Explainable Lung Disease Classification from Chest X-Ray Images Utilizing Deep Learning and XAI

A thesis Report
Submitted in partial fulfillment of the requirements for the Degree of
Bachelor of Science in Computer Science and Engineering

Submitted by

Tanzina Taher Ifty	**190104053**
Tashfia Towhid	**190104085**
Md. Shahid Hasan	**190104130**
Saleh Ahmed Shafin	**190104138**

Supervised by

Mr. Shoeb Mohammad Shahriar

Department of Computer Science and Engineering
Ahsanullah University of Science and Technology

Dhaka, Bangladesh

November 2023

CANDIDATE'S DECLARATION

We, hereby, declare that the thesis presented in this report is the outcome of the investigation performed by us under the supervision of Mr. Shoeb Mohammad Shahriar, Department of Computer Science and Engineering, Ahsanullah University of Science and Technology, Dhaka, Bangladesh. The work was spread over two final year courses, CSE4100: Project and Thesis I and CSE4250: Project and Thesis II, in accordance with the course curriculum of the Department for the Bachelor of Science in Computer Science and Engineering program.

It is also declared that neither this thesis nor any part thereof has been submitted anywhere else for the award of any degree, diploma or other qualifications.

Tanzina Taher Ifty
190104053

Tashfia Towhid
190104085

Md. Shahid Hasan
190104130

Saleh Ahmed Shafin
190104138

CERTIFICATION

This thesis titled, **"Explainable Lung Disease Classification from Chest X-Ray Images Utilizing Deep Learning and XAI"**, submitted by the group as mentioned below has been accepted as satisfactory in partial fulfillment of the requirements for the degree B.Sc. in Computer Science and Engineering in November 2023.

Group Members:

Tanzina Taher Ifty	**190104053**
Tashfia Towhid	**190104085**
Md. Shahid Hasan	**190104130**
Saleh Ahmed Shafin	**190104138**

Mr. Shoeb Mohammad Shahriar
Assistant Professor & Supervisor
Department of Computer Science and Engineering
Ahsanullah University of Science and Technology

Dr. Md. Shahriar Mahbub
Professor & Head
Department of Computer Science and Engineering
Ahsanullah University of Science and Technology

ACKNOWLEDGEMENT

This paper and the research behind it would not have been possible without the exceptional support of my supervisor Mr. Shoeb Mohammad Shahriar, Assistant Professor, Department of Computer Science and Engineering, Ahsanullah University of Science and Technology.

We would like to express our gratitude to Professor Dr. Md. Shahriar Mahbub,Head of the Department of Computer Science and Engineering of Ahsanullah University of Science and Technology. We would also like to thank the librarians, research assistants, and students from the university, who impacted and inspired us. Finally, we would like to thank our parents for their encouragement and support throughout our education.

Dhaka
November 2023

Tanzina Taher Ifty
Tashfia Towhid
Md. Shahid Hasan
Saleh Ahmed Shafin

ABSTRACT

Lung diseases remain a critical global health concern, and it's crucial to have accurate and quick ways to diagnose them. This thesis focuses on classifying different lung diseases into five groups: COVID, viral pneumonia, bacterial pneumonia, tuberculosis, and normal lungs. Employing advanced deep learning techniques, we explore a diverse range of models including CNN, hybrid models, ensembles, transformers, and Big Transfer. The research encompasses comprehensive methodologies such as hyperparameter tuning, stratified k-fold cross-validation, and transfer learning with fine-tuning. Additionally, our study integrates various image enhancement techniques, including Histogram Equalization (HE), Contrast Limited Adaptive Histogram Equalization (CLAHE), CLAHE with Gamma correction, Canny Edge detection, and Complement transformation. These enhancements aim to improve the quality of X-ray images and, consequently, the accuracy of disease classification. Remarkably, our findings reveal that the Xception model, fine-tuned through 5-fold cross-validation, achieves the highest accuracy of 96.51%. This success shows that our methods work well in accurately identifying different lung diseases. It highlights how using advanced deep learning techniques and improving the quality of the lung images can be really helpful in medical diagnosis. The exploration of explainable artificial intelligence (XAI) methodologies further enhances our understanding of the decision-making processes employed by these models, contributing to increased trust in their clinical applications.

key words- lung disease detection, Digital X-ray Images, Image Classification, Explainable AI, Deep learning, Image enhancement techniques, CNN models.

Contents

CANDIDATE'S DECLARATION i

CERTIFICATION ii

ACKNOWLEDGEMENT iii

ABSTRACT iv

List of Figures ix

List of Tables x

1 Introduction 1

1.1 Overview . . . 1
1.2 Motivation . . . 2
1.3 Objective . . . 3
1.4 Challenges and ways to address these challenges . . . 4
1.5 Thesis Contribution . . . 5
1.6 Thesis Structure . . . 5
1.7 Summary . . . 6

2 Background Studies 7

2.1 Overview . . . 7
2.2 Convolutional neural network (CNN) . . . 8
2.2.1 Image Input Layer . . . 8
2.2.2 Convolutional layer . . . 8
2.2.3 Stack Normalization Layer (Batch) . . . 9
2.2.4 Activation Layer for ReLU . . . 9
2.2.5 Layer of Pooling . . . 10
2.2.6 Softmax Activation Function . . . 11
2.2.7 FC Layer . . . 11
2.3 Hybrid Neural Network . . . 12
2.4 Ensemble learning . . . 13
2.5 Vision Transformers and Big Transfer . . . 13

2.6 K-fold cross-validation 14
2.7 Explainable AI 15
2.7.1 Local Interpretable Model-agnostic Explanations (LIME) 15
2.7.2 Gradient Class Activation Map (Grad-CAM) 16
2.8 Underfitting, Optimal fit and Overfitting 17
2.9 Evaluation Matrix 18
2.10 Summary 19

3 Literature Review **20**
3.1 Overview 20
3.2 Deep Learning Approaches 20
3.2.1 "TX-CNN: Detecting tuberculosis in chest X-ray images using convolutional neural network" 20
3.2.2 "A modified deep convolutional neural network for detecting COVID-19 and pneumonia from chest X-ray images based on the concatenation of xception and ResNet50V2" 21
3.2.3 "COVID-19 Diagnosis from Chest X-ray Images Using Deep Learning Approach" 21
3.2.4 "Coronet: A deep neural network for detection and diagnosis of COVID- 19 from chest x-ray images" 22
3.2.5 "Attention-based VGG-16 model for COVID-19 chest X-ray image clas- sification" 23
3.2.6 "CheXImageNet: a novel architecture for accurate classification of Covid-19 with chest x-ray digital images using deep convolutional neural networks" 24
3.2.7 "An efficient mixture of deep and machine learning models for COVID- 19 and tuberculosis detection using X-ray images in resource-limited settings" 24
3.2.8 An explainable artificial intelligence model for identifying local indicators and detecting lung disease from chest X-ray images 25
3.2.9 Explanatory classification of CXR images into COVID-19, Pneumonia and Tuberculosis using deep learning and XAI 26
3.2.10 Multi-Scale CNN: An Explainable AI-Integrated Unique Deep Learning Framework for Lung-Affected Disease Classification 26
3.3 Summary 27

4 Dataset Analysis **29**
4.1 Exploratory Data Analysis (EDA) 30
4.1.1 Visualize Training samples 30
4.1.2 Class Distribution 31

4.1.3 Sample Image Size Distribution . . . 31
4.1.4 Viral Pneumonia Class Pixel Distribution . . . 32
4.1.5 Sample Image Histogram . . . 32
4.1.6 Some Bad Training Images . . . 33
4.2 Summary . . . 33

5 Proposed Methodology **34**
5.1 Overview . . . 34
5.2 Data Collection . . . 34
5.3 Data Preprocessing . . . 34
5.3.1 Image-Processing . . . 35
5.3.2 Image Augmentation . . . 37
5.3.3 Rescaling . . . 37
5.3.4 Shifting . . . 37
5.3.5 Zooming . . . 37
5.3.6 Flipping . . . 37
5.3.7 Rotation . . . 38
5.3.8 Shearing . . . 38
5.3.9 Brightness . . . 38
5.4 Deep Learning Methodology . . . 38
5.4.1 Pre-trained single CNN Model . . . 39
5.4.2 Hyper-parameter Tuning . . . 39
5.4.3 Transfer Learning and Fine Tune . . . 40
5.4.4 Hybrid Neural Network . . . 41
5.4.5 Ensemble Learning . . . 42
5.4.6 Stratified K-fold Cross Validation . . . 43
5.4.7 Transformer and Big Transfer Learning . . . 43
5.5 Explainable Artificial Intelligent (XAI) . . . 44
5.5.1 LIME . . . 44
5.5.2 Grad-CAM (XAI) . . . 44
5.6 Project Management . . . 45
5.6.1 Project Schedule . . . 45
5.6.2 Financial Management . . . 46
5.7 Summary of the Methodology: . . . 47

6 Result Analysis **48**
6.1 Overview . . . 48
6.2 Experiment Results . . . 48
6.2.1 Result by Pre-Trained CNN Model . . . 49
6.2.2 Hyper-parameter Tuning . . . 50

6.2.3 Transfer Learning and Fine Tune 51
6.2.4 Result by Hybrid Model 51
6.2.5 Result by Vision Transformer and Big transfer Model 52
6.2.6 Stratified K-fold Cross Validation 52
6.2.7 Result by Ensemble Model 54
6.3 Interpretability 56
6.3.1 Lime 56
6.3.2 Grad-CAM 56
6.4 Summary 57

7 Conclusion **58**
7.1 Conclusion 58
7.2 Future Work 59

References **60**

List of Figures

1.1 Incidence of pneumonia admissions among children <5 years, Uganda, 2013-2021 [1]. 2
1.2 Hospitalization trends (Average Length of Stay in days and % of admitted out of those treated) for newly diagnosed TB patients in Europe [2]. 3

2.1 Convo layer visualization [3]. 9
2.2 The ReLU function, represented graphically and expressed in equation [4]. . 10
2.3 Visualization of Pooling layer. 10
2.4 Visualization of Fully Connected layer [5] 11
2.5 Structure of Hybrid CNN Models 12
2.6 Structure of Ensemble Learning 13
2.7 Structure of Stratified K-fold Cross-Validation 14
2.8 Visualization of LIME Algorithms. [6] 16
2.9 Visualization of underfitting, optimal and overfitting [7]. 17
2.10 2X2 Confusion Matrix [8]. 18

4.1 Dataset orientation. 29
4.2 Sample Images Per Disease 30
4.3 Class distribution of the training dataset 31
4.4 Size distribution of the images in training dataset 31
4.5 Pixel intensity distribution for a training sample image 32
4.6 Sample Images Histogram visualization 32
4.7 Bad training samples 33

5.1 Shallow Network Vs Deep Neural Network [9] 39
5.2 Gantt Chart of Project Management 45
5.3 This methodology for 4 types of lung disease and normal lung identification: The whole procedure is represented in multiple steps. 47

6.1 Confusion Matrix of Xception Model 50
6.2 Important Regions that help Inception-V3 Model prediction 56
6.3 Wrongly Classified Image by Xception Model prediction 57
6.4 Correctly Classified Image by Xception Model prediction 57

List of Tables

3.1 **Summary of the Review Paper** 27
3.2 **Summary of the Review Paper** 28

5.1 Hyperparameter Tuning with Grid Search 40
5.2 Hybrid Model Combinations 42
5.3 Ensemble Model Combinations and Prediction Methods 43

6.1 Different Enhancement Techniques 48
6.2 Result obtained by single model 49
6.3 Result obtained by Transfer Learning and Fine Tune 51
6.4 Result obtained by hybrid model 52
6.5 Result obtained by transformer and big transfer model 52
6.6 Result obtained by 3-fold 53
6.7 Result obtained by 5-fold 54
6.8 Result obtained by ensemble model using average prediction criteria 55
6.9 Result obtained by ensemble model using majority voting criteria 55

Chapter 1

Introduction

1.1 Overview

Lung diseases are a major health concern worldwide and affect a large number of people. For successful treatment and better patient outcomes, lung illnesses must be identified early and correctly diagnosed. Chest X-ray (CXR)-based method is one of the cheapest alternative options to diagnose the early stage of lung disease compared to other alternatives such as Polymerase Chain Reaction (PCR), Computed Tomography (CT) scan, and so on [10].

The automated classification of lung illnesses from chest X-ray pictures has shown encouraging results because of recent developments in deep learning and artificial intelligence (AI) [11]. In deep learning models, their decision-making procedures are difficult to understand for humans. In this case, a method of machine learning known as explainable AI (XAI) tries to give human-understandable justifications for a model's judgments [6].

The classification of lung disease from chest X-ray pictures may be more accurate and understandable with the use of deep learning and XAI algorithms [12]. The regions of the chest X-ray pictures that the model has used to establish a diagnosis can be understood using XAI approaches in the context of classifying lung diseases [4]. This can boost physicians' trust in the diagnosis by assisting them in comprehending the logic underlying the model's choice.

In this thesis, we develop a CNN-based classification model for the diagnosis of four different types of lung diseases - viral pneumonia, bacterial pneumonia, tuberculosis, and COVID-19 as well as normal lungs. We use pre-trained CNN models, hybrid models, ensemble models, transformer models, K-fold technique, different enhancement technique, hyper parameter tuning to train and test our model [13]. The performance of the models will be evaluated using standard evaluation metrics such as accuracy, precision, recall, and F1-score. We also use Explainable AI (XAI) techniques such as Local Interpretable Model-Agnostic Explanations (LIME) and Grad-CAM to provide visual explanations for the model's predictions.

These techniques will enable us to identify the important features and regions of the chest X-ray images that contribute to the classification decisions, which can be useful for clinicians in the diagnosis and treatment of lung diseases.

1.2 Motivation

Covid-19 which was declared a pandemic by the World Health Organization (WHO), has been a devastating experience for all of the people in the world [14]. The pandemic has highlighted how urgently sophisticated diagnostic instruments are needed, not only for COVID-19 but also for a variety of respiratory conditions that continue to afflict millions of people globally.

Pneumonia claimed the lives of 2.5 million people in 2019. It is the primary cause of death for children under five, with over one-third of all casualties being younger than five [15].

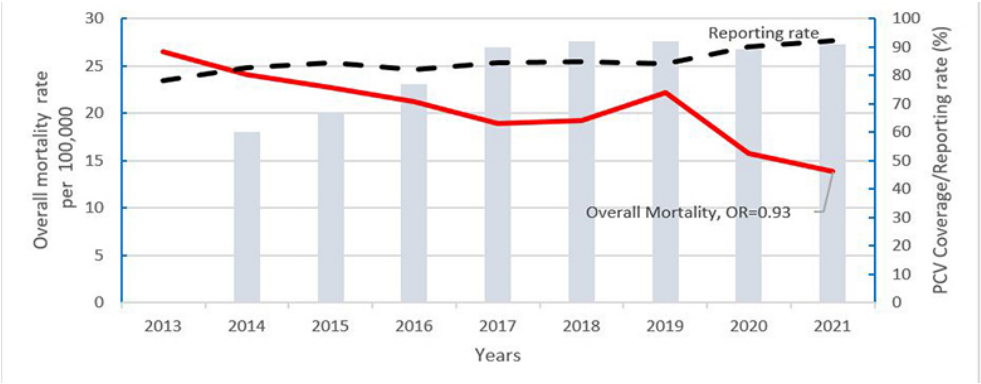


Figure 1.1: Incidence of pneumonia admissions among children <5 years, Uganda, 2013-2021 [1].

In 2021, 1.6 million people—including 187 000 HIV-positive individuals—passed away from TB. Globally, tuberculosis ranks 13th in terms of cause of death and ranks second in terms of infectious mortality (below HIV and AIDS) after COVID-19. Globally, 10.6 million cases of tuberculosis (TB) were reported in 2021. There are 1.2 million children, 3.4 million women, and six million males. Every age group and nation has some form of tuberculosis [16].

Lung diseases are a major health concern in Bangladesh. In our country, there is a shortage of quality healthcare and trained medical professionals, especially in remote areas. This makes it difficult to accurately diagnose and treat lung diseases. Many people with lung diseases do not receive timely treatment. But early diagnosis and treatment of lung diseases is very important [10].

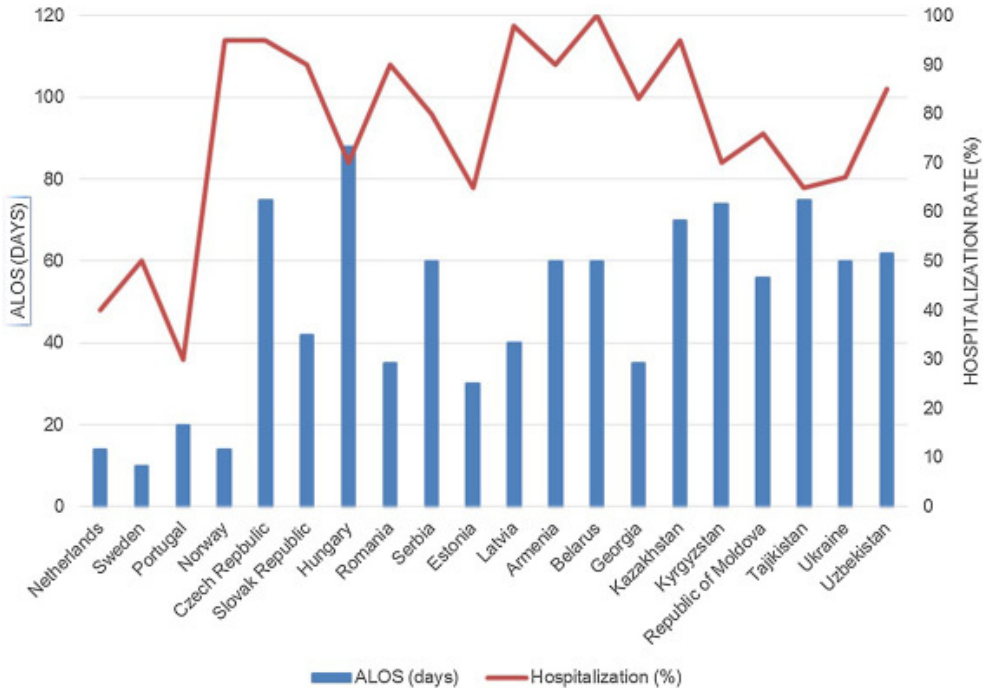


Figure 1.2: Hospitalization trends (Average Length of Stay in days and % of admitted out of those treated) for newly diagnosed TB patients in Europe [2].

1.3 Objective

Deep learning methods, such as convolutional neural networks (CNNs), have been shown to be effective in classifying lung diseases from medical images. So, our main objective is to classify lung diseases using deep learning methods.

Also, this could focus on developing a system that can help diagnose lung diseases in low-resource settings, such as rural areas or developing countries, where access to healthcare may be limited [17]. Developing a cost-effective and portable system can help improve access to healthcare and reduce the burden of lung diseases in these settings. CNNs can learn to identify subtle patterns in images that are not visible to the human eye [18]. So we propose Explainable artificial intelligence (XAI) methods that can be used to explain how CNNs make decisions. This can help clinicians to better understand the results of CNN-based classifications and to make more informed treatment decisions [19].

Therefore, CNNs and XAI have the potential to significantly improve the diagnosis and treatment of lung diseases in Bangladesh. By providing early diagnosis and treatment, these technologies can help to save lives and improve the quality of life for people with lung diseases. In order to ultimately enhance patient health outcomes, this thesis paper seeks to build a more efficient and open method for diagnosing lung diseases. Lastly, the ultimate goal is to develop a comprehensible AI system for categorizing lung disorders that lead to better patient outcomes.

1.4 Challenges and ways to address these challenges

Deep learning models have the potential to revolutionize the diagnosis of lung diseases. However, there are a number of challenges that need to be addressed before these models can be used in a clinical setting.

One of the biggest challenges is data scarcity. There is a limited amount of labeled data available for training deep learning models for lung disease classification. This is due to the fact that collecting and labeling chest X-ray images is a time-consuming and expensive process. Another challenge is data imbalance. The distribution of classes in the available data is often imbalanced, with a small number of positive cases and a large number of negative cases. This can make it difficult for deep learning models to learn to identify the positive cases. Deep learning models are also often complex and difficult to interpret. This can make it difficult to understand how the model is making its decisions and to identify potential sources of bias [20]. Finally, deep learning models are often not able to explain their decisions in a way that is understandable to humans. This can make it difficult to trust the decisions of the model and to use the model in a clinical setting. Despite these challenges, there is a lot of potential for deep learning models to improve the diagnosis of lung diseases. By addressing these challenges, we can develop deep learning models that are more accurate, reliable, and explainable. This will help us to improve the early detection and treatment of lung diseases, saving lives and improving the quality of life for patients.

Here are some of the ways to address these challenges [19]:

Data augmentation: Data augmentation is a technique that can be used to increase the amount of labeled data available for training deep learning models [21]. This can be done by creating new data points from existing data points by applying transformations, such as cropping, flipping, and rotating.

Cost-sensitive learning: Cost-sensitive learning is a technique that can be used to address the problem of data imbalance. This technique assigns different costs to different classes, so that the model is more likely to learn to identify the positive cases [22].

Explainable AI (XAI): XAI is a field of research that focuses on developing techniques to make deep learning models more explainable. This can be done by developing methods to visualize the decisions of the model or to explain the model's predictions in a way that is understandable to humans [23].

By addressing these challenges, we can develop deep learning models that are more accurate, reliable, and explainable. This will help us to improve the early detection and treatment of lung diseases, saving lives and improving the quality of life for patients.

1.5 Thesis Contribution

We conducted a review of several related research studies to comprehend the current methodologies in lung disease detection. Each of these studies differs in terms of methodology, dataset, disease classification categories, and the incorporation of explainable artificial intelligence (XAI). Our work encompasses a comprehensive set of popular approaches, leveraging XAI. Our models demonstrate the capability to classify four diseases in addition to normal lung conditions. Furthermore, we introduced unique enhancements to X-ray images, distinguishing our work from existing research. The utilization of the 5-fold cross-validation technique [24] led us to achieve our highest accuracy.

1.6 Thesis Structure

- **Chapter 1 Introduction:** This chapter provides a brief overview of our proposed work, the rationale behind selecting the subject, the difficulties we encountered and how we overcame them, and the goal of our project.
- **Chapter 2 Background Studies:** In this chapter, we gave a short summary of the topics that we studied and learned to apply to our thesis. This knowledge is the root of this project.
- **Chapter 3 Literature Review:** We reviewed some related papers and combined them in a tabular format to understand the performance and limitations of those papers. This chapter gives clear knowledge of the topic and a better understanding of how to improve our performance.
- **Chapter 4 Dataset Analysis:** This chapter contains information about the dataset we used. Some preprocessing techniques have been adapted to enhance the image quality for better accuracy.
- **Chapter 5 Proposed Methodology:** Details about the steps that we took to achieve the goal of lung disease classification have been discussed in this chapter. We took several approaches, like CNN models, hybrid models, ensemble models, transfer learning, and XAI, to reach our goal.
- **Chapter 6 Result Analysis:** This chapter has a representation of the results that we achieve after executing different approaches. We made comparisons among the models to find the best approach. The interpretability section discussed explainable artificial intelligence.

- **Chapter 7 Conclusion:** The conclusion of our work, its limitations, and potential future works are covered in the last chapter.

1.7 Summary

Establishing the critical context, the introduction defines the problem, research questions, motivations, challenges, structure of the thesis and our contribution, objectives, and significance. It introduces our project to the reader.

Chapter 2

Background Studies

2.1 Overview

This investigation delves into the domain of lung disease detection, offering a thorough examination of key methodologies. The focal point revolves around the CNN, a potent tool for images analysis. The CNN architecture is deconstructed into distinct layers, each serving a specific purpose—from the Image Input Layer to the Fully Connected Layers. Together, these layers contribute to the network's capacity to identify essential patterns and features crucial for precise disease detection [25].

Expanding beyond CNN, the exploration encompasses advanced techniques like the Hybrid Neural Network, Ensemble Learning, and Vision Transformers, presenting a diverse array of approaches to address the intricacies of lung disease detection. K-fold cross-validation emerges as a crucial strategy, ensuring the reliability and adaptability of the models.

The study also introduces Explainable AI techniques, offering insights into the opaque nature of neural networks. Two methods for interpreting and understanding the choices made by the models are discussed: Gradient-Class-Activation-Map (Grad-CAM) [26] and Local-Interpretable-Model-agnostic-Explanations (LIME) [27].

Concepts such as underfitting, optimal fit, and overfitting are clarified, imparting a nuanced comprehension of model performance. The significance of a robust Evaluation Matrix is underscored, acting as a vital tool to evaluate the efficacy of the developed models.

Lastly, This paper offers a thorough investigation of several neural network topologies, ensemble methods, interpretability techniques, and evaluation metrics, establishing a thorough groundwork for the development and comprehension of advanced lung disease detection models.

2.2 Convolutional neural network (CNN)

CNNs are advanced AI systems built upon multi-layer neural networks, capable of performing tasks such as object identification, recognition, classification, and image object detection and segmentation [28]. CNNs, also known as ConvNets, are particularly favored in the field of deep learning due to their ability to learn directly from input data without the need for manual feature extraction by humans. This characteristic sets CNNs apart from traditional machine learning algorithms, as they possess the capability to automatically learn and extract meaningful features from the input objects, leading to improved performance and efficiency in various computer vision tasks [28–30].
Convolutional neural networks (CNNs) are extensively employed in a wide range of applications, including visual recognition, medical image analysis, image segmentation, and natural language processing (NLP). CNNs outperform regular neural networks because they possess the ability to automatically recognize essential components from the input data, eliminating the need for human intervention [28–30]. This makes CNNs highly efficient and powerful in tasks that involve analyzing visual information, as they can automatically learn and extract relevant features from the input data, leading to improved performance and accuracy. A typical convolutional neural network (CNN) is composed of four main types of layers [31]:

1. Convolutional
2. Pooling
3. Function of Activation
4. Fully Connected

2.2.1 Image Input Layer

Pixels serve as the fundamental components of a digital image, representing the basic building blocks of visual data. They are binary units that make up the image's representation. In a digital image, pixels are arranged in a matrix-like structure, ordered sequentially from 0 to 255. Each pixel's position within the matrix determines its location in the image [32]. The brightness and hue characteristics of a pixel are determined by its assigned pixel value, which specifies the intensity or color information associated with that particular pixel.

2.2.2 Convolutional layer

Filters are used in this layer to extract image features. This layer can be used to apply different sizes and quantities of filters as needed [33]. Convolution kernels are another name for

these filters. The convolution operation works between the filters and the input images [33]. It produces a map of features. The convolution operation is [34]

$$\text{Conv}\ (m, n) = I\ (x, y) \otimes F\ (x, y)$$
$$\text{where,}\ \otimes = \text{convolution operation, I}\ (x, y) = \text{matrix of input image,}$$
$$F\ (x, y) = \text{function of filter.}$$

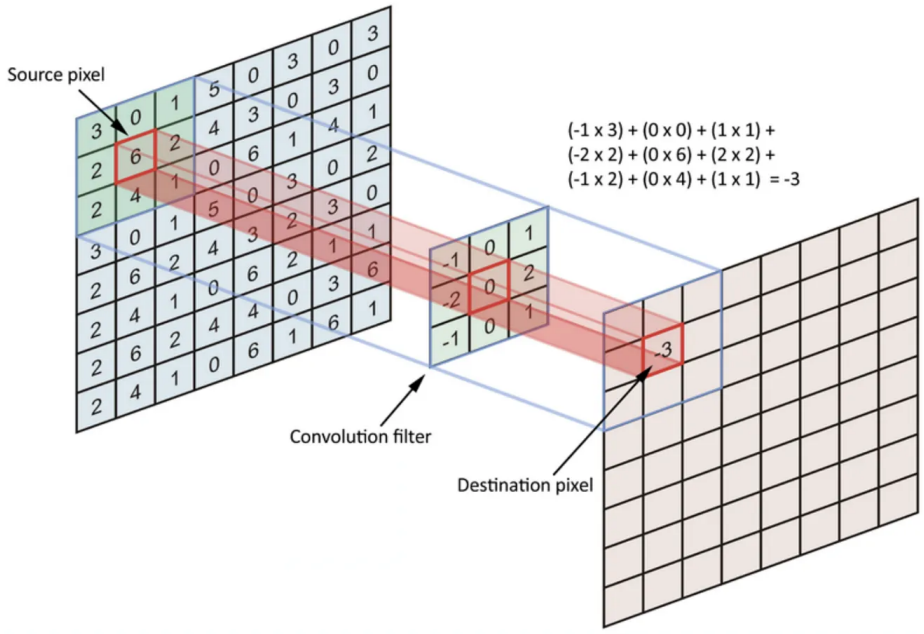


Figure 2.1: Convo layer visualization [3].

The convolution procedure at this layer is seen in Fig.Figure 2.1 's depiction of the path taken by the filter or kernel through the input picture. An integer value is generated by this procedure, which is repeated to process the entire image. The next layer receives this result.

2.2.3 Stack Normalization Layer (Batch)

A batch normalization layer has been used with the aim of adjusting the scale of the input layer. This layer expedites the process of training. It aids in bringing stability to the network. Comparably, it resembles the dropout layer, which is put before to the activation layer [35].

2.2.4 Activation Layer for ReLU

Deep learning uses a variety of activation function types. Most of the network employs the ReLU layer since the sigmoid activation function is in charge of ignoring the picture information. Rectified Linear Unit, or ReLU, is a quicker and easier-to-use non-linear activation

function. [36], The ReLU function is represented by the equation below Fig. Figure 2.2

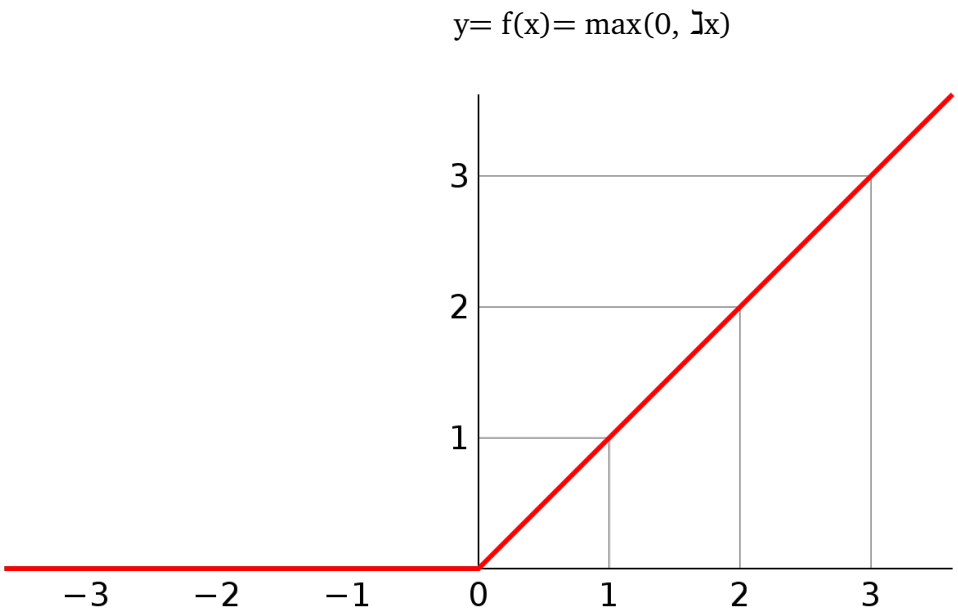


Figure 2.2: The ReLU function, represented graphically and expressed in equation [4].

2.2.5 Layer of Pooling

The pooling layer expedites network processing by reducing the data amount of the preceding network layer [32]. It is used to lower the dimensionality inside two twisted layers [31]. Typically, the network is expanded using two forms of pooling: average and max-pooling. The max-pooling layer reduces the size by taking the highest value from the convoluted output. Down-sampling is another term for this procedure [4]. It is shown in the next Fig. Figure 2.3. The stride size and filter size in Fig. Figure 2.3 are both maintained at 2×2, with the highest value from the sub-region being taken into account [37].

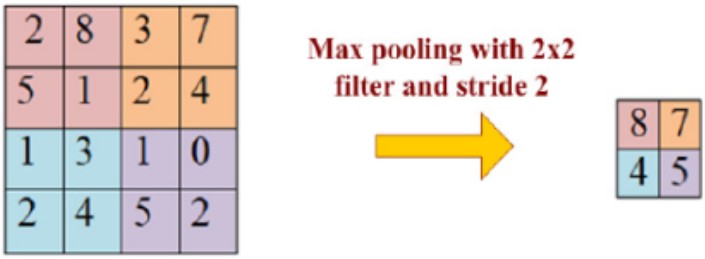


Figure 2.3: Visualization of Pooling layer.

2.2.6 Softmax Activation Function

For a multi-class classifier, Softmax is a rational method [37]. This function generates the last layer's probabilistic output and is mostly used in the classification final layers. The mathematical expression is [38].

$$\text{Softmax}(x_i) = \frac{\exp(x_i)}{\sum_j \exp(x_j)}$$

2.2.7 FC Layer

This layer represents the configuration of a feed-forward or standard neural network [37]. Every node in this layer has a direct connection to the layers before and after it. The feature from the layer above is leaned by a completely linked layer. There may be one or more completely linked levels. The output layer, which is the last FC layer and makes the required class predictions, is a popular term for it. This component is part of the CNN classification system, which operates subsequent to feature extraction. When compared to previous methods for handwritten English digit identification using various datasets, the suggested CNN architecture is a little bit unique and distinct. Compared to previous models, the suggested CNN model is more effective, straightforward, and time-efficient. Two datasets were utilized to demonstrate the efficacy. The increased accuracy demonstrates the efficacy of the suggested method's automatic CNN classifier.

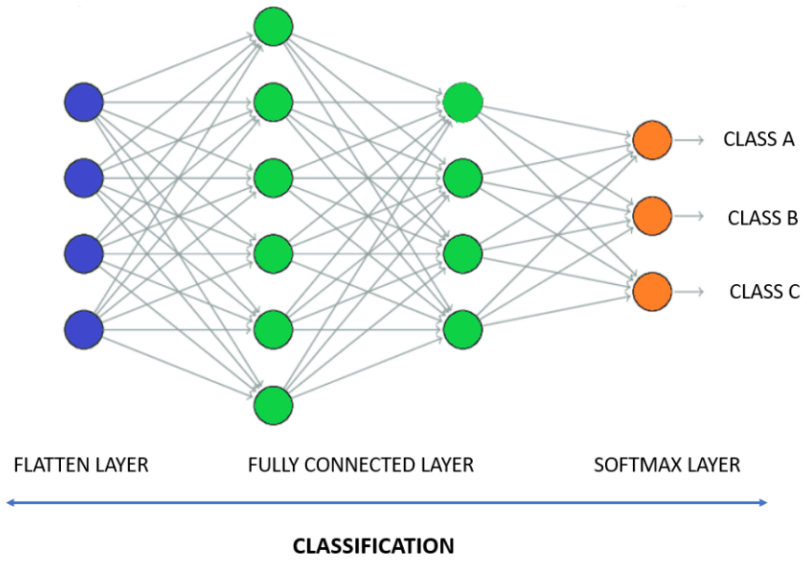


Figure 2.4: Visualization of Fully Connected layer [5]

.

2.3 Hybrid Neural Network

Recently, there's been a big change in how we classify images, thanks to fancy techniques like deep learning, especially using something called Convolutional-Neural-Networks (CNNs). These networks are really good at learning important stuff from pictures, helping us recognize things in images super accurately. As we want even fancier models, scientists are mixing things up by combining pre-trained CNNs in new ways to get the best of both worlds: the power of deep learning and the efficiency of transfer learning [39].

The Hybrid Model in this thesis is all about using pre-trained CNNs. It takes advantage of their ability to find important features in images and mixes that with some fine-tuning on a specific dataset to make it even better at a particular task. The pre-trained CNN acts like a feature finder, grabbing high-level features from lots of different images. The following layers are then adjusted to fit the details of the specific job it needs to do.

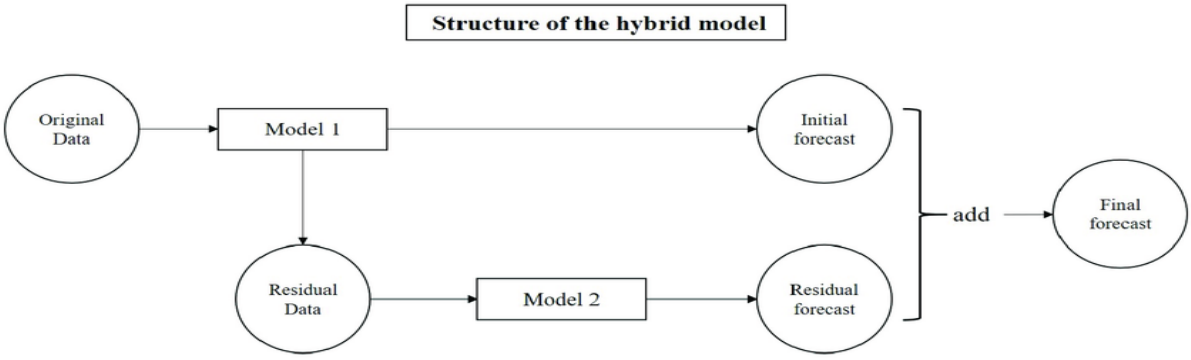


Figure 2.5: Structure of Hybrid CNN Models

One cool thing about this Hybrid Model is that it's great at dealing with not having a ton of labeled data. The pre-trained CNN helps by transferring its knowledge from a big set of images, making the model better at understanding things, especially when getting a lot of labeled data for a specific task is hard or expensive [40].

Also, this Hybrid Model lets researchers try out different pre-trained CNNs, like VGG, ResNet, or Inception. This flexibility means they can pick the best starting model based on what kind of image classification problem they're dealing with. So, it's like having a model that can adapt to different datasets and situations, making it useful for lots of different things.

The experiments in this study show that the Hybrid Model is really effective compared to regular CNNs and pre-trained models used on their own. It not only gets good accuracy but also learns faster and uses computer resources better.

To sum it up, this Hybrid Model is a cool solution for classifying images [41]. It uses the strengths of pre-trained CNNs and tackles the problem of not having a lot of labeled data. This research adds to what we already know and gives us more ideas about how to use

hybrid models in deep learning for looking at images.

2.4 Ensemble learning

Ensemble learning is a collaborative strategy in the area of artificial intelligence that utilizes the capabilities of many neural networks to produce greater performance. Each network, like a team member, provides its own insights, and by pooling their forecasts, the entire model increases resilience and accuracy. This joint effort parallels the interaction of brain networks, where different talents complement one other to promote decision-making. Ensemble learning in CNN is analogous to creating a professional team to handle complicated tasks, resulting to a more dependable and trustworthy AI system [42].

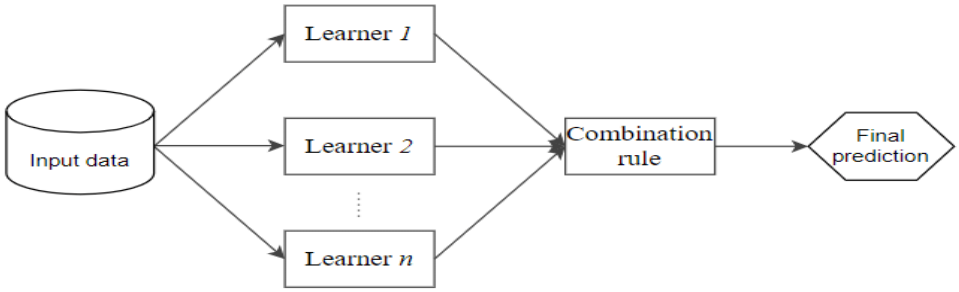


Figure 2.6: Structure of Ensemble Learning

2.5 Vision Transformers and Big Transfer

Vision Transformers (ViT): Traditionally, Convolutional-Neural-Networks (CNNs) have been the go-to for image-related tasks. They break down an image into smaller parts, extract features, and then piece things together. ViTs, on the other hand, leverage Transformer architecture, originally designed for sequences like text. ViTs treat the entire image as a sequence of patches and allow the model to process the global context of the image at once. This helps capture relationships between different parts of the image more efficiently. It's like looking at the whole puzzle instead of just its pieces [43].

Big Transfer (BiT): BigTransfer (also known as BiT) is a state-of-the-art transfer learning system for picture categorization. Transfer of pre-trained representations increases sampling efficiency and simplifies hyperparameter adjustment when training deep neural networks for vision. BiT returns the paradigm of pre-training on huge supervised datasets and fine-tuning the model on a target task. The necessity of carefully picking normalization layers and expanding the architectural capacity as the quantity of pre-training data rises.

BigTransfer(BiT) is trained on public datasets, together with code in TF2, Jax and Pytorch. This will assist anybody to attain state-of-the-art performance in their job of interest, even with only a few of tagged photos each class.

2.6 K-fold cross-validation

K-fold cross-validation is a smart way to ensure that a machine learning model isn't just a one-trick pony. Imagine you're learning a new skill, and instead of practicing it in the same setting every time, you decide to mix things up. K-fold cross-validation does something similar for models. If you choose, let's say, K=5, your data is split into five parts. The model gets trained and tested five times, each time using a different part as the test set and the rest for training. This process gives a more comprehensive assessment of the model's performance, making sure it's not just excelling in one specific scenario but is adept at handling various situations. It's like honing your skills from different angles to become an all-around performer in the world of machine learning.

K-fold cross-validation comes in various flavors, adapting to different needs. Here are a couple of common types:

Standard K-fold Cross-Validation: In the standard version, the dataset is divided into K folds, and the model is trained and evaluated K times. Each time, a different fold is used as the test set, and the remaining folds are used for training. The results are then averaged to provide a more robust performance estimate.

Stratified K-fold Cross-Validation: This variation is useful when dealing with imbalanced datasets, where some classes may have fewer examples. A stratified K-fold ensures that each fold maintains the same class distribution as the original dataset. This helps prevent situations where a specific class is overly represented or underrepresented in the training or test sets.

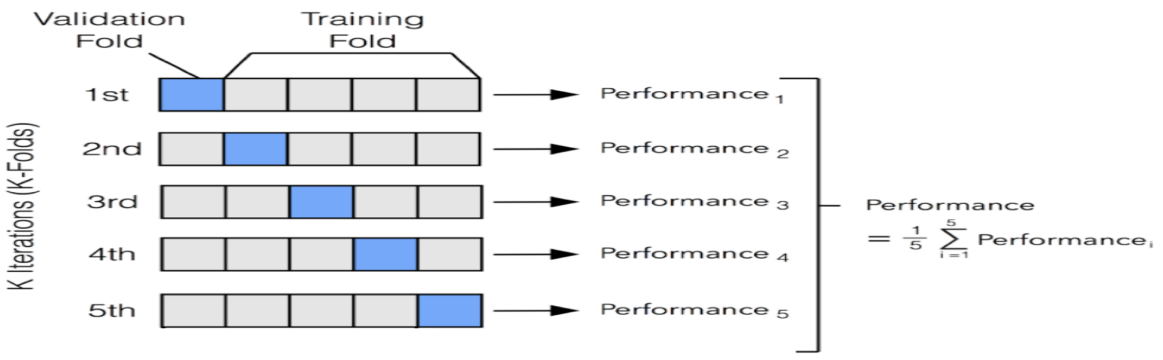


Figure 2.7: Structure of Stratified K-fold Cross-Validation

Leave-One-Out Cross-Validation (LOOCV): Extreme K-fold! In LOOCV, K is set to the total number of samples in the dataset. Each observation is treated as a separate fold for testing, and the model is trained on the rest. This can be computationally expensive but provides a thorough evaluation, especially with smaller datasets.

Time Series Cross-Validation: Specifically designed for time-ordered data, like stock prices or weather patterns. It ensures that training occurs on past data, and testing occurs on future data. This mimics real-world scenarios where predictions are made on new, unseen observations.

2.7 Explainable AI

As the number of deep learning-based techniques increases, there is an increasing need for explainability, particularly in domains where high-stakes decisions need to be made, such medical image analysis [44]. In order to improve readability for medical experts, the DL model findings are further interpreted and explained. This will aid in the quick and correct diagnosis of COVID-19, TB, and pneumonia disorders [45]. Popular XAI algorithms for this include Grad-CAM and LIME, among others.

2.7.1 Local Interpretable Model-agnostic Explanations (LIME)

The LIME (Local Interpretable Model-Agnostic Explanations) method is a model-independent approach for explainable artificial intelligence (XAI). It aims to identify the statistical relationship between the inputs and outputs of a given model [46]. Instead of training global surrogates, LIME perturbs the inputs during training to understand their impact on the model's output.

The process involves creating local surrogate models that approximate the behavior of the original model for specific instances. These surrogate models provide interpretable representations and visualizations of the underlying decision-making process.

The mathematical definition of LIME can be expressed as follows: [46]

Let's consider a prediction model F that takes an input instance X and produces an output Y. LIME aims to approximate the behavior of F around a specific instance X' by training an interpretable model g, which can explain the relationship between the inputs and outputs. The goal is to find a locally interpretable model g(x') that minimizes the loss function L(g, F, πx') while satisfying the complexity constraint Ω(g), where πx' represents the perturbed instances around X'.

The mathematical representation of LIME can be given as: [46]

$$g(x') = \arg\min L(g, F, \pi x') + \Omega(g)$$

The loss function L measures the discrepancy between the predictions of the surrogate model g and the original model F. The complexity constraint $\Omega(g)$ controls the complexity of the interpretable model to ensure its interpretability.

A matrix consisting of 150 perturbations as rows, superpixels as columns, a kernel size of 3 $\times$ 3, and a maximum distance of 100 units is constructed using random ones and zeros. The split ratio of the matrix is 0.2. By sampling from a normal(0,1) and carrying out the inverse operations of mean-centering and scaling, the matrix was perturbed for the top 20 numerical features based on the averages and standard deviation in the training data. Perturb used training distribution-based sampling to generate binary features for categorical characteristics, with a value of 1 indicating correspondence with the example under description.

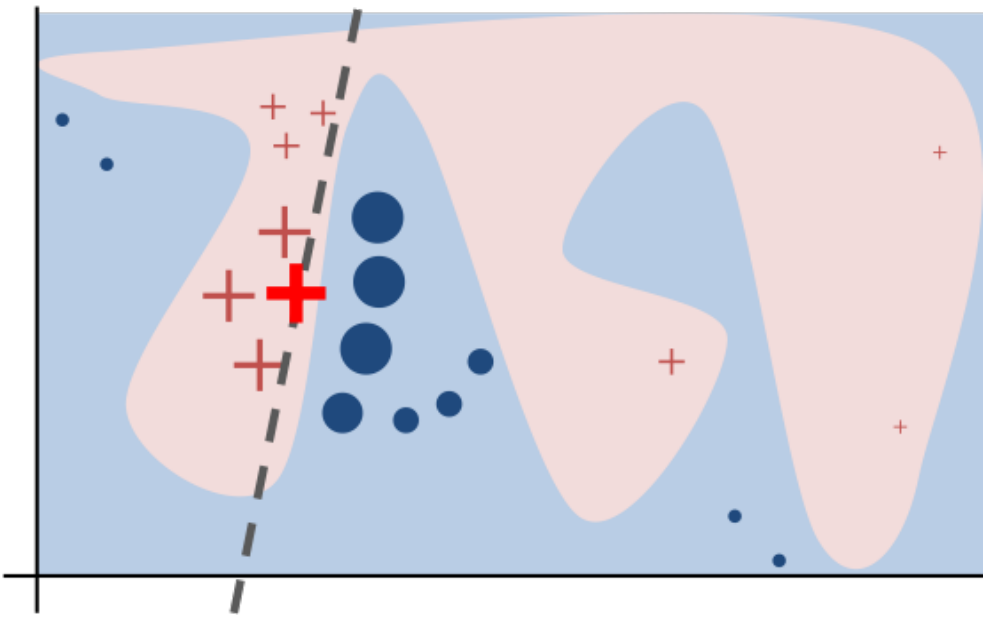


Figure 2.8: Visualization of LIME Algorithms. [6]

2.7.2 Gradient Class Activation Map (Grad-CAM)

A deep learning visualization method called Grad-CAM (Gradient Class Activation Map) creates visual representations of convolutional neural network models. Grad-CAM produces a heatmap with a "class-discriminative localization map" as its output, in which a certain class is represented by the hot spot.

2.8 Underfitting, Optimal fit and Overfitting

Underfitting in deep learning refers to a situation where a machine learning model is unable to adequately capture the underlying patterns and relationships present in the training data. It occurs when the model is too simplistic or lacks the capacity to learn the complexities of the data. As a result, the model fails to generalize well to unseen or test data, leading to poor performance and low accuracy. Also it means that the model is not able to "fit" the training data properly. It is characterized by high bias and low variance [47].

A good fit is often characterized by the model's ability to achieve satisfactory performance metrics, such as high accuracy, low error rate, or appropriate evaluation scores (e.g., precision, recall, F1 score) depending on the specific task [7]. The model should be able to generalize well to new data by correctly identifying patterns and making accurate predictions.

On the other hand, Overfitting in deep learning refers to a situation where a machine learning model performs exceptionally well on the training data but fails to generalize to new, unseen data. It occurs when the model becomes too complex and starts to memorize the noise and random fluctuations present in the training data instead of learning the underlying patterns.Overfitting can be observed when the model has low bias and high variance [47]. It means that the model is highly flexible and sensitive to the specific data points in the training set, but it lacks the ability to capture the broader trends and patterns in the data.

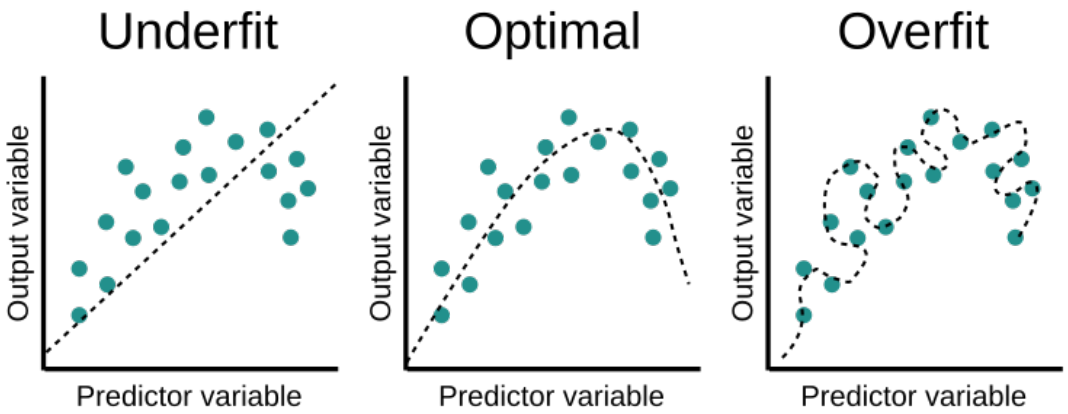


Figure 2.9: Visualization of underfitting, optimal and overfitting [7].

2.9 Evaluation Matrix

An evaluation matrix, also known as an evaluation metric or performance measure, is used to assess the performance of a machine learning model. It provides a quantitative measure of how well the model is performing on a given task or dataset. There are various evaluation matrices used in different machine learning tasks, and the choice of matrix depends on the specific problem and the desired performance criteria [12]. Here are some commonly used evaluation matrices in machine learning:

Accuracy: It measures the proportion of correctly classified instances to the total number of instances in the dataset. It is commonly used for classification problems when the classes are balanced [8].

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN}$$

Precision and Recall: Precision measures the proportion of correctly predicted positive instances among all instances predicted as positive, while recall measures the proportion of correctly predicted positive instances among all actual positive instances [12]. Precision and recall are often used together, particularly in imbalanced classification problems.

$$Precision = \frac{TP}{TP+FP}$$

$$Recall = \frac{TP}{TP+FN}$$

F1 Score: The F1 score is the harmonic mean of precision and recall. It provides a single metric that balances both precision and recall [8]. It is useful when you want to consider both precision and recall equally.

$$F1 = \frac{2*Precision*Recall}{Precision+Recall} = \frac{2*TP}{2*TP+FP+FN}$$

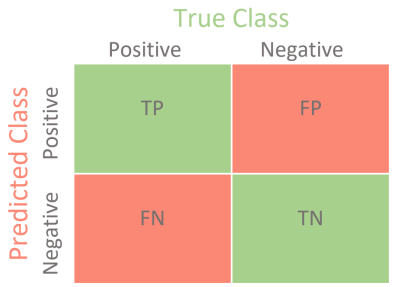


Figure 2.10: 2X2 Confusion Matrix [8].

2.10 Summary

This study delves into the intricacies of lung disease detection, focusing primarily on the Convolutional Neural Network (CNN) and its various layers. From Image Input to Fully Connected layers, each component contributes to the network's ability to identify crucial patterns for accurate disease detection. Beyond CNN, the exploration extends to advanced techniques like the Hybrid Neural Network, Ensemble Learning, and Vision Transformers, offering diverse approaches to tackle the complexities of lung disease identification. The study introduces K-fold cross-validation as a key strategy to ensure the reliability of the models.

Explainable AI techniques, such as Local Interpretable Model-agnostic Explanations (LIME) and Gradient Class Activation Map (Grad-CAM), are discussed, providing insights into the decision-making processes of neural networks. Key concepts like underfitting, optimal fit, and overfitting are explained, offering a nuanced understanding of model performance. The significance of a robust Evaluation Matrix is emphasized as a critical tool for assessing the effectiveness of the developed models. In summary, the study provides a comprehensive exploration of neural network architectures, ensemble methods, interpretability techniques, and evaluation metrics, establishing a solid foundation for advanced lung disease detection models.

Chapter 3

Literature Review

3.1 Overview

To gain a comprehensive understanding and keep up with the latest research trends and findings, We read a large number of publications regarding this topic. Here are a few of the most recent research papers on lung disease classification.

3.2 Deep Learning Approaches

3.2.1 "TX-CNN: Detecting tuberculosis in chest X-ray images using convolutional neural network"

Chang Liu et al. proposed a novel method using Convolutional Neural Network(CNN) to deal with unbalanced, less-category X-ray images [48]. Their method improved the accuracy for classifying multiple TB manifestations by a large margin. Their goal of research is to design and deploy a reliable and efficient system to classify various TB manifestations. They did image classification and trained the network by shuffle sampling with cross-validation. It turned out effective and efficient that it gave outstanding result. They achieved 85.68% of classification accuracy whereas their non-shuffle sampling's accuracy was 53.02%. They claimed they surpassed any state-of-art classification accuracy in that area. They used AlexNet for binary classification and GoogLeNet for full classification. They calculate the precision, recall, f1-score and miss rate. The AlexNet is conducted to determine whether a chest X-ray image contains TB or not. They divided the whole dataset into 5 equal folders and used 4 folders as the training set and the remaining 1 folder as the test set. They claimed their method showed the stability and universality in various CNN architectures.

Moreover, they said their work is the first research trial using CNN for TB detection in a large TB dataset.

3.2.2 "A modified deep convolutional neural network for detecting COVID-19 and pneumonia from chest X-ray images based on the concatenation of xception and ResNet50V2"

Mohammad Rahimzadeh et al. trained several deep convolutional networks with introduced training techniques for classifying X-ray images into three classes: normal, pneumonia, and COVID-19, based on two open-source datasets [49]. They presented a concatenated neural network based on Xception and ResNet50V2 networks for classifying the chest X-ray images. They used two open source datasets where 180 images were containing Covid19 symptoms and 6054 images were containing Pneumonia symptoms. There were also 8851 images of normal people. They achieved an average accuracy of 99.50%, and 80.53% sensitivity for the COVID-19 class, and an overall accuracy equal to 91.4% between five folds. They designed a neural network by concatenating the extracted features of Xception and ResNet50V2 and then connected the concatenated features to a convolutional layer that is connected to the classifier. They claimed this concatenated Neural Network has shown higher accuracy compared to others. They trained the networks using the Categorical cross-entropy loss function and Nadam optimizer. Data augmentation methods were also implemented to increase training efficiency and prevent the model from overfitting. However, they can improve their work as there are several meaningless results due to imbalance dataset.

3.2.3 "COVID-19 Diagnosis from Chest X-ray Images Using Deep Learning Approach"

This paper addresses a critical issue in the field of COVID-19 diagnosis and proposes a deep learning-based solution using chest X-ray images [50]. The topic is timely and relevant, given the challenges associated with the traditional RT-PCR testing method. The abstract provides a concise overview of the paper's objectives and the motivation behind the study. It effectively highlights the need for a faster and more efficient diagnostic method, leading to the exploration of chest X-ray images and deep learning techniques. The research paper mentions the development of a CNN architecture model for binary, three-class, and four-class classifications. The achieved testing accuracies of 99.7%, 95.02%, and 94.53% are impressive and indicate the potential of the proposed model for COVID-19 diagnosis using chest X-ray images. However, it would be helpful to have more details on the dataset used,

including its size, composition, and any potential biases or limitations. The paper claims superiority over other related works in the field, but it would be beneficial to provide more in-depth comparisons and discussions. This could include a detailed analysis of the strengths and weaknesses of the proposed model in comparison to other existing methods. Additionally, the authors could consider discussing any limitations or challenges encountered during the development and evaluation of the model. Overall, the paper presents a promising approach for COVID-19 diagnosis using chest X-ray images and deep learning. The high testing accuracies achieved by the proposed model demonstrates its potential usefulness. However, further exploration and analysis are needed to fully evaluate its performance in comparison to other existing approaches.

3.2.4 "Coronet: A deep neural network for detection and diagnosis of COVID- 19 from chest x-ray images"

The research paper "Coronet: A deep neural network for detection and diagnosis of COVID-19 from chest x-ray images" addresses a critical need for an accurate and efficient diagnostic tool to aid in the detection of COVID-19 using chest X-ray images [51]. The abstract provides a concise background and objective, emphasizing the global impact of COVID-19 and the shortage of testing kits and resources. The authors propose CoroNet, a CNN model based on the Xception architecture, trained on a dataset comprising COVID-19 and chest pneumonia X-ray images from public databases. This choice of architecture and dataset is well-motivated and aligns with established practices in deep learning. The results section reports an overall accuracy of 89.6% for the proposed model. Additionally, the precision and recall rates for COVID-19 cases in the 4-class classification (COVID vs. Pneumonia bacterial vs. pneumonia viral vs. normal) are reported as 93% and 98.2%, respectively. In the 3-class classification (COVID vs. Pneumonia vs. normal), the proposed model achieves a classification accuracy of 95%. These results are encouraging, indicating the potential effectiveness of CoroNet in COVID-19 diagnosis using chest X-ray images. The conclusion highlights the significance of the proposed model, particularly during the COVID-19 pandemic, as a valuable tool for clinical practitioners and radiologists in diagnosis, quantification, and follow-up of COVID-19 cases. However, it is important to note that the study acknowledges the use of a small prepared dataset and expresses the need for more training data to further improve the model's performance. It would be beneficial if the paper provides insights into the limitations and challenges faced during data collection, dataset composition, and potential biases. Finally, "Coronet: A deep neural network for detection and diagnosis of COVID-19 from chest x-ray images" presents a promising deep learning-based approach to COVID-19 detection. The reported results indicate the potential usefulness of the proposed model, especially when trained on larger and more diverse datasets. Further validation

and evaluation on external datasets would be valuable to establish the generalizability and robustness of CoroNet.

3.2.5 "Attention-based VGG-16 model for COVID-19 chest X-ray image clas- sification"

The research paper "Attention-based VGG-16 model for COVID-19 chest X-ray image classification" addresses the need for an accurate and efficient CAD method for the early detection of COVID-19 using CXR images [52]. The abstract provides a clear overview of the paper's objective and highlights the advantages of CXR-based methods as a cost-effective alternative to other diagnostic techniques. The authors identify the limitation of previous works, which fail to consider the spatial relationships between ROIs in CXR images, and propose an attention-based deep learning model using the VGG-16 architecture. This approach demonstrates the authors' innovative thinking in combining attention mechanisms with a well-established CNN architecture. The paper emphasizes the importance of the attention module in capturing the spatial relationships between ROIs in CXR images. By incorporating an appropriate convolution layer (4th pooling layer) of the VGG-16 model, the proposed method achieves fine-tuning in the classification process, leading to improved performance. The strength of the paper is the extensive experiments that the authors have conducted. The authors have used three different datasets, which helps to ensure that the results are generalizable. The authors have also evaluated the performance of their model on the test set, which is a good way to assess the generalization ability of the model. Also, the limitations of the paper is the paper only used three datasets. It would be helpful to see the performance of the model on more datasets and the paper did not evaluate the clinical impact of the model. It would be helpful to see how the model performs in a clinical setting. The authors conduct extensive experiments using three COVID-19 CXR image datasets to evaluate the performance of their proposed method. Although the paper mentions the stable and promising performance of the proposed method compared to state-of-the-art approaches, it would be beneficial to provide more specific details about the evaluation metrics used, such as accuracy, precision, recall, or F1 score. Additionally, insights into the datasets used, including their size, composition, and potential biases, would enhance the transparency of the study. Therefore, "Attention-based VGG-16 model for COVID-19 chest X-ray image classification" presents an innovative approach to COVID-19 detection using CXR images. The incorporation of the attention module with the VGG-16 architecture addresses the limitations of previous works and shows promise in improving classification performance. Further validation and comparison with external datasets would strengthen the credibility and generalizability of the proposed method.

3.2.6 "CheXImageNet: a novel architecture for accurate classification of Covid-19 with chest x-ray digital images using deep convolutional neural networks"

The research paper "CheXImageNet: a novel architecture for accurate classification of Covid-19 with chest x-ray digital images using deep convolutional neural networks" focuses on the development of an accurate diagnostic tool for the detection of Covid-19 using digital chest X-ray images [53]. The abstract provides a clear overview of the current challenges in Covid-19 detection, emphasizing the need for early identification to control the spread of the virus. Radiologic imaging, specifically X-ray imaging, is recognized as a critical approach for diagnosing and treating Covid-19 patients. The authors suggest that deep learning methodologies, when combined with digital radiological imaging, can be valuable in accurately identifying the disease. The paper introduces the CheXImageNet model, designed for detecting Covid-19 using openly accessible chest X-ray image datasets. The reported accuracies of 100% for both binary and three-class classifications are impressive. However, it would be beneficial to provide more details about the dataset used, including its size, composition, and potential biases. Additionally, insights into the evaluation metrics employed, such as precision, recall, or F1 score, would provide a more comprehensive understanding of the model's performance. The proposed CheXImageNet model has the potential to be a useful tool for Covid-19 detection, particularly in areas with limited access to competent physicians. However, it is important to note that further validation and evaluation on external datasets would be necessary to establish the generalizability and robustness of the model. Overall, "CheXImageNet: a novel architecture for accurate classification of Covid-19 with chest x-ray digital images using deep convolutional neural networks" presents an innovative approach to Covid-19 detection using digital chest X-ray images. The reported accuracies indicate the potential effectiveness of the proposed model. However, more comprehensive details and external validation would strengthen the credibility of the study.

3.2.7 "An efficient mixture of deep and machine learning models for COVID- 19 and tuberculosis detection using X-ray images in resource-limited settings"

This research paper focuses on the development of an efficient and accessible methodology for COVID-19 detection in resource-limited settings using chest X-ray images [54]. The abstract provides a clear overview of the challenges faced in quickly and accurately detecting COVID-19, especially in resource-limited settings without access to biotechnology tests. The authors highlight the overlap in symptoms between COVID-19 and tuberculosis, adding to the complexity of accurate diagnosis. The proposed methodology involves extracting deep

features (DF) from chest X-ray images and utilizing machine learning classifiers to classify the images into different classes. The authors evaluate multiple pipelines and identify the best-performing one, which combines ResNet-50 for DF computation and an ensemble of subspace discriminant classifiers. The paper presents promising results, with the best-performing pipeline achieving a detection accuracy of 91.6% for the five-class classification problem. The methodology is computationally efficient, with a short extraction time of 0.19 seconds per X-ray image and a training time of 2 minutes on a CPU machine. The paper provides specific accuracy results for simpler three-class and two-class classification problems as well. The availability of the constructed dataset is a valuable contribution to the research community, facilitating further studies and comparisons. The authors emphasize the potential benefits of their pipeline in the detection of COVID-19, particularly in resource-limited settings where accessible X-rays and limited computational resources are available. Lastly, "An efficient mixture of deep and machine learning models for COVID-19 and tuberculosis detection using X-ray images in resource-limited settings" presents an effective methodology for COVID-19 detection using chest X-ray images. The results demonstrate the potential of the proposed pipeline in accurately classifying different classes related to COVID-19, and the computational efficiency makes it well-suited for resource-limited settings. The availability of the dataset further contributes to the research community.

3.2.8 An explainable artificial intelligence model for identifying local indicators and detecting lung disease from chest X-ray images

In order to identify lung sickness, the authors Shiva Prasad Koyyadaa and Thipendra P. Singh use chest X-ray images to design and train a model on a binary classification problem COVID vs. NORMAL lungs [55]. They simulated the decision-making process of a radiologist by using weekly supervised learning to identify local discriminating regions of a chest X-ray image, deriving rules, and elucidating why the Deep Learning (DL) method yields such results. Compared to state-of-the-art techniques, they achieved very good results while using a small number of photos. There are three stages to this process. During the first step, the data is split into validation and train sets. Then they trained a model using the train data set to predict lung illness. Here, transfer learning methods such as VGG16, GoogleNet, MobileNet, etc are used as feature extractors to classify the medical images. Finding crucial regions and training a model on the photos that have been shown to have critical regions constitute phase two. In this phase, they share the model and image with Local Interpretable Model-agnostic Explanations (LIME). Phase three classifies the diseases by combining the global and local information with additional patterns learned. Here, they were extracting global and local features from the flattened layer separately. Significant progress has been made by the local and fusion models, achieving 99.6% accuracy with fewer epochs.

3.2.9 Explanatory classification of CXR images into COVID-19, Pneumonia and Tuberculosis using deep learning and XAI

This paper presents that utilizing chest X-rays (CXR) images, physicians and other medical professionals can detect and classify COVID-19, pneumonia, and tuberculosis (TB) with greater persuasiveness and coherence using the XAI and DL models [10]. The abstract of this paper provides a clear overview of the result, methodology, and other processing. In the introduction, Mohan Bhandari et al. mentioned that they used DL models and these DL models have some limitations. To address these limitations, they propose an XAI-based DL architecture to detect and classify COVID-19, Pneumonia, and TB with CXR images. To classify and identify the affected area of the CXR picture, the used CNN models and used three popular XAI algorithms: Shapley additive explanation (SHAP), local interpretable model-agnostic explanation (LIME), and Gradient weighted Class Activation Mapping (Grad-CAM). They use a small dataset of 7132 chest x-ray (CXR) medical images. Finally, the CNN model achieved 95.76±1.15%training accuracy, an average test accuracy of 94.31 ± 1.01%, and a validation accuracy of 94.54 ± 1.33% for 10-fold cross-validation. In conclusion, compared to the most advanced techniques, the suggested model is more accurate and has a lightweight architecture for categorizing CXR photos in addition to XAI.

3.2.10 Multi-Scale CNN: An Explainable AI-Integrated Unique Deep Learning Framework for Lung-Affected Disease Classification

Ovi Sarkar et al. introduce a Deep Learning (DL) architecture called Multi-Scale Convolutional Neural Network (MS-CNN) [56]. It is developed to efficiently identify between normal and six different lung-related disorders (specifically, bacterial pneumonia, COVID-19, fibrosis, lung opacity, tuberculosis, and viral pneumonia). After that, they integrate explainable AI (XAI) to enhance prediction capability. This new approach sets a new benchmark for lung disease diagnosis, outperforming many other approaches currently in use. The integration of XAI techniques, such as Grad-CAM and SHAP, has improved the predictability and interpretability of the model. The discoveries have the potential to significantly increase the confidence and speed of diagnostic judgments made in the identification of lung diseases. They combine four publicly available datasets to provide a strong dataset with 6650 CXR pictures divided into seven different categories. With average values for precision, recall, F1-score, and AUC for the seven-class classification at 0.97, 0.95, 0.95, and 0.94, respectively, results showed an astounding accuracy of 96.05%. In conclusion, the MS-CNN model detects COVID-19 with impressive efficiency compared with TL models (VGG16 and VGG19) and other SOTA models. So, the MS-CNN model detects COVID-19 with impressive efficiency.

3.3 Summary

Table 3.1: **Summary of the Review Paper**

Ref	Year	Cases	Classes	Method	XAI	Acc(%)
[48]	2017	TB=4248 Normal=453	2	Transfer learning with AlexNet and GoogleNet	No	85.68
[49]	2020	Normal=8851 Covid19=180 Pneumonia=6054	3	Ensemble of Xception and ResNet50	No	91.40
[50]	2020	Normal=1583 Covid19=576 Pneumonia=4273 TB=155	4	Custom CNN	No	94.53
[51]	2020	Normal=310 PneumoniaB=330 PneumoniaV=327 Covid19=284	4	CNN-based CoroNet	No	89.60
[52]	2021	Normal=310 PneumoniaB=330 PneumoniaV=327 Covid19=284	4	Attention based VGG	No	85.43

Table 3.2: **Summary of the Review Paper**

Ref	Year	Cases	Classes	Method	XAI	Acc(%)
[53]	2022	Normal=1341 Covid19=864 Pneumonia=1345	3	Inception-V3 with Transfer learning	Yes	93.00
[54]	2020	Normal=439 Covid-19=435 PneumoniaB=439 PneumoniaV=439 TB=434	5	Transfer learning with ResNet18	No	91.60
[55]	2023	Normal=1341 Covid-19=2896	2	CNN	Yes	99.6
[10]	2022	Normal=1583 Covid-19=576 PneumoniaB=4273 TB=700	4	CNN	Yes	94.31
[56]	2023	Normal=950 Covid-19=950 Fibrosis=950 LungOpacity=950 PneumoniaB=950 PneumoniaV=950 TB=950	7	MS-CNN	Yes	96.05

Chapter 4

Dataset Analysis

A dataset is a critical component of deep learning because it is used to train and evaluate the performance of machine learning models. The quality and quantity of the dataset play a crucial role in the accuracy and generalization of the trained models [45].

Lungs Disease Dataset (4 types) appears to be a public dataset of chest X-ray images that has been categorized into 4 types of lung diseases (Viral Pneumonia, Bacterial Pneumonia, Covid, and Tuberculosis) and Normal Lungs.

The dataset contains 10,095 images and is split into three directories:
1. **Test**
2. **Train** and
3. **Validation**.

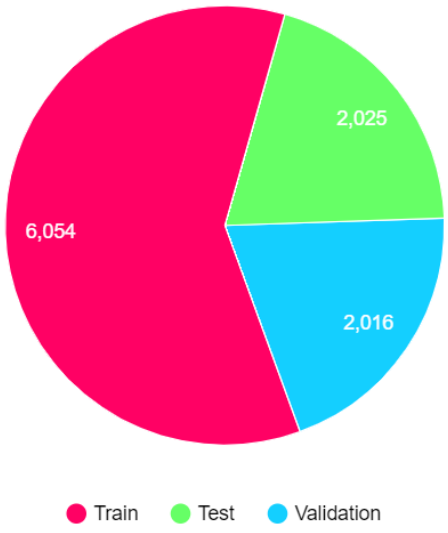


Figure 4.1: Dataset orientation.

The Test directory contains 2,025 images, the Train directory contains 6,054 images, and the Val directory contains 2,016 images. It is prepared from various datasets. It is combined to remove the same images in the dataset using VisiPics [When two identical pictures are stored in different formats or resolutions, Visipics will identify them as duplicates even if they are identical but for little aesthetic differences.] [57].

1. **Test-** 2025 images of Bacterial Pneumonia (403), Corona Virus Disease (407), Normal (404), Tuberculosis(408), Viral Pneumonia(403)
2. **Train**- 6054 images of Bacterial Pneumonia (1205), Corona Virus Disease (1218), Normal (1207), Tuberculosis(1220), Viral Pneumonia(1204)
3. **Validation**- 2016 images of Bacterial Pneumonia (401), Corona Virus Disease (406), Normal (402), Tuberculosis(406), Viral Pneumonia(401)

4.1 Exploratory Data Analysis (EDA)

Any dataset analysis and comprehension process begins with EDA. In the case of a medical dataset containing information about different-types-of lung diseases, we perform various analyses to gain insights into the data. Here's a general EDA on our medical lung disease dataset:

4.1.1 Visualize Training samples

Visualizing sample images from your dataset is crucial for understanding the nature of the data and gaining insights into the characteristics of different classes.

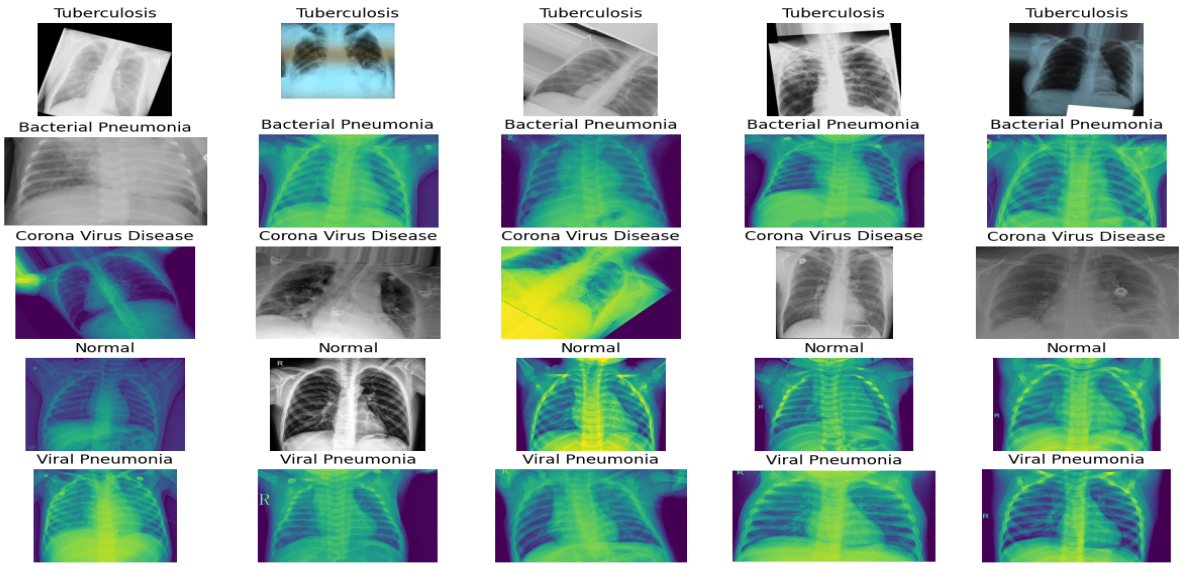


Figure 4.2: Sample Images Per Disease

4.1.2 Class Distribution

Class distribution refers to the proportion of data points belonging to each class in a dataset. Balanced class distribution is ideal for training classification models, while imbalanced class distribution can lead to biased predictions.

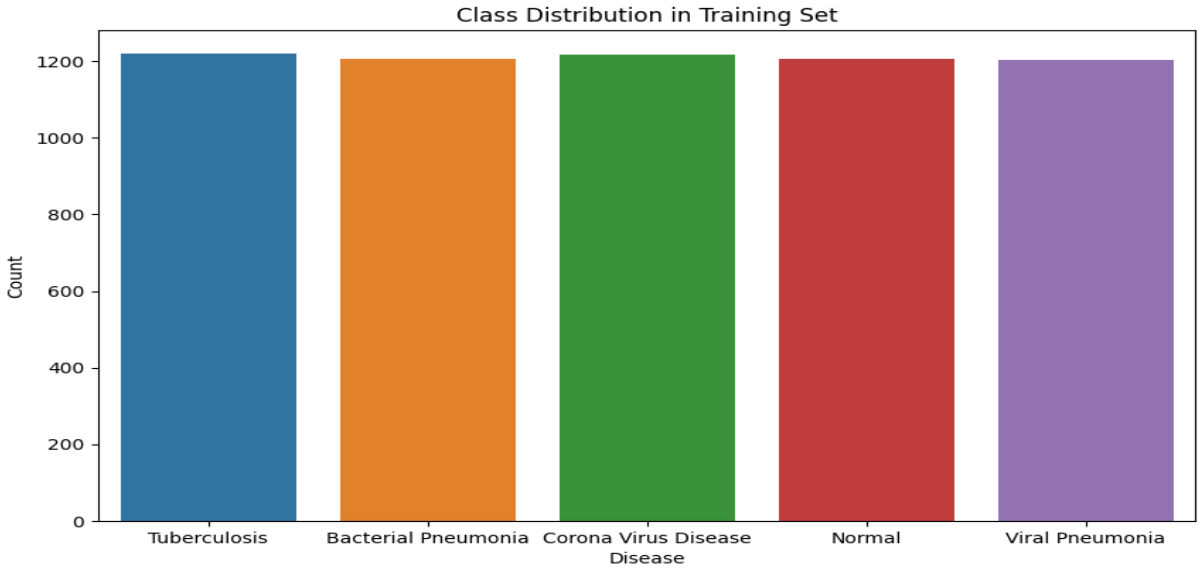


Figure 4.3: Class distribution of the training dataset

4.1.3 Sample Image Size Distribution

The distribution of image sizes refers to the frequency of different image dimensions (width and height) within a dataset.

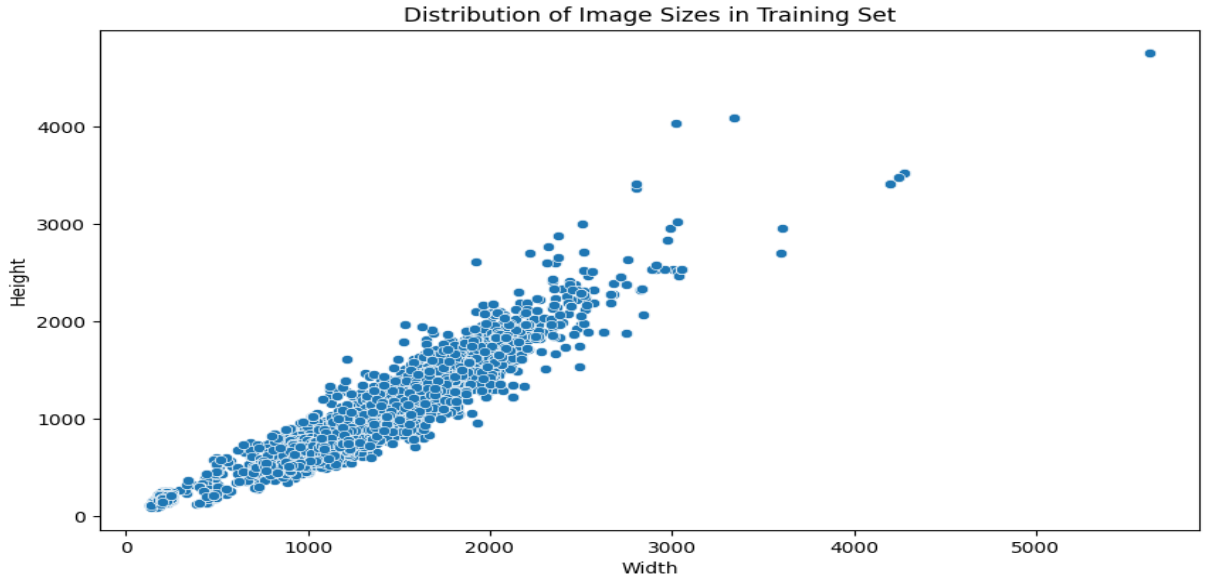


Figure 4.4: Size distribution of the images in training dataset

4.1.4 Viral Pneumonia Class Pixel Distribution

Pixel distribution is a statistical measure that describes the frequency of different pixel intensities within an image. It is a crucial aspect of image processing and analysis, as it provides insights into the overall brightness, contrast, and color characteristics of an image.

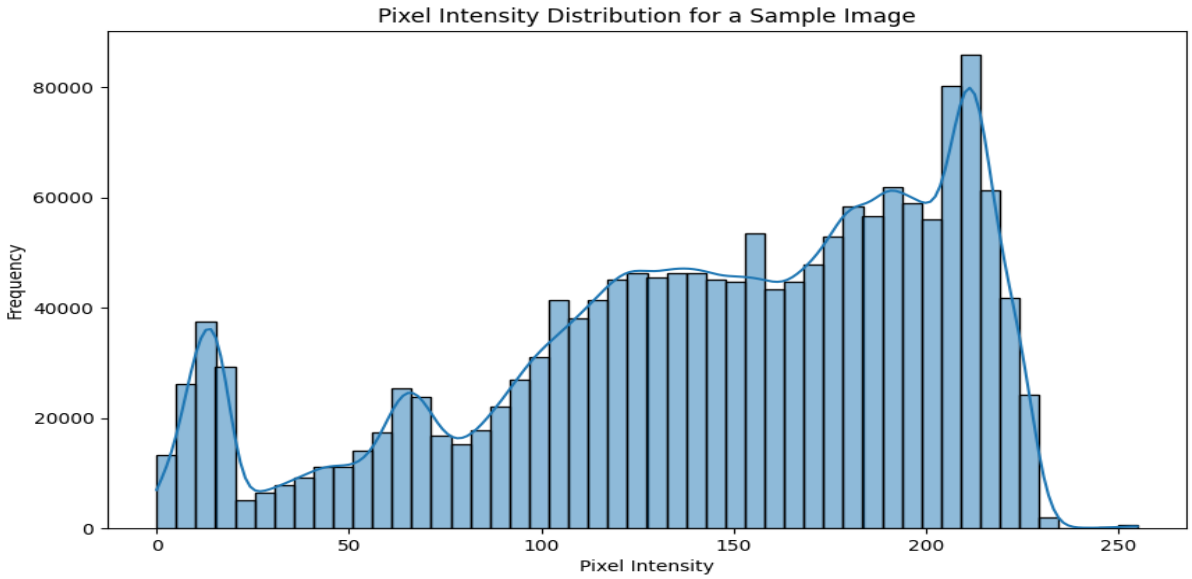


Figure 4.5: Pixel intensity distribution for a training sample image

4.1.5 Sample Image Histogram

An image histogram is a graphical representation of the tonal distribution in a digital image. It plots the number of pixels for each tonal value. By looking at the histogram for a specific image, a viewer will be able to judge the entire tonal distribution at a glance.

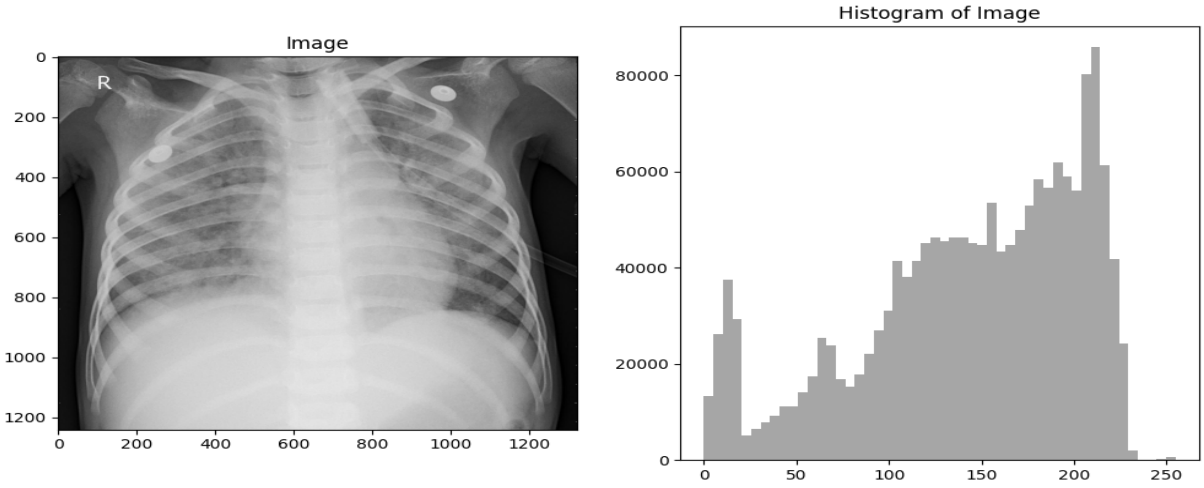


Figure 4.6: Sample Images Histogram visualization

4.1.6 Some Bad Training Images

Most of the images in the dataset are not in good condition. Some images are hazy and can not be perceived. Also, some are deep white that can not understand anything inside the lungs. Most of the images are in shear or rotated and white dominates these images.

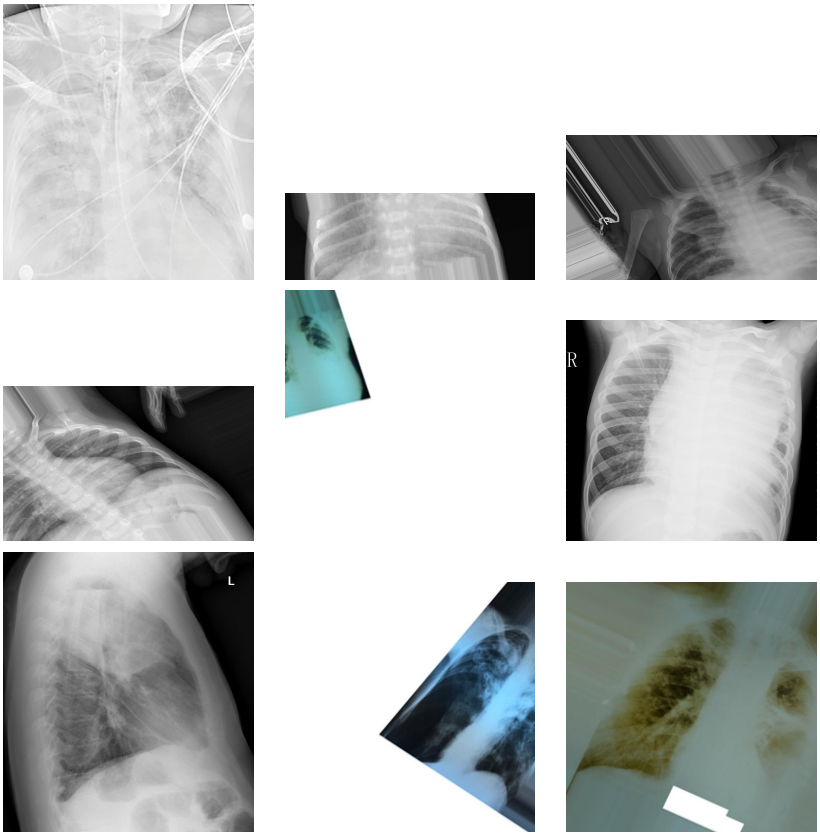


Figure 4.7: Bad training samples

4.2 Summary

Finally, it can be said that the dataset is split into 20.06 percent test data, 59.97 percent of train data, and 19.97 percent of validation data. This dataset is used for developing and evaluating our deep-learning models for the detection and classification of lung diseases.

Chapter 5

Proposed Methodology

5.1 Overview

We tried to experiment with a Chest X-ray of the lung to detect four types of lung disease and normal lungs. In this section, we will explain how we started our experiment and how it continued further.

5.2 Data Collection

We used the dataset Lungs Disease Dataset (5 types) [45], which is collected from the Kaggle website. This dataset was prepared from various datasets and combined together. This dataset has been categorized into 4 types of lung diseases (Viral Pneumonia, Bacterial Pneumonia, Covid, and Tuberculosis) and Normal Lungs. Same images were removed by using VisiPics. The dataset was augmented with factor 6. This is a balanced dataset as the number of samples or instances in each class or category is approximately equal or well-distributed [58]. In other words, all classes in the dataset have a similar number of samples, resulting in a balanced representation of each class.

5.3 Data Preprocessing

Data Preprocessing section is divided into two subsections. One is Image processing another is Image augmentation.

5.3.1 Image-Processing

Histogram Equalization

Histogram equalization is a powerful image preprocessing technique aimed at improving contrast and visual quality. Beginning with the importation of essential libraries such as OpenCV and NumPy, the process involves loading the image in grayscale to simplify subsequent operations and analyzing its histogram to understand pixel intensity distribution. The computation of the Cumulative Distribution Function (CDF) and its normalization ensures that pixel intensities are appropriately scaled. By mapping the original pixel values to their adjusted counterparts, the technique effectively redistributes intensities, enhancing contrast. Converting the resulting image to an unsigned 8-bit integer ensures compatibility with standard image formats [59]. Overall, histogram equalization proves to be a comprehensive approach for contrast enhancement, making images more suitable for subsequent image processing tasks.

CLAHE

Contrast Limited Adaptive Histogram Equalization (CLAHE) is an advanced image enhancement technique, building upon traditional histogram equalization. Unlike global histogram equalization, CLAHE operates on small regions of the image, adaptively enhancing contrast. This adaptive nature prevents over-amplification of noise in homogeneous regions. The algorithm divides the image into tiles and applies histogram equalization independently to each tile. To avoid abrupt intensity transitions at tile borders, contrast limitations are imposed. CLAHE is particularly effective in scenarios with varying illumination across an image, providing improved local contrast and better-preserving details. Implementation often involves setting parameters such as tileGridSize=(8, 8) and clipLimit 2.0 to tailor the enhancement to image characteristics, making CLAHE a valuable tool in medical imaging, satellite imagery, and various computer vision applications [60].

CLAHE and Gamma

Contrast Limited Adaptive Histogram Equalization (CLAHE) and gamma correction are two distinct image processing techniques, each addressing different aspects of image enhancement.

CLAHE, as mentioned earlier, is focused on adaptive contrast enhancement. This adaptive approach is particularly beneficial in scenarios where the illumination varies across the image, providing localized contrast enhancement. Implementation often involves setting pa-

rameters such as tileGridSize=(8, 8) and clipLimit 2.0 to tailor the enhancement to image characteristics [61].

On the other hand, gamma correction is a technique used to adjust the brightness and contrast of an image by altering the gamma value. The gamma value is a parameter that controls the relationship between the pixel values in the input and output images. Adjusting gamma can be useful for correcting nonlinearities in image acquisition devices or compensating for the nonlinear response of human vision. In our dataset, we set the gamma value as 2.0.

Combining CLAHE with gamma correction can yield enhanced results. Gamma correction can be applied before or after CLAHE, depending on the specific requirements of the image. We applied the Gamma correction after CLAHE.

Canny Edges

Canny edge detection is a prominent image processing technique designed to identify and highlight edges within an image while minimizing noise. This algorithm involves smoothing the image with a Gaussian filter to reduce noise, calculating gradients to highlight intensity changes, and thinning edges through non-maximum suppression. It employs a hysteresis-based approach to categorize and track edges, distinguishing between strong and weak edges and ensuring connectivity for a refined edge map. Canny edge detection finds widespread application in computer vision tasks, including object recognition, image segmentation, and feature extraction. Instead of a Gaussian filter, we used a median filter and set the two threshold values according to our dataset. So threshold1 was 120 and threshold2 was 255.

Complement

In the context of image processing, the term "complement" typically refers to the complement of an image. The complement of an image is obtained by subtracting each pixel value from the maximum intensity value.

Mathematically, for a grayscale image with pixel values ranging from 0 to 255, the complement (C) can be expressed as:

C(x,y)=max intensity valueI(x,y)

where I(x,y) represents the pixel intensity at position (x,y) in the original image. This formula subtracts each pixel value from the maximum intensity value to obtain the complement.

5.3.2 Image Augmentation

After collecting the dataset, we did some preprocessing so that the models can run the dataset smoothly. To improve the classification model's performance and increase the number of samples in the dataset, various image augmentation techniques were applied. These techniques introduce variations to the existing images by applying transformations such as rotation, zooming, shifting, shearing, and flipping. The specific parameters used for augmentation were as follows:

5.3.3 Rescaling

The rescale=1./255 parameter rescales the image values to be between 0 and 1. This is a common preprocessing step for image classification tasks, as it normalizes the pixel values and makes the data more suitable for neural networks. Rescaling ensures that all images have the same value range, regardless of their original intensity levels.

5.3.4 Shifting

The width_shift_range=0.2 and height_shift_range=0.2 parameters randomly shift the image horizontally and vertically, respectively. This augmentation technique effectively expands the dataset by creating new variations of existing images. By slightly shifting the image, the model is forced to learn more robust features that are not sensitive to small changes in position.

5.3.5 Zooming

The zoom_range=0.2 parameter randomly zooms in or out of the image by up to 20%. Zooming introduces variations in the scale of the objects in the images, making the model more resilient to different object sizes. By learning to recognize objects at different scales, the model can better generalize to new images.

5.3.6 Flipping

The horizontal_flip=True parameter randomly flips the image horizontally. This augmentation technique doubles the dataset by creating mirror images of the original images. Horizontal flipping forces the model to learn features that are invariant to left-right orientation, making it more robust to variations in image orientation.

5.3.7 Rotation

The rotation_range=10 parameter randomly rotates the image by up to 10 degrees. Rotation introduces variations in the orientation of the objects in the images, making the model more resilient to different object orientations. By learning to recognize objects at different rotations, the model can better generalize to new images.

5.3.8 Shearing

The shear_range=0.2 parameter randomly shears the image horizontally or vertically by up to 20%. Shearing introduces distortions in the shape of the objects in the images, making the model more robust to variations in object shapes. By learning to recognize objects with different shapes, the model can better generalize to new images.

5.3.9 Brightness

The brightness_range=[0.8, 1.2] parameter randomly changes the brightness of the image by up to 20%. Brightness augmentation introduces variations in the lighting conditions of the images, making the model more resilient to different lighting environments. By learning to recognize objects under different lighting conditions, the model can better generalize to new images.

5.4 Deep Learning Methodology

Deep learning methodology refers to the systematic approach and set of techniques used in the application of deep learning algorithms to solve complex problems. It involves a structured process that encompasses steps such as problem definition, data collection and preprocessing, model architecture design, model training, model evaluation, hyperparameter tuning, and deployment [9].

The methodology aims to effectively utilize deep neural networks to extract meaningful representations from large datasets and learn complex patterns and relationships within the data. It involves iterative experimentation, optimization, and refinement to develop models that can accurately generalize and make predictions on unseen data. Deep learning methodology combines principles from machine learning, neural networks, and computational algorithms to tackle challenging tasks across various domains such as computer vision, natural language processing, and speech recognition [33].

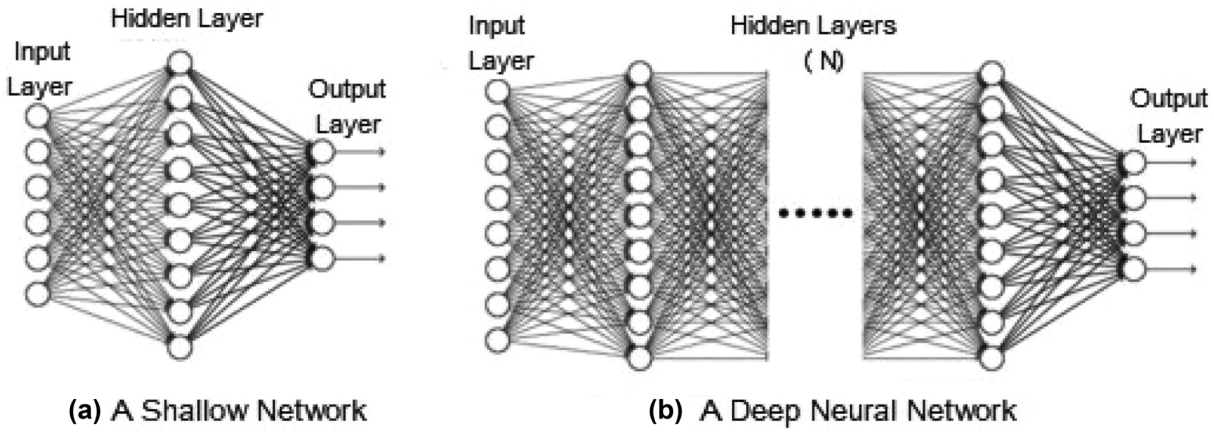


Figure 5.1: Shallow Network Vs Deep Neural Network [9]

5.4.1 Pre-trained single CNN Model

We ran 7 (seven) pre-trained models on the dataset namely Xception, Inception-V3, VGG19, EfficientNetB7, DenseNet201, DenseNet121 and ResNet50 [13]. The batch size is set to 32, indicating that during each iteration of training, the model processes 32 samples simultaneously. The learning rate is set to 1e-4, representing the step size at which the optimizer adjusts the model's parameters during training. The training process is composed of 6000 iterations, and an epoch is defined as the number of iterations required to process the entire training dataset once. The number of epochs is calculated as the total number of iterations divided by the number of batches in the training dataset. That means

$$\text{Epochs} = \left\lfloor \frac{\text{num\_iters}}{\left(\frac{\text{generator\_train.n}}{\text{batch\_size}}\right)} \right\rfloor$$

In this case, it is set to 31. The optimizer used is Adam, a popular optimization algorithm that adapts the learning rates of each parameter during training. The loss function employed is categorical_crossentropy, which is commonly used for multiclass classification problems, measuring the difference between the predicted class probabilities and the true class labels. This configuration collectively defines the training setup for a neural network, specifying crucial parameters for the optimization process. After completing the training phase, we got some results on test data.

5.4.2 Hyper-parameter Tuning

Hyperparameter tuning is like finding the best settings for a computer to learn and make good predictions. In machine learning, we have these settings called hyperparameters, which are a bit like the recipe for training the computer. Getting the hyperparameters just right is crucial. We can't let the computer figure them out on its own, so we have to choose them before training. Hyperparameter tuning is like adjusting these settings to get the best

performance from the computer.

The goal is to find the perfect combination of settings so the computer can learn well from the data and make accurate predictions. If we don't get these settings right, the computer might struggle to learn (underfitting) or remember the training data too much (overfitting), which is not good when it sees new data.

We followed the grid search technique and worked hard to make our computer model perform its best. Here are the details:

Optimizer: We tried different methods (optimizers) to help the model learn. Among choices like Adam, SGD, and Adagrad, we found that using Adam with a learning rate of 1e-4 worked the best. Adam's special learning features made our model learn efficiently.

Dropout layers: We didn't want our model to remember the training data too much, so we played with something called dropout rates. After trying rates like 0.5, 0.6, 0.7, 0.8, and 0.9, we found that setting it at 0.6 was just right. This keeps a good balance between learning from the data and not memorizing it too well.

Batch Size: We looked at different batch sizes, or the number of examples the model works within each step of learning. The best size was 32, beating out options like 8, 16, and 64. This way, the model learns well without using too much computer power.

Epochs: We played around with how many times the model goes through the whole set of data (epochs) during training. After trying 10, 20, 31, and 40 rounds, we saw that 31 was the best number. This strikes a good balance between learning well and not getting too stuck on the training data.

Best	All Possible Parameters
Adam (learning_rate=1e-4)	[Adam, SGD, Adagrad]
0.6	[0.5, 0.6, 0.7, 0.8, 0.9]
32	[8, 16, 32, 64]
31	[10, 20, 31, 40]

Table 5.1: Hyperparameter Tuning with Grid Search

5.4.3 Transfer Learning and Fine Tune

Transfer learning:

Transfer learning is a machine learning technique where a model trained on one task is adapted for a second related task. Instead of training a model from scratch on a new task, transfer learning takes advantage of the knowledge gained during the training of a related

task. This can be especially beneficial when the amount of labeled data for the new task is limited.

In the context of neural networks, transfer learning often involves using a pre-trained model on a large dataset (usually for a more general task like image classification) and then applying this model to a different but related task. The intuition is that the features learned by the model in the first task can be useful for the second task.

```
for layer in new_model.layers[-50:]:
    layer.trainable = True
```

The model (new_model) was used which was a pre-trained DenseNet121 model. The layers of this pre-trained model are then fine-tuned for a specific task using the last 50 layers.

Fine-Tuning:

Fine-tuning is a specific form of transfer learning where a pre-trained model is further trained on a new task. Instead of training the entire model, which could be computationally expensive, fine-tuning typically involves adjusting the weights of some of the layers in the pre-trained model while keeping others frozen.

In practice, this means that the early layers of the model, which capture general features, are often kept frozen, and only the later layers are adjusted to the specifics of the new task. This allows the model to adapt its learned representations to the nuances of the new data without losing the valuable knowledge gained during the pre-training phase [62].

The loop iterated over the last 50 layers of the model, making them trainable. This means that during training, the weights of these specific layers can be adjusted based on the new data for the task at hand.

5.4.4 Hybrid Neural Network

We employed a hybrid model approach by combining three prominent pre-trained models, namely InceptionV3, VGG19, and Xception, in various pairs to address a multi-class means 5 (five) classification task. Our hybrid models are:

The training process involved applying data augmentation techniques to enrich the dataset, and hyperparameters such as a learning rate of 1e-4, 10 epochs, and batch size 10 were carefully chosen. The Adam optimizer was utilized with the categorical cross-entropy loss function to optimize the models. Post-training, the models were evaluated on a distinct test dataset, and their performance was measured in terms of accuracy. Notably, the hybrid models demonstrated varying degrees of accuracy, with the InceptionV3+Xception combination exhibiting the highest accuracy of 89%. The choice of batch size differed for different model

Model Combination	Batch Size
InceptionV3+VGG19	10
InceptionV3+DenseNet201	10
InceptionV3+Xception	10
Xception+DenseNet201	10
Xception+VGG19	10
VGG19+DenseNet201	10

Table 5.2: Hybrid Model Combinations

combinations, emphasizing the impact of this parameter on training dynamics. The results provide insights into the efficacy of hybrid models for image classification tasks, laying the groundwork for further investigations and optimizations in model architecture and training parameters.

5.4.5 Ensemble Learning

The ensemble model approach involves combining multiple individual models, each trained on the same dataset, to generate a single prediction. This strategy aims to leverage the strengths of each model and mitigate the weaknesses of any single model. There are different strategies for combining predictions, and two common approaches are average prediction and majority voting

Average Prediction

In the average prediction method, the individual predictions from each model are averaged to produce a final prediction. This technique assumes that each model contributes equally to the overall accuracy, and it can improve performance by reducing the impact of outliers or individual model biases [63].

Majority Voting

Majority voting, on the other hand, relies on the most frequent prediction among the individual models. This approach assumes that the majority of models are more likely to provide the correct prediction, and it can be particularly effective when there is a clear consensus among the individual models [64].

These combinations suggest that the ensemble model explores the performance of different model architectures and their potential synergistic effects. By comparing the performance of different ensemble models and prediction methods, it is possible to identify the most effective approach for the specific image classification task.

Model Combination	Prediction Method
Xception + InceptionV3 + DenseNet201	Avg pred and Majority voting
Xception + InceptionV3 + VGG19	Avg pred and Majority voting
Xception + InceptionV3 + ResNet50	Avg pred and Majority voting
Xception + DenseNet201 + VGG19	Avg pred and Majority voting
Xception + InceptionV3 + DenseNet201 + VGG19	Avg pred and Majority voting
Xception + InceptionV3 + DenseNet201 + ResNet50	Avg pred and Majority voting
Xception + InceptionV3 + DenseNet201 + VGG19 + ResNet50	Avg pred and Majority voting

Table 5.3: Ensemble Model Combinations and Prediction Methods

5.4.6 Stratified K-fold Cross Validation

In the context of machine learning, stratified K-fold cross-validation is a specific variation of K-fold cross-validation that aims to address the issue of imbalanced splitting of datasets.

Stratified K-fold cross-validation is particularly useful for evaluating the performance of classification models on imbalanced datasets. By ensuring that each fold is a balanced representation of the dataset, prevents the model from being biased toward the majority class and provides a more accurate assessment of its ability to classify minority classes. It involves dividing the dataset into k folds, typically k=5 or k=10 [24]. Then, the model is trained k times, each time using a different fold as the testing set and the remaining folds as the training set. The average performance across the k folds is then used as the estimate of the model's generalization error. For our resource problem, we fold our dataset into two different types of K values K= 3 and K= 5 with random_state= 42.

5.4.7 Transformer and Big Transfer Learning

Vision Transformer (ViT) and BigTransfer (BiT), on a specific image classification task. The models were trained using the same hyperparameters and training duration, allowing for a direct comparison of their effectiveness.

This suggests that the ViT model's architecture and training approach were better suited for the specific image classification task. Both models were trained for 79 epochs, implying that they had sufficient exposure to the training data to learn the underlying patterns and features. This consistency in training duration facilitates a fair comparison of their performance. The use of identical hyperparameters for both models: a batch size of 32, a learning rate of 1e-4, the Adam optimizer, and the categorical cross-entropy loss function. This standardization of hyperparameters ensures that the difference in performance is primarily attributed to the models' architectures and training approaches [35].

The superior performance of the Vision Transformer (ViT) model suggests that its architecture and training strategy were more effective in extracting relevant information from the

image data and generalizing to unseen images. This highlights the importance of careful model design and training optimization for achieving high accuracy in image classification tasks.

5.5 Explainable Artificial Intelligent (XAI)

We implemented XAI on the models so that we can understand how our best model is working on the images. XAI can help us to identify and mitigate biases in AI models. This is important because biased AI models can lead to unfair or discriminatory outcomes. Lastly, when we understand how an AI model works, we can better troubleshoot it and debug it when it makes mistakes.

5.5.1 LIME

we used the LIME algorithm to detect the areas in which the model is focusing more than the other regions. But for most of the images, our model focuses right region.

$\Pi_x(z)$ was utilized as a proximity measure between an instance z to x, to define locality around x, for our model f: Rd $\rightarrow$ R, where f(x) is the likelihood of x to belong in any of five classes. To determine how dishonest g was in approximating f in the locality described by Π_x, the fidelity function $\Im(f,g,\Pi_x)$ was used. In order to maximize the number of interpretations, $\Omega(g)$ was reduced, along with the fidelity function. [10] This is a summary of the explanation produced by LIME.

$$\xi(x) = \arg\min_{g \in G} L(f, g, \pi_x) + \Omega(g)$$

5.5.2 Grad-CAM (XAI)

Grad-CAM, an abbreviation for Gradient-weighted Class Activation Mapping, serves as a powerful tool for interpreting the decisions of convolutional neural networks (CNNs). Unlike some interpretation methods, Grad-CAM does not require retraining the model and operates by extracting the gradients of the predicted class score with respect to the feature maps in the final convolutional layer [26]. These gradients are then used to generate a weighted combination of feature maps, creating a heatmap that visually highlights the pivotal regions in the input image for the model's decision. This approach facilitates a transparent understanding of CNN's decision-making process by emphasizing the spatial importance of certain image regions in contributing to the classification outcome.

In contrast to techniques like LIME, which uses a proximity measure and fidelity function for local interpretability, Grad-CAM provides a global perspective by identifying the most influential regions without explicitly defining a proximity metric. The resulting heatmap effectively communicates which areas of the input image are most impactful in driving the CNN's prediction for a specific class, enhancing the transparency and interpretability of deep learning models in image classification tasks.

5.6 Project Management

5.6.1 Project Schedule

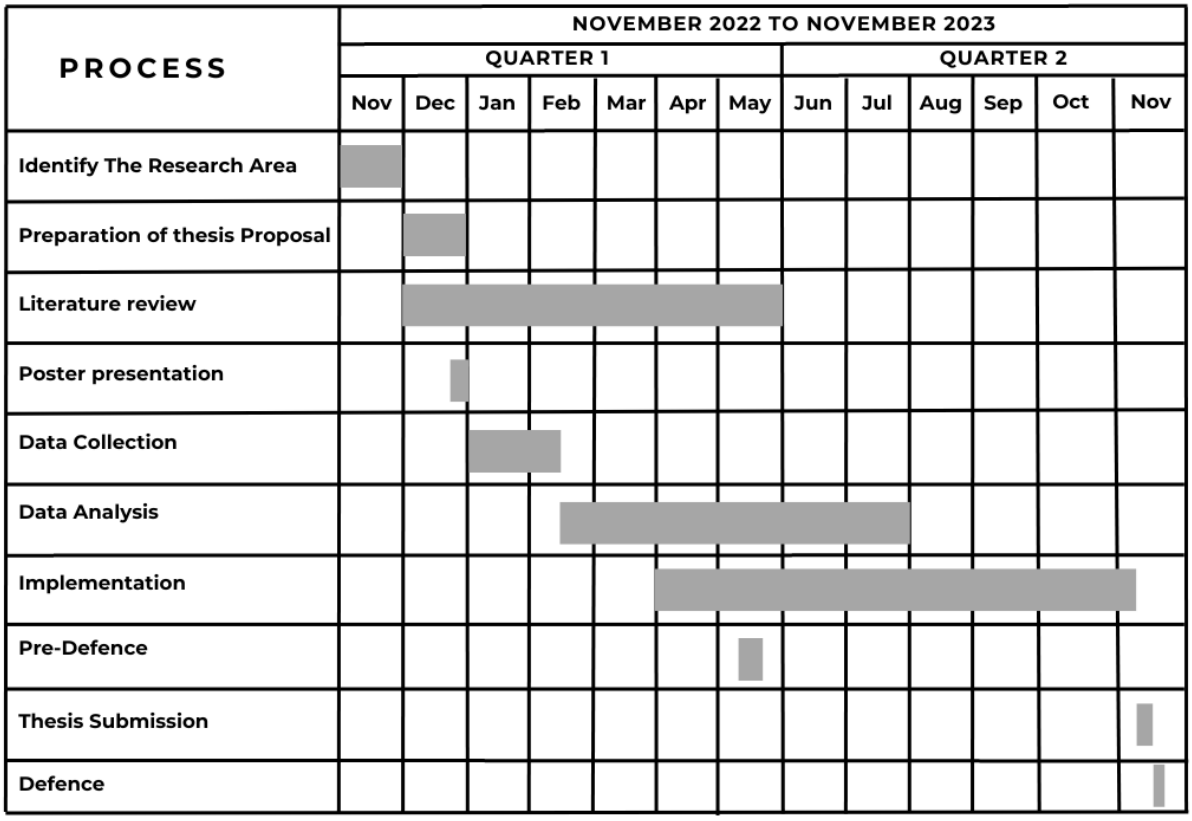


Figure 5.2: Gantt Chart of Project Management

5.6.2 Financial Management

Expense	Date	Amount(TK)	Vendor/Supplier
Computer	2022-12-04	1,50,000	Star Tech
Software	2023-09-11	2500	Turnitin, Canva Pro
Printing	2023-11-14	1500	Nilkhet
Binding	2023-11-14	800	Nilkhet

5.7 Summary of the Methodology:

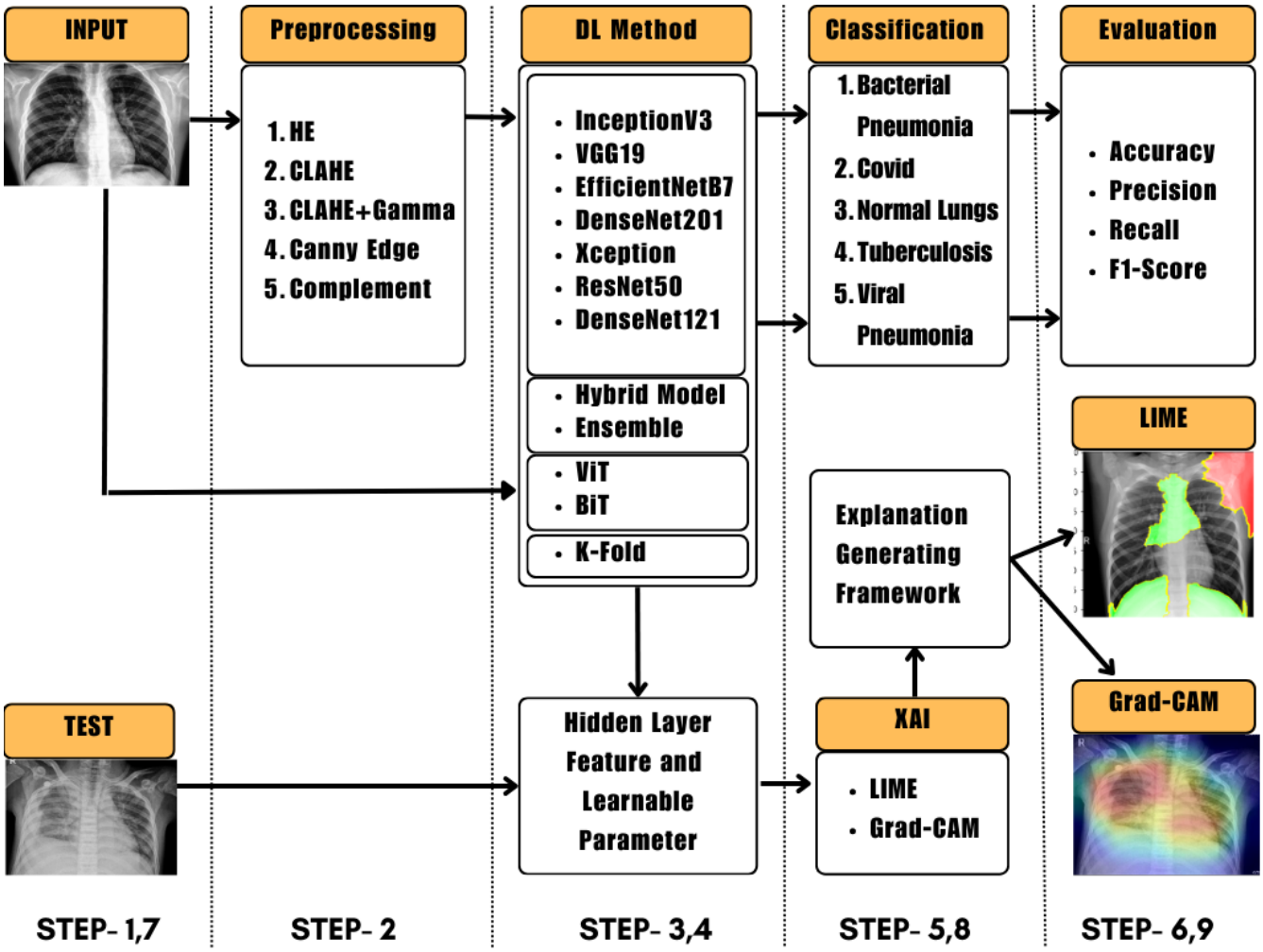


Figure 5.3: This methodology for 4 types of lung disease and normal lung identification: The whole procedure is represented in multiple steps.

STEP- 1,7: In step 1, the Images from the training dataset are presented to the proposed model batch by batch and take over to the pre-processing step. In step 7, the test Images are presented to the train model.

STEP- 2: We pre-processed each image with different image processing and augmentation techniques. Then pass to the image next step.

STEP- 3,4: This is the model learning step and it learns from the train image.

STEP- 5,8: In step 5, the model performs the classification procedure among 5 classes. In step 8, visualize the model prediction from the test data.

STEP- 6,9: Finally, evaluate the model with different types of evaluation matrix and interpreted with XAI

Chapter 6

Result Analysis

6.1 Overview

In this chapter, we will show the statistical result of our proposed model. We will represent our result with different visualization tools to make it easier to understand.

6.2 Experiment Results

In this section, we will discuss the experimental results of the proposed methodology. We applied various image enhancement techniques discussed earlier. Specifically, on histogram-equalized images, we ran both single models and hybrid models to assess if they achieved higher accuracy compared to the base dataset. Additionally, ensemble models and transfer models were executed on images enhanced by CLAHE and Gamma. We also explored alternative enhancement techniques, including CLAHE, Canny edge, and Complement.

All these enhanced images were utilized to evaluate the performance of the Xception model, allowing us to compare and identify which enhancement technique yielded the highest accuracy.

Enhancement Technique	**Model**	**Accuracy**	**Precision**	**Recall**	**F1**
HE	Xception	91	91-91	90-90	91-91
CLAHE	Xception	92	92-92	92-92	92-92
CLAHE+Gamma	Xception	93	94-93	93-93	93-93
Canny Edge	Xception	85	86-86	85-85	85-85
Complement	Xception	93	93-93	93-93	93-93

Table 6.1: Different Enhancement Techniques

6.2.1 Result by Pre-Trained CNN Model

Model Name	Precision	Recall	F1	Accuracy	After Enhancement Accuracy
Xception	93	93	93	93	91
InceptionV3	92	92	92	92	91
DenseNet121	92	92	92	92	91
DenseNet201	91	91	91	91	90
VGG19	90	90	90	90	88
EfficientNetB7	89	89	89	89	87
ResNet50	86	86	86	86	85

Table 6.2: Result obtained by single model

We conducted experiments using seven models on both the base dataset and an enhanced dataset, the latter of which was processed with histogram equalization. We assessed the models' performance by measuring accuracy, precision, recall, and F1-score.

The Xception model achieved the highest accuracy of 93% on the base dataset and 91% on the enhanced dataset. The remaining six models—InceptionV3, DenseNet201, EfficientNetB7, ResNet50, DenseNet121, and VGG-19—attained accuracies of 92%, 91%, 89%, 86%, 92%, and 90%, respectively, on the base dataset. On the enhanced dataset, their accuracies were 91%, 90%, 87%, 85%, 91%, and 88%, respectively.

Interestingly, the enhanced dataset did not consistently outperform the base dataset across all models. Detailed results for each model can be found in Table **??**.

As we can see, the Xception model demonstrated the best performance among the models we conducted; therefore, we conducted further exploration and analysis on this model.

Xception: After running 31 epochs we get 98% of training accuracy and 92% of validation accuracy on the main dataset. Test accuracy was 93% with a weighted average of 93% for precision, recall, and f1-score. We got a 5*5 confusion matrix for five classes (Bacterial pneumonia, Coronavirus disease, Normal, Tuberculosis, and Viral pneumonia).

From the figure 6.1 of the confusion matrix, we can see that the model got confused with Bacterial Pneumonia and Viral Pneumonia. It predicted 68 images of Viral Pneumonia as Bacterial Pneumonia and 46 images of Bacterial Pneumonia as Viral Pneumonia. Other than that it predicted 356, 405, 396, 404, and 315 images of Bacterial Pneumonia, Coronavirus disease, Normal, Tuberculosis, and Viral Pneumonia respectively.

It is difficult to determine a particular model as the best one because there are a lot of things that must be considered. The setting of hyperparameters affects a lot on performance of various models. We used the same hyperparameter for all the models.

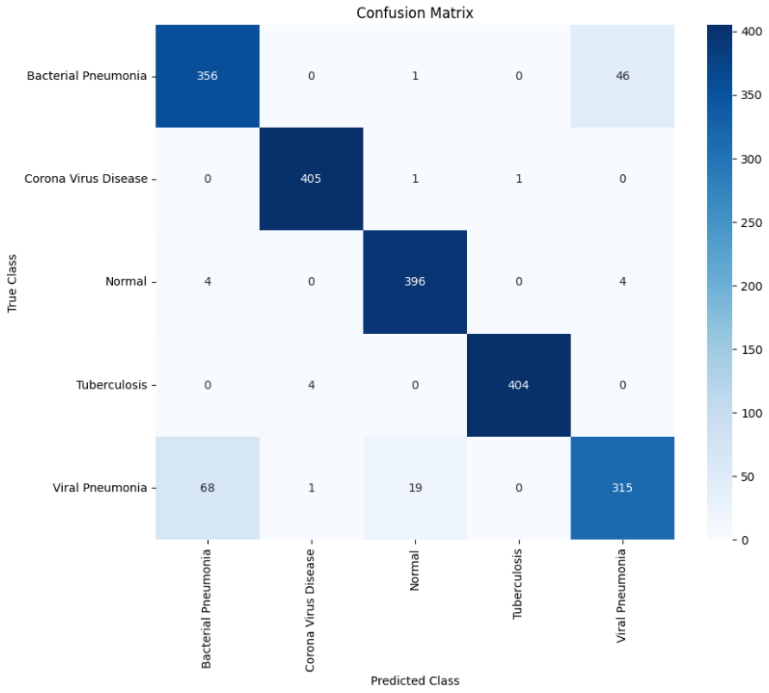


Figure 6.1: Confusion Matrix of Xception Model

6.2.2 Hyper-parameter Tuning

Optimization's Effect

Updating the model's weights during training is the optimizer's responsibility. A hyperparameter tweaking experiment revealed that Adam, with a learning rate of 1e-4, was a better option than SGD, the default optimizer for the Xception model. This shows Adam can adjust the model's weights more successfully for the given situation.

Dropout Rate's Effect

Overfitting may be avoided by using the regularization approach known as dropout. Although 0.6 was a superior option, the hyperparameter tweaking experiment revealed that the Xception model's default dropout rate of 0.5 was still valid. It seems from this that a greater dropout rate works better in avoiding overfitting in this particular activity.

Batch Size's Effects

The quantity of training cases handled simultaneously is known as the batch size. Although the hyperparameter tuning experiment indicated that 32 was the optimal value, the Xception model's default batch size is 32. This indicates that for training the model on this problem, a batch size of 32 is most effective.

The Influence of Eras

The total number of training runs the model receives on the training dataset, which is called

epochs. While the hyperparameter tuning experiment indicated that 31 was the optimal number of epochs, the Xception model's default number is thirty-one. This shows that to get decent results on this task, the model only has to be trained for 31 epochs.

Hyperparameter	Tuned Value	Before Tuning Accuracy	After Tuning Accuracy
Optimizer	Adam(learning_rate=1e-4)	90%	89%
Dropout Rate	0.6	88%	94%
Batch Size	32	86%	96%
Epochs	31	89%	93%
Overall		89%	**93%**

The experiment aimed to tune the hyperparameters, and the findings indicate that the Xception model's performance may be greatly enhanced. A excellent place to start when fine-tuning the model for a particular job is with the particular hyperparameters that were adjusted.

6.2.3 Transfer Learning and Fine Tune

Applying transfer learning and fine-tuning, we achieved the highest accuracy of 90% on the base dataset using the DenseNet121 model. However, when extended to the enhanced dataset processed with CLAHE and Gamma, it couldn't surpass the accuracy of the base dataset, falling short by 1%.

Scene	Model	Accuracy	Precision	Recall	F1
Before enhancement	DenseNet121	90	90-90	90-90	90-90
After enhancement	DenseNet121	89	90-90	89-89	89-89

Table 6.3: Result obtained by Transfer Learning and Fine Tune

6.2.4 Result by Hybrid Model

Hybrid models were created using various combinations, such as InceptionV3 with VGG19, InceptionV3 with DenseNet201, InceptionV3 with Xception, Xception with DenseNet201, Xception with VGG19, and VGG19 with DenseNet201. These models were trained on histogram equalized images. The highest accuracy, reaching 89%, was achieved by the InceptionV3 with Xception and VGG19 with DenseNet201 combinations.

Model Name	Accuracy	Precision	Recall	F1
InceptionV3+VGG19	85	85	85	85
InceptionV3+DenseNet201	88	88	88	88
InceptionV3+Xception	89	89	89	89
Xception+DenseNet201	86	86	86	86
Xception+VGG19	88	88	88	88
VGG19+DenseNet201	89	89	89	89

Table 6.4: Result obtained by hybrid model

6.2.5 Result by Vision Transformer and Big transfer Model

We also implemented Vision Transformer (ViT) and BigTransfer (BiT) models. Vision Transformer achieved an accuracy of 91%, while BigTransfer achieved an accuracy of 86%. Both models were trained on an enhanced dataset using CLAHE and Gamma, with 79 epochs and a batch size of 32.

Model Name	Accuracy	Precision	Recall	F1
Vision Transformer (ViT)	91	91	91	91
BigTransfer (BiT)	86	86	86	86

Table 6.5: Result obtained by transformer and big transfer model

6.2.6 Stratified K-fold Cross Validation

We implemented the K-fold technique to determine the best accuracy. Utilizing both 3-fold and 5-fold cross-validation, we applied this technique to both the base dataset and the enhanced dataset processed with CLAHE and Gamma. This approach yielded the highest accuracy of 96.51%.We utilized the Xception model in this K-fold analysis since it showed the best accuracy in single model evaluation.

Using 3-fold:

Before Enhancement: The model achieved an accuracy of 90% on one fold, 95% on another fold, and 98% on the third fold. The average accuracy across all three folds is 94.33%. The precision, recall, and F1 scores are all very high, ranging from 94% to 95%.

After Enhancement: The model achieved an accuracy of 90% on one fold, 97% on another fold, and 98% on the third fold. The average accuracy across all three folds is 95.00%. The precision, recall, and F1 scores are all very high, ranging from 95% to 96%.

Scene	Model	Accuracy	Precision	Recall	F1
Before enhancement	Xception	94.33	94	94	94
After enhancement	Xception	95	95	95	95

Table 6.6: Result obtained by 3-fold

3-Fold Classification Report

Class	Precision	Recall	F1-Score	Support
Bacterial Pneumonia	0.93	0.99	0.96	402
Corona Virus Disease	0.99	1.00	0.99	406
Normal	1.00	0.96	0.98	402
Tuberculosis	1.00	0.99	0.99	407
Viral Pneumonia	0.98	0.95	0.96	402
Accuracy			0.98	2019
Macro Avg	0.98	0.98	0.98	2019
Weighted Avg	0.98	0.98	0.98	2019

3-Fold Confusion Matrix

	Predicted				
Actual	**Bacterial Pneumonia**	**Corona Virus Disease**	**Normal**	**Tuberculosis**	**Viral Pneumonia**
Bacterial Pneumonia	398	1	0	0	3
Corona Virus Disease	0	406	0	0	0
Normal	10	0	385	1	6
Tuberculosis	0	5	0	402	0
Viral Pneumonia	18	0	1	0	383

Using 5-fold:

Before Enhancement: The 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 precision, recall, and F1 scores are all very high, ranging from 96% to 97%.

After Enhancement: The model achieved an accuracy of 90% on one fold, 97% 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 96.51%. The precision, recall, and F1 scores are all very high, ranging from 96% to 97%.

Scene	Model	Accuracy	Precision	Recall	F1
Before enhancement	Xception	96.20	96	96	96
After enhancement	Xception	96.51	96	96	96

Table 6.7: Result obtained by 5-fold

5-Fold Classification Report

Class	Precision	Recall	F1-Score	Support
Bacterial Pneumonia	0.99	0.96	0.97	401
Corona Virus Disease	0.99	1.00	0.99	407
Normal	0.99	1.00	0.99	402
Tuberculosis	1.00	1.00	1.00	407
Viral Pneumonia	0.97	0.98	0.97	402
Accuracy			0.99	2019
Macro Avg	0.99	0.99	0.99	2019
Weighted Avg	0.99	0.99	0.99	2019

5-Fold Confusion Matrix

	Predicted				
Actual	**Bacterial Pneumonia**	**Corona Virus Disease**	**Normal**	**Tuberculosis**	**Viral Pneumonia**
Bacterial Pneumonia	385	2	0	0	14
Corona Virus Disease	0	407	0	0	0
Normal	0	1	401	0	0
Tuberculosis	0	1	0	406	0
Viral Pneumonia	5	1	4	0	392

6.2.7 Result by Ensemble Model

Ensemble models were created using multiple combinations, employing two criteria: average prediction and majority voting. The combinations for the average prediction criterion are included. The same combinations were also tested using the majority voting criterion. However, the highest accuracy of 93% was achieved, surpassing all other combinations. These ensemble models were applied to enhanced images using CLAHE and Gamma.

Comparing the performance of individual model combinations within each strategy, it's evident that the choice of architectural combinations has a minimal impact on the overall metrics. Whether utilizing Xception, InceptionV3, Densenet201, VGG19, Resnet50, or various combinations thereof, the ensemble consistently achieves accuracies of 92% to 93%, demonstrating the effectiveness of combining diverse models for image classification.

The ensemble's stable performance across different metrics and strategies suggests that it successfully leverages the strengths of each individual model. This approach mitigates the weaknesses of individual architectures, leading to a more robust and reliable predictive

model. The ensemble's ability to generalize well to different combinations of architectures reinforces its suitability for real-world applications.

Average Prediction				
Model Name	**Accuracy**	**Precision**	**Recall**	**F1**
Xception+InceptionV3+DenseNet201	93	93	93	93
Xception+InceptionV3+VGG19	92	92	92	92
Xception+InceptionV3+ResNet50	93	93	93	93
Xception+DenseNet201+VGG19	92	92	92	92
Xception+InceptionV3+DenseNet201+VGG19	93	93	93	93
Xception+InceptionV3+DenseNet201+ResNet50	93	93	93	93
Xception+InceptionV3+DenseNet201+VGG19+ResNet50	93	93	93	93

Table 6.8: Result obtained by ensemble model using average prediction criteria

Majority Voting				
Model Name	**Accuracy**	**Precision**	**Recall**	**F1**
Xception+InceptionV3+DenseNet201	93	93	93	93
Xception+InceptionV3+VGG19	92	92	92	92
Xception+InceptionV3+ResNet50	92	92	92	92
Xception+DenseNet201+VGG19	92	92	92	92
Xception+InceptionV3+DenseNet201+VGG19	93	93	93	93
Xception+InceptionV3+DenseNet201+ResNet50	92	92	92	92
Xception+InceptionV3+DenseNet201+VGG19+ResNet50	92	92	92	92

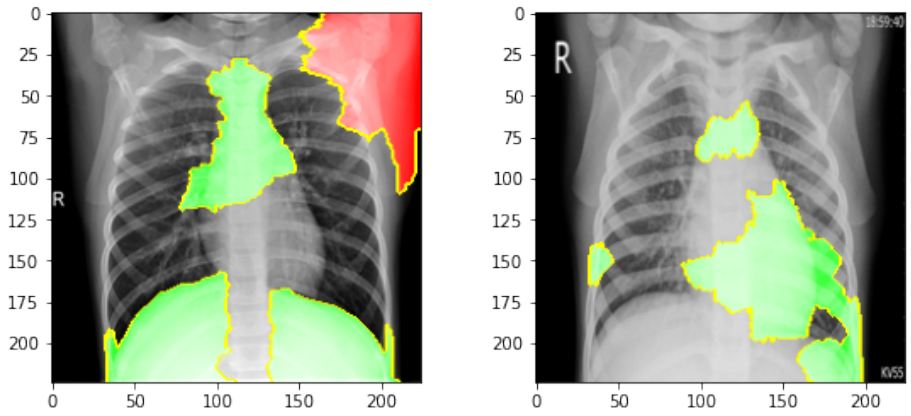
Table 6.9: Result obtained by ensemble model using majority voting criteria

6.3 Interpretability

In this section, we will discuss about the Explainable Artificial Intelligence part of our thesis.

6.3.1 Lime

We used explainable AI to find out in which region the model focuses to make decisions. We used the LIME algorithm [65] to find out regions that is focused by the model and it gave x-ray images marked by two colors to understand in which region the model predicted correctly and in which region it predicted wrong. Correct regions are marked by green color (6.2b) and wrong regions are marked by red color (6.2a).



(a) Wrongly Classified Image (b) Correctly Classified Image

Figure 6.2: Important Regions that help Inception-V3 Model prediction

6.3.2 Grad-CAM

Gradient-weighted Class Activation Mapping (Grad-CAM), uses the gradients of any target concept flowing into the final convolutional layer to produce a coarse localization map highlighting the important regions which is called the Heatmap in the image for predicting the concept.

The method's accuracy and adaptability are improvements above earlier methods. Although it is intricate, fortunately, the result is clear. We wish to generate a Grad-CAM heat map. The fully connected layers are attached for prediction. The input is then sent through the model, and the layer output and loss are obtained. Next, we determine the model loss vs. the gradient of our chosen model layer's output. Next, we choose portions of the gradient that help in the prediction, decrease, resize, and rescale it, and finally overlay the heat map with the original picture.

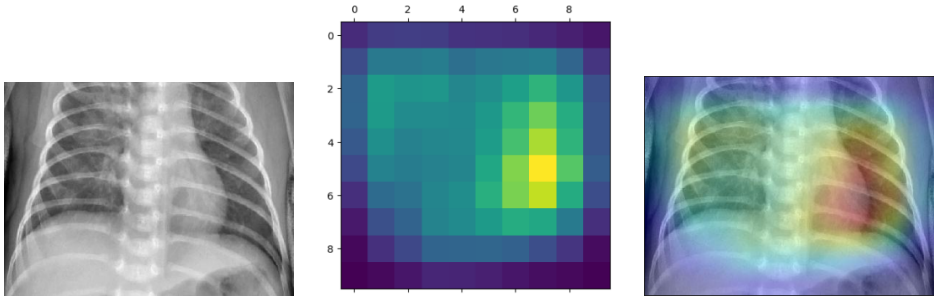


Figure 6.3: Wrongly Classified Image by Xception Model prediction

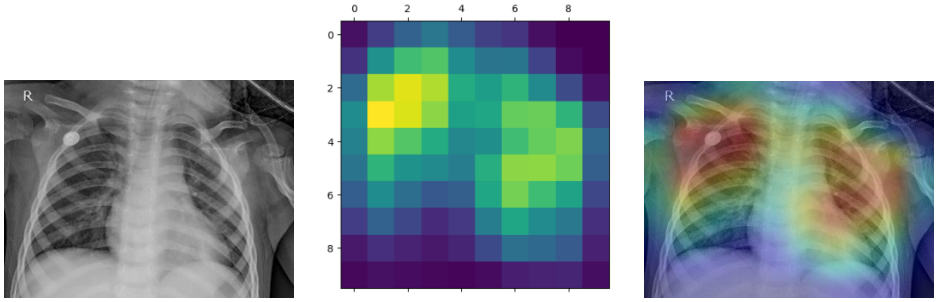


Figure 6.4: Correctly Classified Image by Xception Model prediction

6.4 Summary

We explored various recent popular approaches to optimize results, including Pre-Trained CNN models, hybrid models, ensemble models, transformers, big transfer models, stratified k-fold cross-validation, hyperparameter tuning, and transfer learning with fine-tuning. Among these methods, the k-fold cross-validation technique yielded the highest accuracy of 96.51%. Additionally, we employed diverse image enhancement techniques— HE, CLAHE, CLAHE and Gamma, Canny Edge, Complement—to enhance the quality of X-ray images, thereby contributing to improved accuracy. The Lime and Grad-CAM XAI approaches provided insightful explanations on the decision-making processes of these models, reinforcing trust in their reliability.

Chapter 7

Conclusion

7.1 Conclusion

In conclusion, our study marks a big advance in the area of medical diagnostics, notably in the automated diagnosis of different lung disorders using Chest X-Ray pictures. The comprehensive aspect of our approach is obvious in the suggestion and rigorous assessment of several categorization models, along with a detailed examination of model behavior via the perspective of explainable AI tools, notably the LIME algorithm and Grad-CAM.

The meticulous classification of our dataset into five distinct lung disease categories, encompassing Bacterial Pneumonia, Corona Virus Disease, Tuberculosis, Viral Pneumonia, and a Normal class, underscores the thoroughness and relevance of our methodology in addressing real-world medical challenges. The combination of two datasets—the primary dataset and the improved dataset—has not only widened the area of our inquiry but has also allowed the creation of picture improvement algorithms customized to the particular properties of our dataset. Notably, the inclusion of CLAHE and GAMMA approaches emerged as crucial advancements, considerably boosting the predictive capabilities of our models.

Following rigorous Stratified 5-fold cross-validation on the upgraded photos, the lone Xception model emerged as the apex of our efforts, obtaining a phenomenal accuracy rate of 96.51%. This finding is suggestive of the usefulness of our entire technique in automating and improving the diagnosis of lung disorders, offering a paradigm change towards quicker and more efficient diagnostic procedures in contrast to old manual methods.

Beyond the academic world, our study carries profound implications for healthcare practices. The automation of illness detection, assisted by our rigorously refined models, offers the potential to change clinical operations, offering speedy, accurate, and resource-efficient diagnoses. As we traverse the growing terrain of AI applications in healthcare, our research serves as a cornerstone, exemplifying the transformational ability of technology in boosting

patient outcomes and the overall efficiency of medical systems.

Looking ahead, the insights gathered from this study not only add to the continuing conversation on the integration of artificial intelligence into healthcare but also establish the platform for future development and refining of AI-based diagnostic tools. This trend leads towards a future where the synergy between artificial intelligence and healthcare results in heightened accuracy, accessibility, and efficacy in the identification and treatment of numerous illnesses, eventually ushering in a new era of healthcare excellence.

So, we proposed several models to classify different lung diseases from Chest X-Ray images and we also analyzed how the proposed models behave while predicting using explainable AI (LIME Algorithm and Grad-CAM). We classified the data set into five classes of lung diseases namely Bacterial Pneumonia, Corona Virus Disease, Tuberculosis, Viral Pneumonia, and a Normal labeled class for disease-free lung. Basically, we try to do all the things based on two different types of datasets one is the main dataset and another is the enhanced dataset. Enhanced dataset means we try to enhance the lung images into the technique that is suitable for our dataset. Finally, we tried different techniques but applying the CLAHE and GAMMA techniques gave better results than others also Complement technique gave the same result as before. After Stratified 5-k fold cross-validation on enhanced images, the single Xception model gave us the highest accuracy of 96.51% among the proposed models. Our project will enable us to make an automated approach to detect lung diseases in a shorter time than the manual way.

7.2 Future Work

To improve the overall accuracy and robustness of illness detection algorithms, future research should investigate the integration of different imaging modalities, such as integrating CT or MRI scans with pictures from chest X-rays. A more thorough knowledge of pulmonary pathophysiology may be possible with this multimodal approach.

Moreover, we will try more hybrid models [66] by combining models together and Capsule Network. We have implemented LIME and Grad CAM algorithms for explainable AI. The output images show that the models focus on the lung area, decreasing the accuracy value. To solve the problem, we will do image segmentation [67] so that the model has to focus only on the lung area. We will implement SHAP and different algorithms to analyze the predictions of the models further.

Finally, the dataset might be improved and expanded continuously to support a wider range of unique and uncommon scenarios. This further work would improve the models' capacity to adapt to a greater range of clinical situations and differences in illness presentation.

References

[1] https://uniph.go.ug/trends-and-spatial-distribution-\
of-pneumonia-admissions-and-deaths-among-children-\
under-five-years-in-uganda-2013-2021, Accessed:2023-11-12.

[2] G. B. Migliori, D. Visca, M. van den Boom, S. Tiberi, D. R. Silva, R. Centis, L. D'Ambrosio, T. Thomas, E. Pontali, L. Saderi, H. S. Schaaf, and G. Sotgiu, "Tuberculosis, covid-19 and hospital admission: Consensus on pros and cons based on a review of the evidence," *Pulmonology*, vol. 27, no. 3, pp. 248–256, 2021.

[3] M. Stewart, "Simple introduction to convolutional neural networks," *Towards Data Science*, vol. 27, 2019.

[4] D. Liu, "A practical guide to relu," *Retrieved from medium. com.(Nov. 2017): https://medium. com/tinymind/a-practical-guide-to-relu-b83ca804f1f7*, 2017.

[5] M. Stewart, "Simple introduction to convolutional neural networks," *Towards Data Science*, vol. 27, 2019.

[6] S. Mishra, B. L. Sturm, and S. Dixon, "Local interpretable model-agnostic explanations for music content analysis.," in *ISMIR*, vol. 53, pp. 537–543, 2017.

[7] J. Kolluri, V. K. Kotte, M. Phridviraj, and S. Razia, "Reducing overfitting problem in machine learning using novel l1/4 regularization method," in *2020 4th International Conference on Trends in Electronics and Informatics (ICOEI)(48184)*, pp. 934–938, IEEE, 2020.

[8] J. Mohajon, "Confusion matrix for your multi-class machine learning model," *Medium. Available online: https://towardsdatascience. com/confusion-matrix-for-your-multi-class-machine-learning-model-ff9aa3bf7826(accessed on 24 July 2021)*, 2020.

[9] I. H. Sarker, "Deep learning: a comprehensive overview on techniques, taxonomy, applications and research directions," *SN Computer Science*, vol. 2, no. 6, p. 420, 2021.

[10] M. Bhandari, T. B. Shahi, B. Siku, and A. Neupane, "Explanatory classification of cxr images into covid-19, pneumonia and tuberculosis using deep learning and xai," *Computers in Biology and Medicine*, vol. 150, p. 106156, 2022.

[11] M. Stewart, "Simple introduction to convolutional neural networks," *Towards Data Science*, vol. 27, 2019.

[12] E. Beauxis-Aussalet and L. Hardman, "Visualization of confusion matrix for non-expert users," in *IEEE Conference on Visual Analytics Science and Technology (VAST)-Poster Proceedings*, pp. 1–2, 2014.

[13] https://keras.io/api/applications/ Accessed:2023-05-15.

[14] M. Ciotti, M. Ciccozzi, A. Terrinoni, W.-C. Jiang, C.-B. Wang, and S. Bernardini, "The covid-19 pandemic," *Critical reviews in clinical laboratory sciences*, vol. 57, no. 6, pp. 365–388, 2020.

[15] https://ourworldindata.org/pneumonia, Accessed:2023-11-12.

[16] https://www.who.int/news-room/fact-sheets/detail/tuberculosis#:~:text=Key%20facts,with%20tuberculosis%20(TB)%20worldwide., Accessed:2023-11-12.

[17] S. B. Shuvo, S. N. Ali, S. I. Swapnil, T. Hasan, and M. I. H. Bhuiyan, "A lightweight cnn model for detecting respiratory diseases from lung auscultation sounds using emd-cwt-based hybrid scalogram," *IEEE Journal of Biomedical and Health Informatics*, vol. 25, no. 7, pp. 2595–2603, 2020.

[18] M. Anthimopoulos, S. Christodoulidis, L. Ebner, A. Christe, and S. Mougiakakou, "Lung pattern classification for interstitial lung diseases using a deep convolutional neural network," *IEEE transactions on medical imaging*, vol. 35, no. 5, pp. 1207–1216, 2016.

[19] A. Holzinger, C. Biemann, C. Pattichis, and D. Kell, "What do we need to build explainable ai systems for the medical domain?," 12 2017.

[20] G. Montavon, W. Samek, and K.-R. Müller, "Methods for interpreting and understanding deep neural networks," *Digital Signal Processing*, vol. 73, pp. 1–15, 2018.

[21] C. Shorten and T. M. Khoshgoftaar, "A survey on image data augmentation for deep learning," *Journal of big data*, vol. 6, no. 1, pp. 1–48, 2019.

[22] C. Elkan, "The foundations of cost-sensitive learning," in *International joint conference on artificial intelligence*, vol. 17, pp. 973–978, Lawrence Erlbaum Associates Ltd, 2001.

[23] A. B. Arrieta, N. Díaz-Rodríguez, J. Del Ser, A. Bennetot, S. Tabik, A. Barbado, S. García, S. Gil-López, D. Molina, R. Benjamins, *et al.*, "Explainable artificial intelligence (xai): Concepts, taxonomies, opportunities and challenges toward responsible ai," *Information fusion*, vol. 58, pp. 82–115, 2020.

[24] T.-T. Wong and P.-Y. Yeh, "Reliable accuracy estimates from k-fold cross validation," *IEEE Transactions on Knowledge and Data Engineering*, vol. 32, no. 8, pp. 1586–1594, 2019.

[25] D. Wei, B. Zhou, A. Torrabla, and W. Freeman, "Understanding intra-class knowledge inside cnn," *arXiv preprint arXiv:1507.02379*, 2015.

[26] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra, "Grad-cam: Visual explanations from deep networks via gradient-based localization," in *Proceedings of the IEEE international conference on computer vision*, pp. 618–626, 2017.

[27] J. Dieber and S. Kirrane, "Why model why? assessing the strengths and limitations of lime," *arXiv preprint arXiv:2012.00093*, 2020.

[28] N. Bačanin Džakula *et al.*, "Convolutional neural network layers and architectures," in *Sinteza 2019-International Scientific Conference on Information Technology and Data Related Research*, pp. 445–451, Singidunum University, 2019.

[29] K. O'Shea and R. Nash, "An introduction to convolutional neural networks," 2015.

[30] J. Koushik, "Understanding convolutional neural networks," *arXiv preprint arXiv:1605.09081*, 2016.

[31] M. M. Taye, "Theoretical understanding of convolutional neural network: concepts, architectures, applications, future directions," *Computation*, vol. 11, no. 3, p. 52, 2023.

[32] M. Y. Teow, "Understanding convolutional neural networks using a minimal model for handwritten digit recognition," in *2017 IEEE 2nd International Conference on Automatic Control and Intelligent Systems (I2CACIS)*, pp. 167–172, IEEE, 2017.

[33] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *nature*, vol. 521, no. 7553, pp. 436–444, 2015.

[34] F. Siddique, S. Sakib, and M. A. B. Siddique, "Recognition of handwritten digit using convolutional neural network in python with tensorflow and comparison of performance for various hidden layers," in *2019 5th international conference on advances in electrical engineering (ICAEE)*, pp. 541–546, IEEE, 2019.

[35] R. Xiong, Y. Yang, D. He, K. Zheng, S. Zheng, C. Xing, H. Zhang, Y. Lan, L. Wang, and T. Liu, "On layer normalization in the transformer architecture," in *International Conference on Machine Learning*, pp. 10524–10533, PMLR, 2020.

[36] D.-y. Ge, X.-f. Yao, W.-j. Xiang, X.-j. Wen, and E.-c. Liu, "Design of high accuracy detector for mnist handwritten digit recognition based on convolutional neural network," in *2019 12th International Conference on Intelligent Computation Technology and Automation (ICICTA)*, pp. 658–662, IEEE, 2019.

[37] S. Albawi, T. A. Mohammed, and S. Al-Zawi, "Understanding of a convolutional neural network," in *2017 international conference on engineering and technology (ICET)*, pp. 1–6, Ieee, 2017.

[38] T. Wood, "What is the softmax function," *DeepAI, deepai. org, dostupno na https://deepai. org/machinelearning-glossary-and-terms/softmax-layer [10.7. 2020.]*, 2019.

[39] D. C. Psichogios and L. H. Ungar, "A hybrid neural network-first principles approach to process modeling," *AIChE Journal*, vol. 38, no. 10, pp. 1499–1511, 1992.

[40] K. Dutta, P. Krishnan, M. Mathew, and C. Jawahar, "Improving cnn-rnn hybrid networks for handwriting recognition," in *2018 16th international conference on frontiers in handwriting recognition (ICFHR)*, pp. 80–85, IEEE, 2018.

[41] B. Jena, S. Saxena, G. K. Nayak, L. Saba, N. Sharma, and J. S. Suri, "Artificial intelligence-based hybrid deep learning models for image classification: The first narrative review," *Computers in Biology and Medicine*, vol. 137, p. 104803, 2021.

[42] O. Sagi and L. Rokach, "Ensemble learning: A survey," *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, vol. 8, no. 4, p. e1249, 2018.

[43] X. Zhai, A. Kolesnikov, N. Houlsby, and L. Beyer, "Scaling vision transformers," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 12104–12113, 2022.

[44] B. H. Van der Velden, H. J. Kuijf, K. G. Gilhuijs, and M. A. Viergever, "Explainable artificial intelligence (xai) in deep learning-based medical image analysis," *Medical Image Analysis*, p. 102470, 2022.

[45] https://www.kaggle.com/datasets/omkarmanohardalvi/lungs-disease-dataset-4-types, Accessed:2023-05-15.

[46] M. T. Ribeiro, S. Singh, and C. Guestrin, ""why should i trust you?" explaining the predictions of any classifier," in *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*, pp. 1135–1144, 2016.

[47] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: a simple way to prevent neural networks from overfitting," *The journal of machine learning research*, vol. 15, no. 1, pp. 1929–1958, 2014.

[48] C. Liu, Y. Cao, M. Alcantara, B. Liu, M. Brunette, J. Peinado, and W. Curioso, "Tx-cnn: Detecting tuberculosis in chest x-ray images using convolutional neural network," in *2017 IEEE international conference on image processing (ICIP)*, pp. 2314–2318, IEEE, 2017.

[49] M. Rahimzadeh and A. Attar, "A modified deep convolutional neural network for detecting covid-19 and pneumonia from chest x-ray images based on the concatenation of xception and resnet50v2," *Informatics in medicine unlocked*, vol. 19, p. 100360, 2020.

[50] N. N. Qaqos and O. S. Kareem, "Covid-19 diagnosis from chest x-ray images using deep learning approach," in *2020 international conference on advanced science and engineering (ICOASE)*, pp. 110–116, IEEE, 2020.

[51] A. I. Khan, J. L. Shah, and M. M. Bhat, "Coronet: A deep neural network for detection and diagnosis of covid-19 from chest x-ray images," *Computer methods and programs in biomedicine*, vol. 196, p. 105581, 2020.

[52] C. Sitaula and M. B. Hossain, "Attention-based vgg-16 model for covid-19 chest x-ray image classification," *Applied Intelligence*, vol. 51, pp. 2850–2863, 2021.

[53] K. I. K. S. S. K. . M. V. Shastri, S., "Cheximagenet: a novel architecture for accurate classification of covid-19 with chest x-ray digital images using deep convolutional neural networks," *Health and Technology*, vol. 12, p. 193–204, 2022.

[54] A. H. Al-Timemy, R. N. Khushaba, Z. M. Mosa, and J. Escudero, "An efficient mixture of deep and machine learning models for covid-19 and tuberculosis detection using x-ray images in resource limited settings," *Artificial Intelligence for COVID-19*, pp. 77–100, 2021.

[55] S. prasad Koyyada and T. P. Singh, "An explainable artificial intelligence model for identifying local indicators and detecting lung disease from chest x-ray images," *Healthcare Analytics*, p. 100206, 2023.

[56] O. Sarkar, M. R. Islam, M. K. Syfullah, M. T. Islam, M. F. Ahamed, M. Ahsan, and J. Haider, "Multi-scale cnn: An explainable ai-integrated unique deep learning framework for lung-affected disease classification," *Technologies*, vol. 11, no. 5, p. 134, 2023.

[57] http://www.visipics.info/index.php?title=Main_Page, Accessed:2023-05-15.

[58] A. T. Prihatno, N. Suryanto, S. Oh, T.-T.-H. Le, H. Kim, *et al.*, "Nft image plagiarism check using efficientnet-based deep neural network with triplet semi-hard loss," *Applied Sciences*, vol. 13, no. 5, p. 3072, 2023.

[59] S. Patel, K. Bharath, S. Balaji, and R. K. Muthu, "Comparative study on histogram equalization techniques for medical image enhancement," in *Soft Computing for Problem Solving: SocProS 2018, Volume 1*, pp. 657–669, Springer, 2020.

[60] K. Koonsanit, S. Thongvigitmanee, N. Pongnapang, and P. Thajchayapong, "Image enhancement on digital x-ray images using n-clahe," in *2017 10th Biomedical engineering international conference (BMEICON)*, pp. 1–4, IEEE, 2017.

[61] G. Omarova, Z. Aitkozha, Z. Sadirmekova, G. Zhidekulova, D. Kazimova, R. Muratkhan, A. Takuadina, and D. Abdykeshova, "Devising a methodology for x-ray image contrast enhancement by combining clahe and gamma correction.," *Eastern-European Journal of Enterprise Technologies*, vol. 117, no. 2, 2022.

[62] Y. Guo, H. Shi, A. Kumar, K. Grauman, T. Rosing, and R. Feris, "Spottune: transfer learning through adaptive fine-tuning," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 4805–4814, 2019.

[63] I. H. Laradji, M. Alshayeb, and L. Ghouti, "Software defect prediction using ensemble learning on selected features," *Information and Software Technology*, vol. 58, pp. 388–402, 2015.

[64] R. Polikar, "Ensemble learning," *Ensemble machine learning: Methods and applications*, pp. 1–34, 2012.

[65] M. T. Ribeiro, S. Singh, and C. Guestrin, ""' why should i trust you?" explaining the predictions of any classifier," in *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*, pp. 1135–1144, 2016.

[66] C. N. Dang, M. N. Moreno-García, and F. De la Prieta, "Hybrid deep learning models for sentiment analysis," *Complexity*, vol. 2021, pp. 1–16, 2021.

[67] S. Minaee, Y. Boykov, F. Porikli, A. Plaza, N. Kehtarnavaz, and D. Terzopoulos, "Image segmentation using deep learning: A survey," *IEEE transactions on pattern analysis and machine intelligence*, vol. 44, no. 7, pp. 3523–3542, 2021.

Generated using Undegraduate Thesis LaTeX Template, Version 1.4. Department of Computer Science and Engineering, Ahsanullah University of Science and Technology, Dhaka, Bangladesh.

This thesis was generated on Monday 20th November, 2023 at 4:44pm.