	0.87	0.85
Stacking Ensemble	**---**	**0.971**	**0.976**

Table 6 *Comparison of all the implemented models for dataset-1*

Among the individual models in **Table 6**, ResNet50 and VGG16 achieved high test and validation accuracies of 0.96/0.97 and 0.96/0.97 respectively. However, the Stacking Ensemble model, which combines the predictive strengths of these individual models, outperformed all, achieving a test accuracy of 0.971 and a validation accuracy of 0.976 in **Table 6**. This result demonstrates the effectiveness of the ensemble approach in leveraging multiple models to enhance overall prediction accuracy and robustness.

Author	Models Used	Dataset	Accuracy
K. Vij [1]	VGG19+ResNet50	Chest X-rays	97.60%
Tanzina Taher Ifty [2]	InceptionV3+Xception	COVID-19 Detection X-Ray Dataset	96.21%
Apostolopoulos [3]	VGG19	Github Repository	98.75%
P. Mahajan [4]	InceptionV3	Covid-19 Radiography Dataset	91.06%
A. Gokarn [5]	VGG19	Covid-19 Radiography Dataset	95.45%
A. C. Mazari [6]	MobileNetV2	Covid-19 Radiography Dataset	97.32%
Goram Mufarah [7]	VGG19 + CNN	Multiple Public Datasets	96.48%
Proposed Ensemble Model	**VGG16 + ResNet50 + Xception**	**Covid-19 Radiography Dataset**	**97.60%**

Table 7 *Comparison of proposed model with existing literature for datset-1*

Our Proposed Ensemble Model of VGG16 + ResNet50 + Xception has performed very well with a validation accuracy of 97.60% on comparison with existing literature in **Table 7**. And the literature papers P. Mahajan [4], A. Gokarn [5], A. C. Mazari [6] with the same dataset as ours (Covid-19 Radiography Dataset) when compared our model performs better than them.

Comparison of stacking ensemble model with base models confusion matrix:

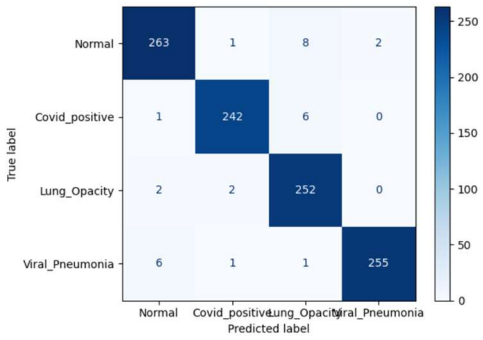

Figure 17 *Stacking Ensemble model confusion matrix*

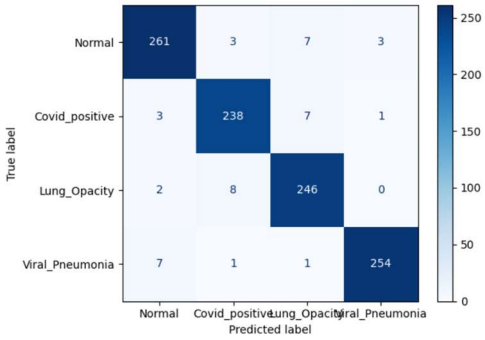

Figure 18 *Resnet50 model confusion matrix*

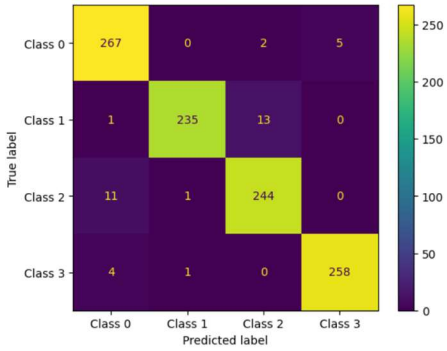

Figure 19 *Xception model confusion matrix*

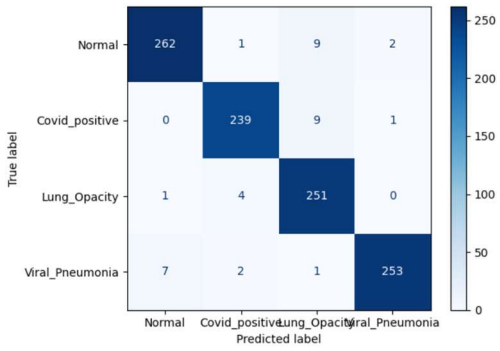

Figure 20 *VGG16 model confusion matrix*

The ensemble model in **Figure 17** provides the best overall classification performance, with fewer misclassifications compared to the ResNet in **Figure 18** and Xception model in **Figure 19**, and VGG16 model in **Figure 20**. The ResNet model performs well but shows more misclassifications than the ensemble, particularly in the Covid_positive and Lung_Opacity classes. The Xception model shows more significant misclassifications,

particularly in the Covid_positive and Lung_Opacity classes, indicating it might be less effective than the ensemble and ResNet models for this task.

The ensemble model's superior performance demonstrates the effectiveness of combining predictions from multiple models to improve classification accuracy in the lung disease classification task.

5. CONCLUSION AND FUTURE WORKS

In this study, we have effectively leveraged deep learning techniques to tackle the challenge of classifying lung diseases using chest X-ray images. The approach involved training and fine-tuning multiple cutting-edge Convolutional Neural Network (CNN) models, which are known for their strong performance in image classification tasks. Specifically, we employed several architectures, including ResNet50, Xception, MobileNetV2, VGG16, VGG19, InceptionV3, and DenseNet121. Each of these models was trained on the chest X-ray dataset, and fine-tuning was performed to enhance their performance on the specific task of lung disease classification. Fine-tuning the models to better generalize and improve their classification accuracy.

Recognizing that individual models have strengths and weaknesses depending on the features of the data they encounter, we explored ensemble learning techniques to further boost performance. Ensemble learning, which involves combining multiple models to make predictions, can often lead to better results than any single model can achieve on its own. Among the ensemble strategies we evaluated, we developed a stacked ensemble model. This model specifically combined the predictions of ResNet50, Xception, and VGG16, three of the top-performing models in our experiments.

The stacking ensemble model achieved an impressive accuracy of 97.60% in classifying lung diseases from the chest X-ray images, a result that reflects the effectiveness of this ensemble approach. Our findings suggest that this ensemble model not only achieved competitive

results but also matched or exceeded the performance of some of the best results reported in existing literature on similar tasks. This indicates that ensemble methods, particularly those that combine the predictive power of multiple CNN architectures, hold great promise for advancing the accuracy and reliability of medical image classification. The high accuracy achieved in this study underscores the potential of using deep learning and ensemble techniques in critical healthcare applications, such as the early detection and diagnosis of lung diseases.

REFERENCES

[1] K. Vij, S. Thakur, A. Sharma and R. Mohana, "Lung Disease Classification using X-Ray Imaging with Ensemble Learning," *2023*

[2] Tanzina Taher Ifty , et.al, “Explainable Lung Disease Classification from Chest X-Ray Images Utilizing Deep Learning and XAI”, 2024.

[3] Apostolopoulos, I.D., Mpesiana, T.A. “Covid-19: automatic detection from X-ray images utilizing transfer learning with convolutional neural networks”2023.

[4] P. Mahajan, N. Karlupia, P. Abrol and P. K. Lehana, "Identifying COVID-19 Pneumonia using Chest Radiography using Deep Convolutional Neural Networks," *2021.*

[5] A. Gokarn, K. Patni, Y. Purohit and R. Sonkusare, "COVID-19 Radiography Using ConvNets," *2022*.

[6] A. C. Mazari and H. Kheddar, "Deep Learning-and Transfer Learning-based Models for COVID-19 Detection using Radiography Images," *2023.*

[7] Goram Mufarah M. Alshmrani , et.al “A deep learning architecture for multi-class lung diseases classification using chest X-ray (CXR) images” 2023.

[8] Hong, Min, et al. ”Multi-Class Classification of Lung Diseases Using CNN Models.” (2021): 9289. DOI: [10.3390/app11199289](https://doi.org/10.3390/app11199289).

[9] Alshmrani, Goram Mufarah M., et al. ”A deep learning architecture for multi-class lung diseases classification using chest X-ray (CXR) images.” *Alexandria Engineering Journal* 64 (2023): 923-935.

[10] Lyu, Juan, and Sai Ho Ling. ”Using multi-level convolutional neural network for classification of lung nodules on CT images.” *2018 40th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*. IEEE, 2018.
[11] Qian, Xuelin, et al. ”Lung-Sys: A deep learning system for multi-class lung pneumonia screening from CT imaging.” *IEEE journal of biomedical and health informatics* 24.12 (2020): 3539-3550.