Building Trust with Explainable AI in Lung Disease Detection Using Deep Learning

Submitted By

JARIN SULTANA

Batch: 35 (A)

ID: 212-35-742

Department of Software Engineering

Daffodil International University

Supervised By

Mr. Khalid Been Md. Badruzzaman Biplob

Lecturer (Senior Scale)

Department of Software Engineering

Daffodil International University

Thesis submitted in fulfillment of the requirements for the award of the degree of Bachelor of Science

APPROVAL

This thesis titled on "**Building Trust with Explainable AI in Lung Disease Detection Using Deep Learning**", submitted by **JARIN SULTANA (ID: 212-35-742)** to the Department of Software Engineering, Daffodil International University has been accepted as satisfactory for the partial fulfillment of the requirements for the degree of Bachelor of Science in Software Engineering and approval as to its style and contents.

BOARD OF EXAMINERS

____________________________ **Chairman**

Dr. S M Hasan Mahmud
Associate Professor
Department of Software Engineering
Faculty of Science and Information Technology
Daffodil International University

____________________________ **Internal Examiner 1**

Tapushe Rabaya Toma
Assistant Professor
Department of Software Engineering
Faculty of Science and Information Technology
Daffodil International University

____________________________ **Internal Examiner 2**

Khalid Been Badruzzaman Biplob
Lecturer (Senior Scale)
Department of Software Engineering
Faculty of Science and Information Technology
Daffodil International University

____________________________ **External Examiner**

Dr. Md. Sazzadur Rahman
Professor
Institute of Information Technology
Jahangirnagar University

DECLARATION

I hereby declare that; this project has been done by me under the supervision of **Mr.Khalid Been Md.Badruzzaman Biplob, Department of Software Engineering, Faculty of Science and Information Technology, Daffodil International University.** I also declare that neither this project nor any part of this project has been submitted elsewhere for the award of any degree or diploma.

Jarin Sultana

JARIN SULTANA
ID: 212-35-742
Department of Software Engineering
Daffodil International University

Certified by

Mr.Khalid Been Md.Badruzzaman Biplob
Lecturer (Senior Scale)
Department of Software Engineering
Faculty of Science and Information Technology
Daffodil International University

ACKNOWLEDGEMENT

First and foremost, I am grateful to Almighty Allah, who has given me the strength, wisdom, and perseverance to complete this research. All through my academic journey, I am grateful for the unconditional love, support, and encouragement of my parents. It is always their belief in me that has motivated and inspired me the most.

I would like to thank my supervisor, lecturer Mr.Khalid Been Md.Badruzzaman Biplob for his valuable advice, support, and guidance throughout the research. His knowledge and insight have highly affected this work. The departmental head, Dr. Imran Mahmud, is also highly appreciated for his support, guidance, and valuable comments which helped me to successfully complete my journey. Finally, I would like to thank all my friends and all those who helped and encouraged me during this process.

ABSTRACT

The primary aim of this research is to design an XAI system for lung disease diagnosis that achieves the highest degree of accuracy possible in a more user-friendly manner than current systems. The EfficientB5 Network model is proposed for implementation as the main architecture. It has got the promise to achieve high accuracy rates up to 99.13%, but with more confidence, it could classify three types of lung diagnoses: Benign, Malignant, and Normal. For the augmentation of trust and usability in clinical practice, Grad-CAM visualization techniques based on Explainable AI will be utilized. The methodology highlights the class-specific probabilities and regional contributions as interpretable justification from the patient's or clinician's viewpoint for understanding the prediction outcome. While lung cancer remains the most lethal cancer among all cancers, it had 2.5 million cases worldwide in 2022, with more than 1.8 million deaths just in that year. The datasets used in this research are public and hence no restrictions are exercised on the utilization of the data. Validation has thus far been carried out comparing EfficientNetB5 against another pre-trained models, InceptionV3, ResNet50 and Vision Transformer. In particular, all pre-trained models were subjected to extensive experimentation to demonstrate the need for achieving a combination of explain ability plus technical accuracy ensuring trust in AI systems for a patient-centered healthcare, and thereby pushing naturally to become a new reality with this medical season.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1
INTRODUCTION

1.1 INTRODUCTION

The development and spread of aberrant cells throughout the body is cancer. Cancer is becoming one of the most frequent causes that lead to deaths around the world. According to the latest statistics of global cancer statistics (GLOBOCAN), there are 19.3 million new cancer cases around the world. In 2020, 9.96 million people died from cancer. It leads to malignant or tumor growth in the body. Lung cancer is the most commonly diagnosed cancer and the leading cause of death of men and women globally. Bad behaviours like smoking or drinking may bring that on or be inherited and immune system-related [3]. Other variables that lead to lung cancer include contamination of the environmental exposure, air pollution, mutations, and single nucleotide polymorphisms (SNPs). It causes a breathing problem in the areas of inhalation and exhalation in the chest [6]. Compared to other types of cancer, the rates of illness and death from lung cancer are gradually increasing [7].

The human body has over 30 trillion cells. When these cells get damaged, the body sends signals for them to grow. If the body ignores signals that tell cells to stop dividing—through programmed cell death—abnormal cells keep multiplying. This can lead to the formation of squamous cells that develop into cancerous tumors [11]. Nearly any part of the human body can develop cancer, and tumors can be classified as malignant or non (benign). Malignant tumors, also known as cancerous tumors, can spread to surrounding new tumors [16].

This disease develops very quickly; 49%-53% of the cancer cells are detected in stage 4, which is the last advanced stage of the disease, where the percentage of their survival does not exceed 6%. Patients with lung cancer can increase their chances of survival by 82% if they are diagnosed early with stage 1 non-small cell lung cancer [23].

Optimal outcomes for therapy depend greatly on early identification. Lung nodules, which may be an indication of lung cancer. An vast amount of computed tomography (CT) scan image data for the lungs could help detect lung cancer [18]. Machine learning and deep learning algorithms can utilize these images to improve cancer prediction and diagnosis as early as possible and find the best treatment strategies. Machine learning, predictions are based on prior data [13]. A subset of machine learning, known as deep learning, uses multiple layers of simple computational parts. In recent years, deep learning has become a top tool for detecting objects and for medical image analysis [20]. This study solves the challenge of automatic classification and prediction of CT data to eminent between patients with Benign cases, Malignant cases and Healthy subjects.

We will use different deep learning models such as InceptionV3, RseNet50, EfficientNetB5 and Vision Transformer to train our dataset that will be taken from Kaggle and compare the findings to other previous research results, trying to achieve more accurate and valuable results. Before all of that, we will apply image enhancement techniques. Pre-processing methodologies such as augmentation will be used to improve the generalization of the dataset. Once trained, the model can take an image and predict which class it belongs to. This study implements Grad-CAM++, a technique used to understand which parts of an image influence a deep learning model's decision the most. It works by capturing the activations and gradients from a specific layer inside the model during a forward and backward pass.

The rest of this article is organized as follows. Section 2 covers the literature survey review. Section 3 focuses on the analysis of the proposed method. Section 4 presents the results and discussions, while Section 5 concludes the article.

1.2 BACKGROUND

Artificial Intelligence (AI) and deep learning become important tools in medical image analysis. Traditional diagnostic methods like CT scans perform manual interpretation, are often time-consuming and can occur human error. Development of deep learning-based computer-aided diagnosis (CAD) systems has overcome these challenges and shown progress in improving accuracy and detection of lung cancer effectively.

Models such as VGG16, ResNet50, EfficientNetB5 and InceptionV3 have achieved strong performance in classifying lung nodules. The use of Explainable Artificial Intelligence (XAI) techniques, such as Grad-CAM, can provide visual explanations of model predictions. The goal of this research is to improve classification accuracy and ensure the reliable prediction that can be supportive for clinical decision making.

1.3 MOTIVATION

Lung cancer actually is a malignant tumor that causes uncontrolled cellular growth inside lung tissue. If left untreated, this tumor expands throughout various tissues. Mostly half of patients learnt their diagnosis at the last stage. At that time, the survival probability got below about 6 percent. Patients who receive their non-small cell lung cancer diagnosis at stage 1 have a better chance of surviving, about 82 percent.

Existing AI-based systems can achieve high accuracy, but they often lack interpretability. This gap between accuracy and explainability inspired me to work on this research. The motivation behind this research is developing deep learning models and integrating Explainable AI that can achieve high diagnostic accuracy and clear visual explanations for predictions. This will help to reduce misdiagnosis and support faster decision-making.

1.4 PROBLEM STATEMENT

The expansion of medical imaging technology has failed to sort out the extended manual diagnostic action that can generate human diagnostic errors. The automatic classification of CT scan data to individualize between benign, malignant patients and healthy individuals continues to be a raise an object. Researchers have developed several deep learning methods to solve this problem, but they face challenges in acquire better model generalization and accuracy.

The favorable results from deep learning models in previous research become limited when there is plentiful data annotation. When the model workings are not sensible, that can restrict their use in clinical practice. Researchers should create such models that can accomplish high accuracy. It should have interpretability to automatically detect lung cancer in CT images.

1.5 RESEARCH SCOPE

The research examines methods for lung cancer detection and classification through CT image analysis. The research focuses some important facts; these are:

- The research relies on publicly available Kaggle datasets as its data source.
- The research put in InceptionV3, ResNet50, EfficientNetB5 deep learning architectures and Vision Transformer for image classification tasks.
- The study applies three enhancement CNN models together with enlargement techniques to correctly predict training results.
- The method assembles images into three groups, which are formed with benign cases, malignant cases and healthy cases.
- The research measures performance by combining metric evaluation with visual illustration methods through Grad-CAM++.

1.6 THESIS ORGANIZATION

The 1st section of the paper is the research background, motivation and scope details of this research. It shows different research problems and their objectives. This introduction section focuses on the detection of lung cancer using image processing. Also, about how the deep learning process helps with it. Our research has some additional sectors.

The second chapter literature review will be a summary of previous research work. Also, the relation to detecting lung cancer and the process of research. It also provides for missing research gaps and places my research in comparison with other researchers. It is mainly based on general knowledge of their findings.

The third chapter will present the methodology of my research. In this, my paper describes the steps of data collection and data pre-processing techniques also work analysis. This section recaptures how the above-discussed methods and models operate operationally.

The fourth chapter's methodology findings will discuss various accuracy measures to evaluate various types of findings. Analysis of this paper becomes easy when we inquire about the accuracy of lung cancer detection and classification results by using deep learning techniques.

The final chapter is the conclusion. The abstract of my findings will be presented with a detailed explanation of my work. Here, I've talked about the work I'll perform going forward to improve the work.

1.7 SUMMARY

The research employs deep learning methods that can identify lung cancer with the help of traditional CT scan image processing techniques.
The research mainly focuses on the gravity of lung cancer and its necessity to diagnose with a research of diagnostic accuracy by deep learning.

The research study measures that earlier studies established with existing issues can improve precision and result interpretation clarity. Training with different models will improve existing problems through a combination of advanced frameworks with the help of visualization methods.

CHAPTER 2

LITERATURE REVIEW

2.1 INTRODUCTION

There are multiple cancer-detection and DL models present in the literature, each classification model having its paper. This chapter reviews various papers. With the help of this analysis, the issue of cancer detection and classification can develop a new framework. For predicting lung cancer in the early stage, growing research has used deep learning models. In this chapter, different authors' research works are highlighted.

2.2 PREVIOUS LITERATURE REVIEW

In this paper [3], Shariff et al. provide various deep learning models and their effectiveness in lung cancer detection. It highlights models such as VGG16, CNN, and SVM to improve accuracy through pre-processing techniques. VGG16 achieves an accuracy of 96.07% in Mammogram detection. This review paper demonstrated effective classification of CT images using SVM by focusing on tumor segmentation for better diagnosis. The study shows the importance of pre-processing techniques like edge identification and segmentation, which are important to improving the quality of input images and enhancing diagnostic accuracy.

This paper [6], Sudhir et al., describes the early stage of lung cancer identification. Stages were categorized using an intelligent deep learning algorithm. The paper shows that convolutional neural network techniques of the deep learning algorithm are most effective for medical image processing. LIDC/IDRI data sets are used in the research. They use DICOM software and a CNN model. This paper works with methods such as Random Forest, Decision Tree, Logistic Regression, Naive Bayes, SVM, and proposed IDLA. Among them proposal to perform a high accuracy of 92.81%.

This paper [7], Q. Shatnawi et al., focuses on the automatic classification and prediction of lung cancer using CT scans. They are employing the deep learning strategies (DL), especially enhanced Conventional Newton Networks, to enable accurate image analysis. Here in this research, they trained several models. They have used deep learning models, ConvNextSmall, VGG16, ResNet50, InceptionV3, and EfficientNetB0 for lung cancer classification. The VGG16 has an accuracy rate highest 99%. The dataset is mainly divided into four classes. 338 images were a part of the adenocarcinoma section, 187 images were in the carcinoma section, 260 images were in the carcinoma section, and 215 were normal. This paper shows that enhancing CNN models can be effective for lung cancer diagnosis.

This paper [11], Hadrien T. Gayap et al., applied deep transfer learning for classifying lung nodule malignancy by evaluating Conventional Neural Networks, including VGG16, MobileNet, InceptionV3, ResNet50, and DenseNet201. They are used to extracting deep features and then classifying them using machine learning classifiers like SVM and Random Forest. ResNet50 with an SVM-RBF classifier combined performs best, achieving an accuracy of 88.41% and an AUC of 93.19% on the LIDC-IDRI dataset. The research showed that transfer learning using nonmedical images can effectively support lung cancer diagnosis tasks.

In this paper [13], Jiang et al. [2023] conducted a comprehensive review that shows the application of deep learning in medical image-based cancer diagnosis. They analyzed the effectiveness of various deep learning architectures, including Convolutional Neural Networks (CNN), Autoencoders (AE), Recurrent Neural Networks (RNN), Vision Transformer (ViT), and Graph Neural Networks (GNN). They use it in different imaging modalities like CT, MRI, X-ray, Ultrasound, PET, and histopathology. The paper described how these models perform in tasks such as image classification, segmentation, detection and synthesis. The study shows that deep learning models provide significant performance improvements over traditional diagnostic methods, especially in terms of accuracy and efficiency.

This paper [16] Sher et al. provides a review of deep learning-based lung cancer detection methods applied to medical images, particularly CT and X-ray scans. The study shows the performance of various convolutional neural network (CNN) architectures, such as AlexNet, VGG16, LeNet, and ResNet, used in the classification and detection of cancerous regions. The paper highlights that many deep learning approaches achieve high accuracy (some above 97%), sensitivity, and specificity in detecting lung cancer. It also points out major challenges, including the presence of noise in images, the scarcity of high-quality annotated datasets, and the issue of false positives.

This paper [18], Bhoj Raj et al., proposed and deep learning approach for lung cancer classification using CT scan images. They used a Convolutional Neural Network (CNN) combined with an autoencoder for feature extraction and classification. Their innovation was the Multispace Image Reconstruction (MIR) method, which replaces the traditional RISA pooling technique. MIR helps reconstruct CT images more clearly by considering features like gradient, texture, and local patterns, leading to better accuracy. The model uses the Adam optimization algorithm to reduce overfitting. For classification, they use a SoftMax layer. The system was tested on public datasets such as TCGA, LCTSC, and QLC, where it achieved a high classification accuracy of 99.5%. The processing speed improved from 10 to 12 frames per second, making the model more efficient.

This paper [20], Hesamoddin Hosseini et al., provides how deep learning techniques have been applied to lung cancer detection. The study analyzes a wide range of deep learning models, including CNN, DCNN, DenseNet, Mask R-CNN, MTMR-Net, STM-Net, ANN, and Inception-V3, in different datasets such as LIDC-IDRI, LUNA-16, NSCLC-Radiogenomics and Kaggle. The review categorizes based on input types, preprocessing, model architecture, hybrid methods, and transfer learning. Some methods perform with high accuracy, like as 98.5%. The Paper brings to light certain benefits of using deep learning for early-stage detection.

This paper [23], Durgesh Srivastava et al., gives a new feature system that can detect early lung cancer based on deep learning. They use CT images. The paper proposes a modified version of Faster R-CNN called Hybridized Faster R-CNN (HFRCNN), which integrates Feature Pyramid Networks (FPN) and adjusted anchor scales to improve nodule detection accuracy. This two-stage object detection model effectively detects small lung nodules. The model was trained on the LID-IDRI dataset. The proposed HFRCNN model achieved over 97% detection accuracy, outperforming existing methods.

Study ID	Author(s)	Year	Framework	Key Findings	Gap
01	**Wankhade & Vigneshwari**	2023	Hybrid Neural Network (3D-CNN + RNN)	Early and accurate detection; benign vs malignant classification.	Limited datasets; lacks clinical validation.
02	**Shariff, Paritala & Ankala**	2023	Review (CNN, VGG16, AlexNet, SVM, CSA,CAD)	DL models improve accuracy; pre-processing enhances results.	Small datasets; poor standardization; interpretability issues.
03	**Shatnawi et al.**	2025	Enhanced CNN vs pre-trained models	Achieved 100% accuracy with Enhanced CNN.	Small dataset; risk of overfitting; no external validation.
04	Venkatesh et al.	2024	CNN + patch processing on CT	High accuracy, noise reduction, faster detection	Limited datasets, lacks multimodal/clinical validation
05	Gayap & Akhloufi	2024	Review of DL (CNNs, ViT, NLP)	DL outperforms ML, strong results on LIDC/LUNA	Issues: bias, generalization, interpretability
06	Jiang et al.	2023	Review of DL in cancer imaging	DL effective in classification, segmentation, fusion	Dataset scarcity, poor generalization, explainability gaps

Table 2.2: Summary of some included studies

2.3 SUMMARY

Overall, the literature shows massive advancement in AI use for the diagnosis of lung cancers. Several of the studies have cited an assortment of deep-learning models (CNN, ResNet, MobileNetV2, LeNet, EfficientNetB0, and VGG16) capable of achieving 90% accuracy. Such models are usually trained on large datasets and hence have the beauty of performance, but pose problems of imbalanced datasets, generalizability and expensive computation. Future studies could consider lightweight architectures, multimodal datasets, and XAI techniques to harmonize the capabilities of AI with clinical practice.

CHAPTER 3

METHODOLOGIES

3.1 RESEARCH WORKFLOW

Step 1: Data Collection

Gather a dataset containing images of the three classifications of lung images. Benign (tumor): Malignant (abnormal growth) and healthy lungs.

Step 2: Data Preprocessing

Background-removal, cropping, augmenting, rescaling, resizing, and label encoding are the steps to be performed.

Step 3: Train-Test Split

This dataset is to be divided into parts: The training, validation and testing datasets.

Step 4: Selection of Pre-Trained Models

For the purposes of feature extraction, pre-trained models like EfficientNetB5, ResNet50, ViT and InceptionV3 will be selected.

Step 5: Feature Extraction

These pre-trained models will be used for feature extraction with their base layers frozen in order to keep them learned.

Step 6: Prediction

Using the trained model to classify the 3 types of lung condition, and measure accuracy.

Step 7: Explainability with Grad-CAM++

Using Grad-CAM++ to fetch visual explanations from the model through heatmaps over images.

Workflow Diagram

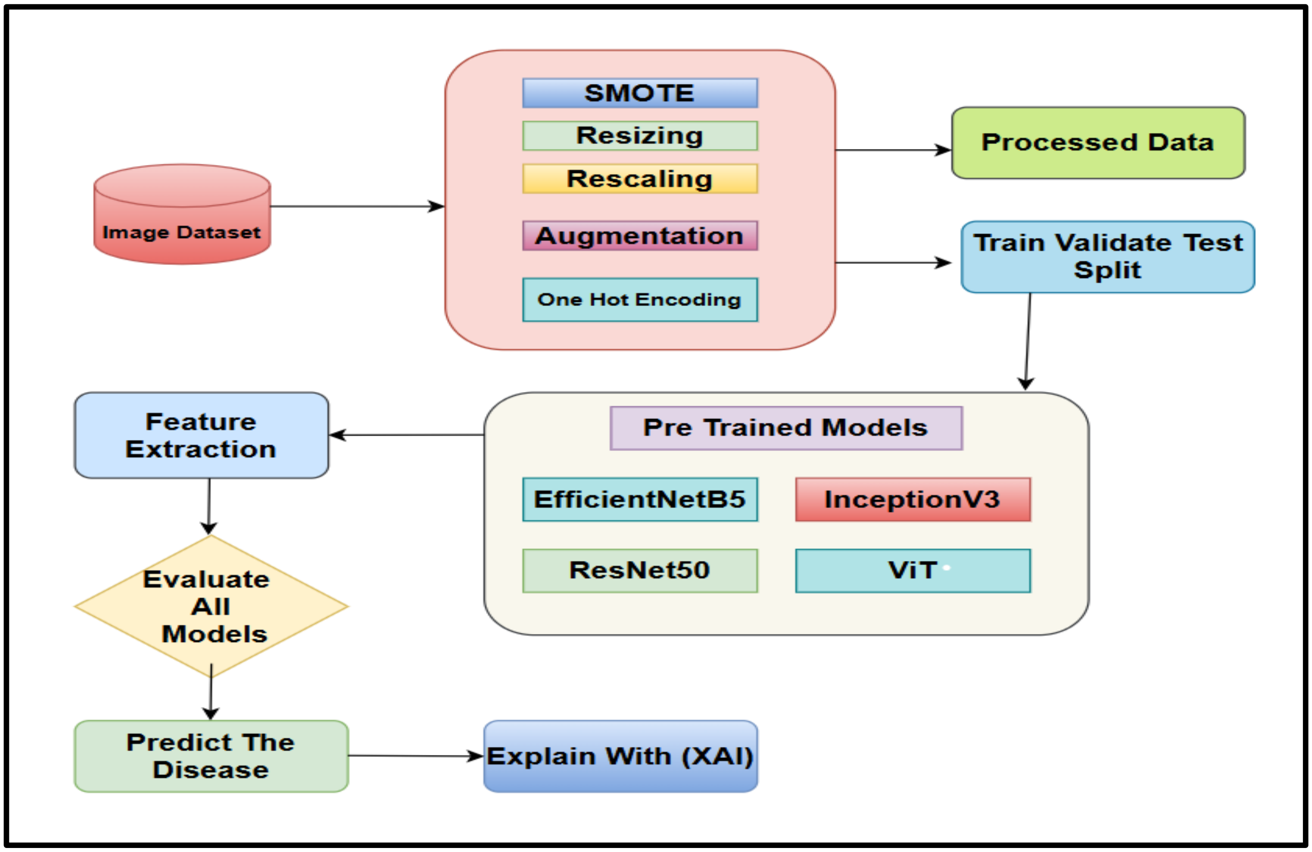


Figure 3.1: Workflow Diagram

3.2 DATASET

3.2.1 DATA COLLECTION

The dataset used in this experiment is a publicly accessible resource collected by Kaggle. This lung cancer dataset was collected over three months from the Iraq-Oncology Teaching Hospital/National Center for Cancer Diseases. The dataset had a total of 1190 images. It represents CT scan slices of 110 cases. A total of 110 labelled images were used in this experiment, in which 15 images are benign, 40 images are malignant, and 55 images are normal. The 110 cases vary in different factors like age, gender, living status, education and living area. They are grouped into three classes: normal, benign, and malignant. The CT scan image is in DICOM format. Three sample images (tumor, cancerous and noncancerous) from the dataset are displayed in Figure 3.2.1.

Images for lung condition

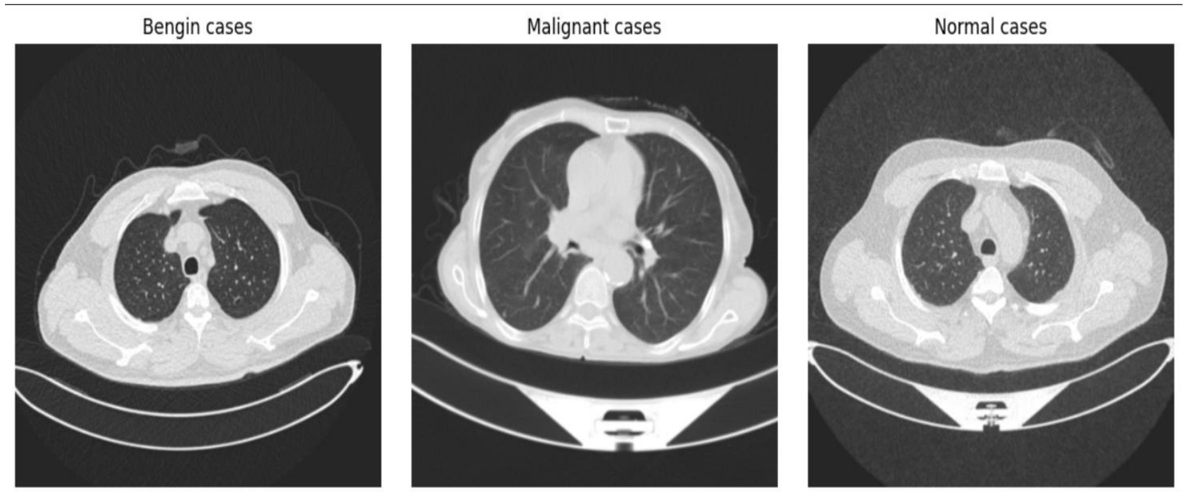


Figure 3.2.1: Sample Images for lung condition

Feature	Value
Dataset Type	Medical Imaging
Dataset Size	1190 CT Scan Images (from 110 Cases)
Source	Iraq-Oncology Teaching Hospital and National Center for Cancer Diseases
Annotation Type	Oncologists' and Radiologists' Markings
Nodule Types	Normal, Benign, and Malignant
Nodule Annotations	Yes
Number of Cases	Normal: 55, Benign: 15, Malignant: 40
Image Format	DICOM
Scanner Used	SOMATOM (Siemens)
CT Protocol	120 kV, 1 mm Slice Thickness, Window Width: 350–1200 HU, Center: 50–600
Breath Status	Full Inspiration Hold
Slice Count per Scan	80 to 200 slices per case
Patient Demographics	Vary in age, gender, occupation, region
Geographical Coverage	Baghdad, Wasit, Diyala, Salahuddin, Babylon
Year of Collection	Fall 2019

Table 3.2.1: Influential and Characteristic features of Dataset

3.2.2 DATA PRE-PROCESSING

The three lung conditions images classified as normal, benign, and malignant were imported from folders and automatically tagged according to the name of each folder. For input segmentation to adapt pre-trained models, all such images will be resized downwards to 456 x 456.

Class Distribution of cases

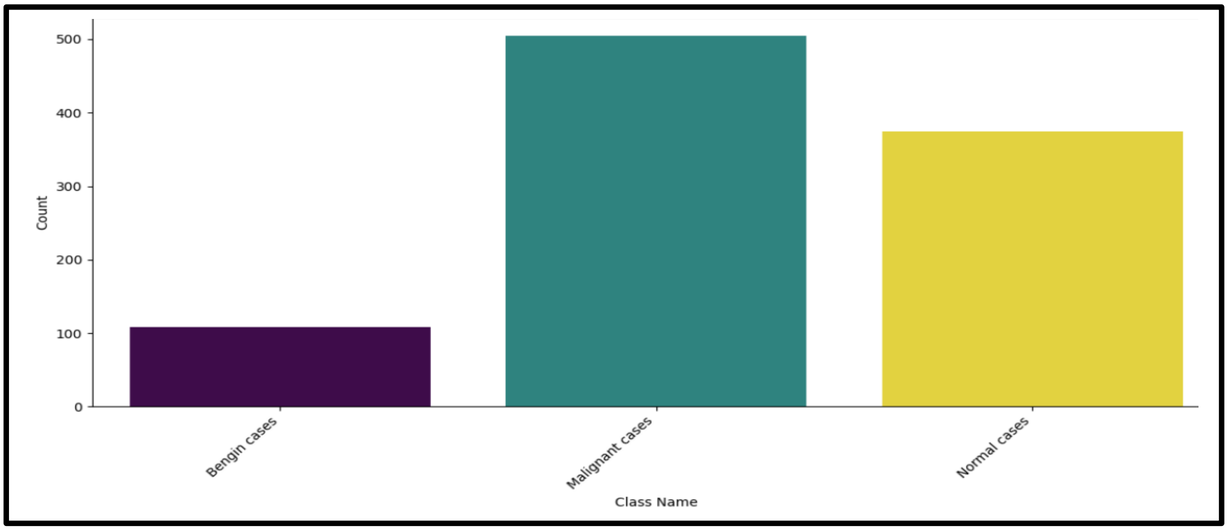


Figure 3.2.2: Initial Class Distribution of cases

Class Distribution of cases After SMOTE

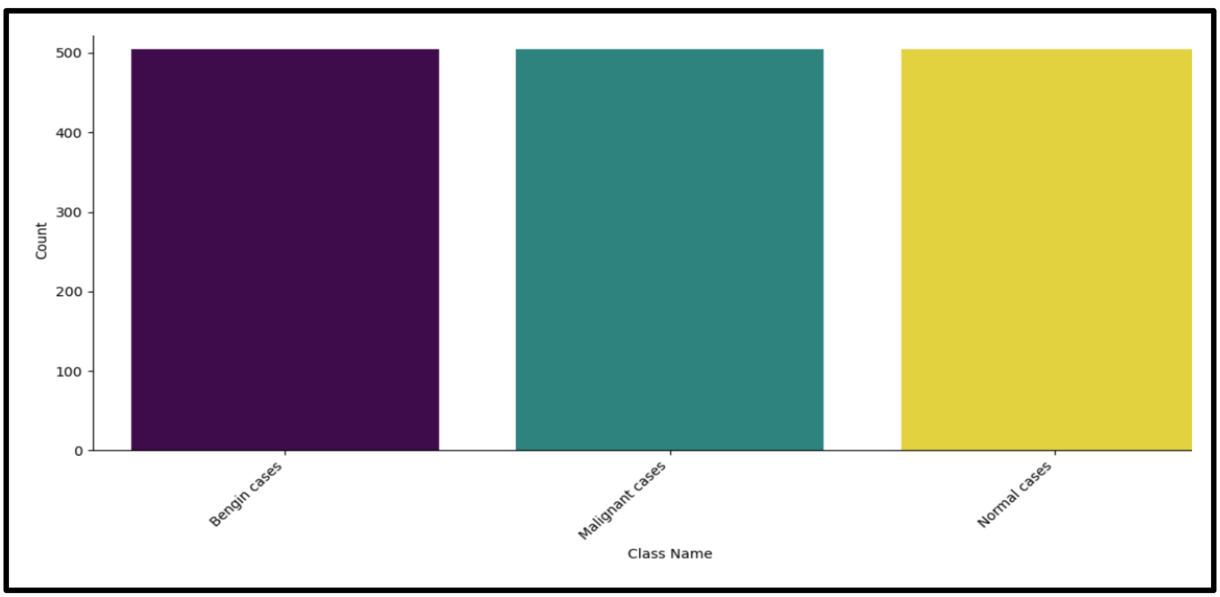


Figure 3.2.3: Class Distribution of cases After SMOTE

Data augmentation

Data augmentation is changing the original image into another form for independent new data points, which greatly increases the size and diversity of the dataset. It acts to minimize overfitting and enhances the generalization potency of the model significantly.

There are augmentations as follows:

Random Flipping: The model uses horizontal flipping for trained Images. This helps the model know that left right orientation cannot affect cancerous features.

Random Rotation: Randomly rotates the image up to 10% to emulate different orientations.

Random Zoom: Alteration is done on the basis of zooming into and out of a certain area in images for feature detection.

Without further application of any data, an even broader dataset learning could be accomplished through the creation of different versions of the same image.

Resize

Images collected were different in their original dimensions. This research resized images according to the input model requirements of the pre-trained models employed in this research. Images were resized according to EfficientNetB5 (456×456), ViT and ResNet50 (224x224) and InceptionV3 (299x299). It standardizes the input size and ensures computational efficiency. After all, processing high-resolution images can also take much time and be complex.

Encoding

These are the three classes of lung diseases in the dataset and how the image labels have been encoded into a one-hot encoding scheme. Conversion occurs while going from 'categorical labels' to a 'numerical format' by generating a 2D NumPy array of one-hot encoded labels used with the loss function.

Rescale

Pixel values were normalized by rescaling to a range of [0, 1] for all images. It had a potential to create instability and slower convergence while training since original pixel values ranged from 0 to 255. Normalized data comes when pixel values are divided by 255 for ranging from 0 to 1. It ensures better numerical stability, training performance, and faster convergence. Both the validation and test datasets were subject to rescaling.

Normalization

Normalization means scaling data into a range, usually between 0 and 1. Pixel values usually lie between 0 and 255. Dividing by 255, the day values are rescaled to range [0, 1]. This step will ensure that weight updates during training are not dominated by a few larger pixel values.

One-Hot Encoding

One-hot encoding in coding transforms the tags of a class with lung cancer into a numeric form that a deep learning model can understand. For each disease category, the binary vector represents it in a classification task. Since three classes are available, normal would be [1, 0, 0] while Benign would be [0, 1, 0]. The level is treated as a distinct category and hence has entirely avoided the unintended relationship between classes.

3.3 DATASET SPLIT

This dataset has the parts of training, validation, and testing for evaluating the model perfectly. There is a folder where images of three different types of lung conditions are stored, used as input, and the model infers the labels based on the name of the folder in which it is stored.
This dataset has been split into the following parts:

Training Set: 90% of the total data is used to train the model. This gives insight into the model under learning and its behavior in terms of lung cancer-related features.
Validation Set: 10% of this data set, which is validated within the training of the model to prevent overfitting of that model.

Testing Set: An independent set of tests is applied to assess the accuracy given by the model. It thus acquires the full generalization ability towards completely new lung images by considering all characteristics of lung images.

Dataset Preprocessing workflow

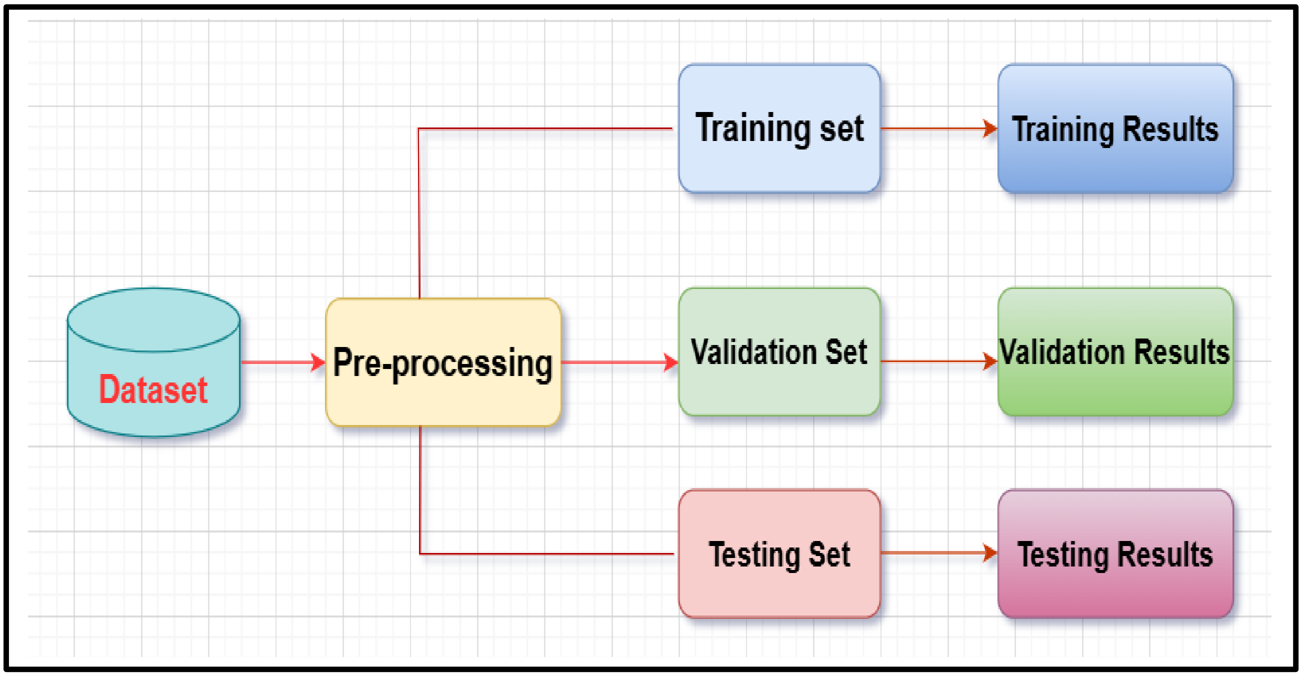


Figure 3.3: Dataset Preprocessing workflow

3.4 PROPOSED MODEL

This proposed model is designed to classify three lung disease conditions--Benign, Malignant and Normal —using a combination of pre-trained deep learning architectures: EfficientNetB5, ViT, ResNet50 and InceptionV3. These architectures have been chosen for their proven ability to extract robust features from medical images.

Resizing according to this initial order goes on with preprocessing the lung image and resizing it to 456×456 for EffectiveNetB5 and 299×299 for InceptionV3, 224×224 for ViT and ResNet50 along with maintaining the training of pre-trained models. These models are feature extractors where the pre-trained convolutional layers help in identifying essential patterns like textures, shapes, and structures of interest in the lung tissues.
Once features are extracted, the output of the models is passed to a Global Average Pooling 2D layer, which reduces the spatial dimension of the features while retaining global features.

This is followed by a Batch Normalization layer, which speeds up the training process and extracts features.

Further processing of the feature set is carried out through a Dense layer of 256 neurons with ReLU activation. A Dropout layer at 0.3 is included to counter overfitting by randomly turning off some neurons during training.

A SoftMax classification layer is used to provide the probabilities of the three lung diseases. It simply provides the probability distribution, wherein the maximum value signifies the predicted class.

Work process block diagram

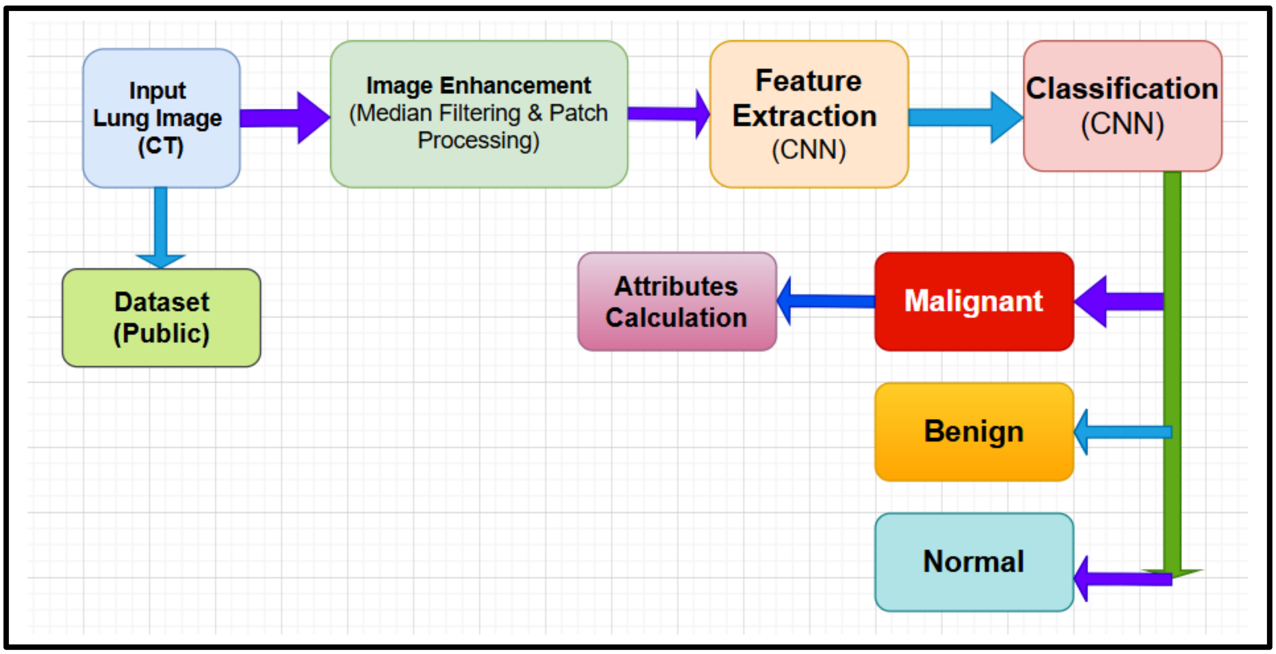


Figure 3.4: Work process block diagram

3.5 PRE-TRAINED MODEL

Fine-tuning training methods on new but related problems learned from other tasks (for example, EfficientNetB5 or InceptionV3 or ResNet50) typically refers to teaching to solve a specific problem-level activity. Convolutional layers extract features such as texture, shape, etc. The fully connected layer is replaced for retraining on the new task. This helps to minimize training time, get better accuracy, which is very important when there's a shortage of data.

3.5.1 **EFFICIENTNETB5**

Among the architectures of transfer learning, EfficientNetB5 comes as one of the most amazing architectures whose application gives very high accuracy with extremely few parameters and requires less computation cost. Compound scaling introduced by EfficientNet allows to reduce depth, width, and input resolution so that performance can be optimally balanced with efficiency. It applies MBConv layers, which are combinations of depth-wise and point-wise convolutional layers to create efficiency in computation, but in the meantime maintain richness in extracted features.

Miracles are still happening in other distinct areas of EfficientNetB5. Unlike the other forms of medical pathology, COVID-19 in chest X-ray classification and the detection of diseases in the retina are said to achieve higher accuracy with fewer parameters. The features have increased the operating precision and recall; training needed to be optimally tuned for hyper parameters due to its efficacious nature, while delay in inferences is compared to its smaller variants.

In this study, EfficientNetB5 was trained to discern three classes of lung conditions: Benign, Malignant and Normal. After first being trained via transfer learning with ImageNet weights, EfficientNetB5 had taken on features more specific to lungs-they included adding three more layers on top of the Global Average Pooling layer, including a Dense layer of 256 units with a Dropout layer to help condense features most effectively (with a probability of 0.3). All of the target classes were predicted via a Softmax output layer. Random flipping, rotation and zooming were included for data augmentation to help reduce overfitting. This study outlines the adaptation and performance of EfficientNetB5 through medical image analysis in a fit-for-purpose and justifiable lung disease classification mode.

EfficentNetB5 Network Architecture

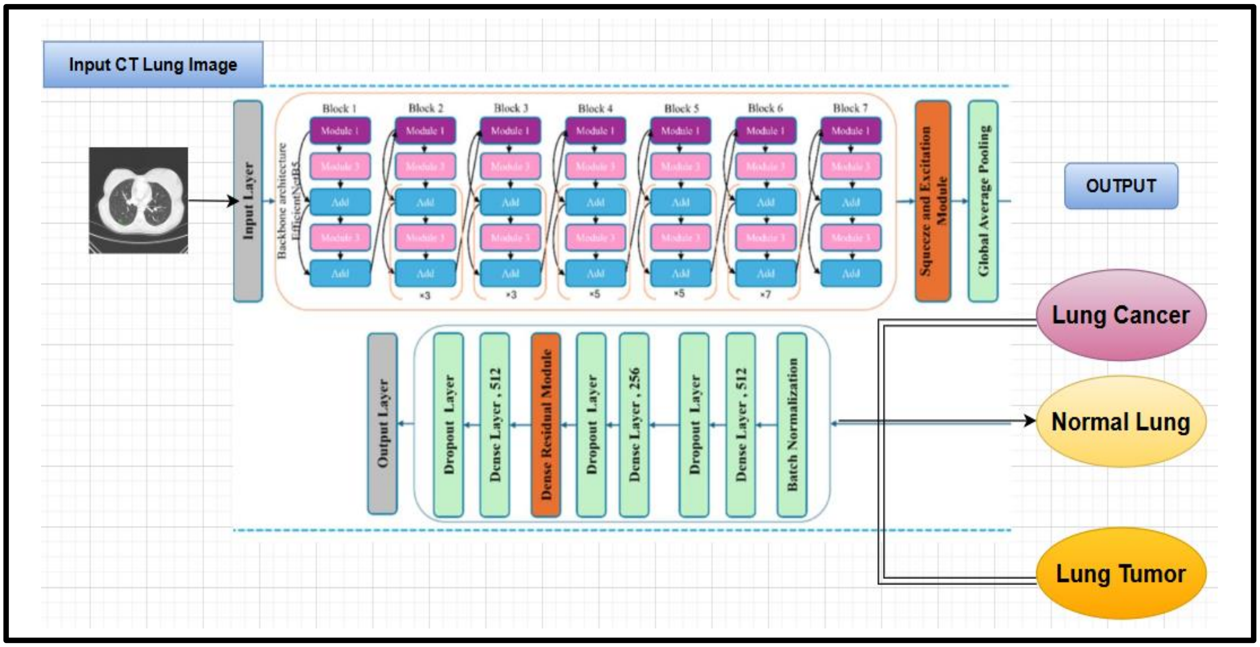


Figure 3.5.1: EfficentNetB5 Network Architecture

3.5.2 Inceptionv3

InceptionV3 is mainly a deep learning model to work towards image classification efficiency and performance. At the same time, its inception modules fetch multiple convolutional filters such as (1×1), (3×3), or (5×5) in order to gather multiple levels of detail in an image. It helps take up coarse-grained details from an image in a highly efficient extraction of fine details. It also provides an optional classifier and implements batch normalization in computation costs to keep it at a minimum with maximum accuracy.

Such outstanding results have been attained by InceptionV3 for applications in medical image processing, like lung pathology, while several other aspects made it so. The most complex visual patterns can be recognized by distinguishing different receptive fields with a CSP-CNN architecture designed on a multi-path approach. After ImageNet pre training weights and transfer learning, the model regrouped lung-specific image features. Then, Global Average Pooling pooled the feature maps, while the Softmax output layer made the final decision.
Data augmentation settings are employed to avoid overfitting by having random flipping, rotation and zooming. It specializes in learning patterns from medical images.

InceptionV3 Network Architecture

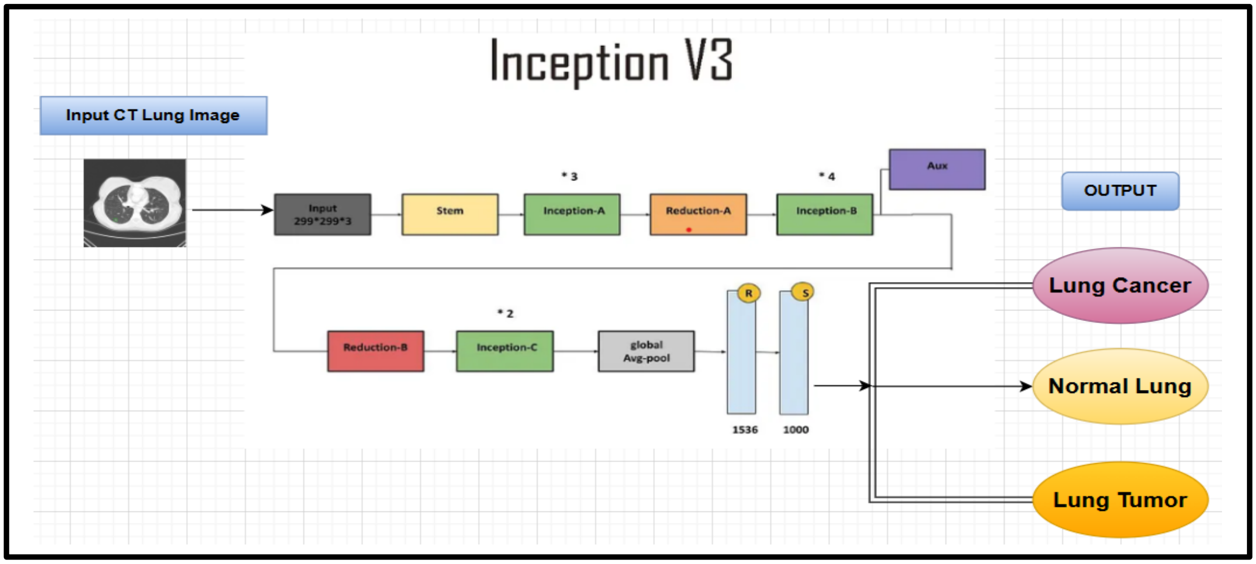


Figure 3.5.2: InceptionV3 Network Architecture

3.5.3 ResNet50

ResNet50 is a deep CNN architecture model, and it tries to vanish gradients in deep networks. Residual learning is the main basis of this model. Here shortcut connections allow gradients to flow directly between layers without any degradation. The model has 50 layers. Some of them are fully connected residual blocks, convolutional and pooling. This model can capture low-level and high-level images. ResNet50 can spot spatial features from chest X-rays and CT scans; thus, it can identify lung cancer. Residual connections are suited for achieving higher accuracy and spotting complex structures like tumors, nodules, and tissue irregularities. ImageNet-pre trained The weights are pre trained on ImageNet, and transfer learning works well with medical data sets with few training data samples. Layers for dimension are applied, followed by Global Average Pooling layers and Softmax layers for classification. Applying random rotation, shifting, and scaling the input data improves generalization and avoids This ResNet50 can learn different levels of hierarchical representations on medical images and can prove its effectiveness in lung cancer detection.

ResNet50 Network Architecture

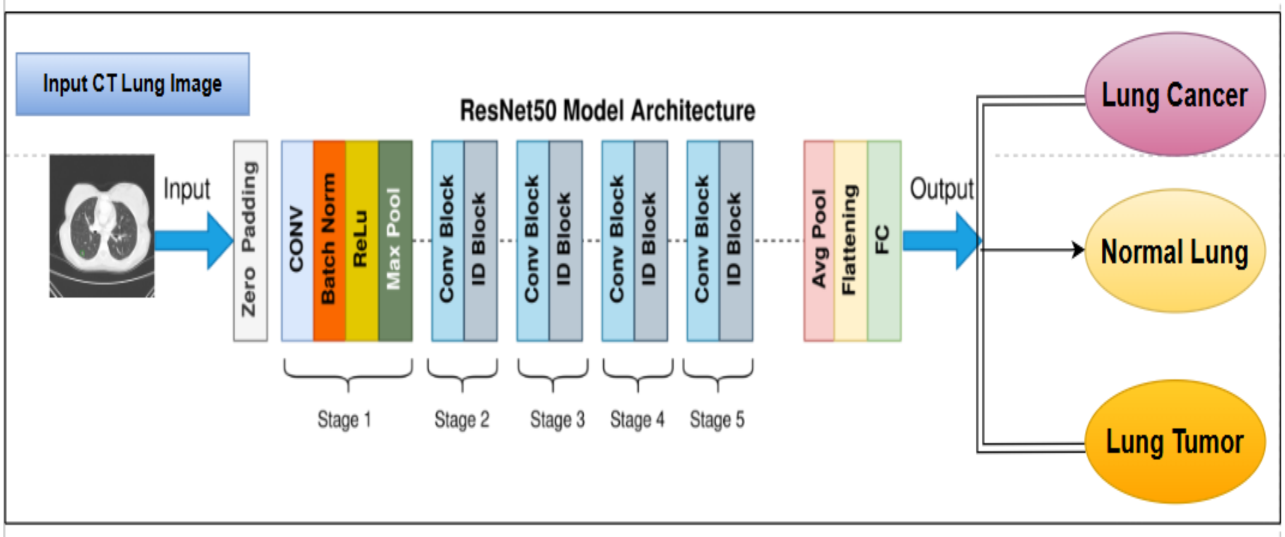


Figure 3.5.3: ResNet50 Network Architecture

3.5.4 Vision Transformer (ViT)

The vision transformer is a DL model. This transformer architecture works as the main structure for tasks in NLP. This model works for classification and image-based tasks. ViT is generally used to cut an image into predefined sizes, linearly set them, and process them in a sequence.

For lung cancer detection, ViT shows its ability to learn hidden spatial local regions and irregular patterns in medical subjects. Sometimes it can be ignored in the traditional convolution process. The model was pre-trained on a large dataset like ImageNet that was adapted into the lung-specific datasets. Then the output is input into a classifier with a Softmax for making a final decision.

ViT is ductile with all kinds of data augmentation, like patch shuffling, critical survey, and random cropping. Transfer learning also leads to better efficiency for small medical datasets. Vision Transformer has the power for capturing global context and complex spatial dependencies so that it has good capability in identifying early-stage lung cancer patterns.

Vision Transformer Network Architecture

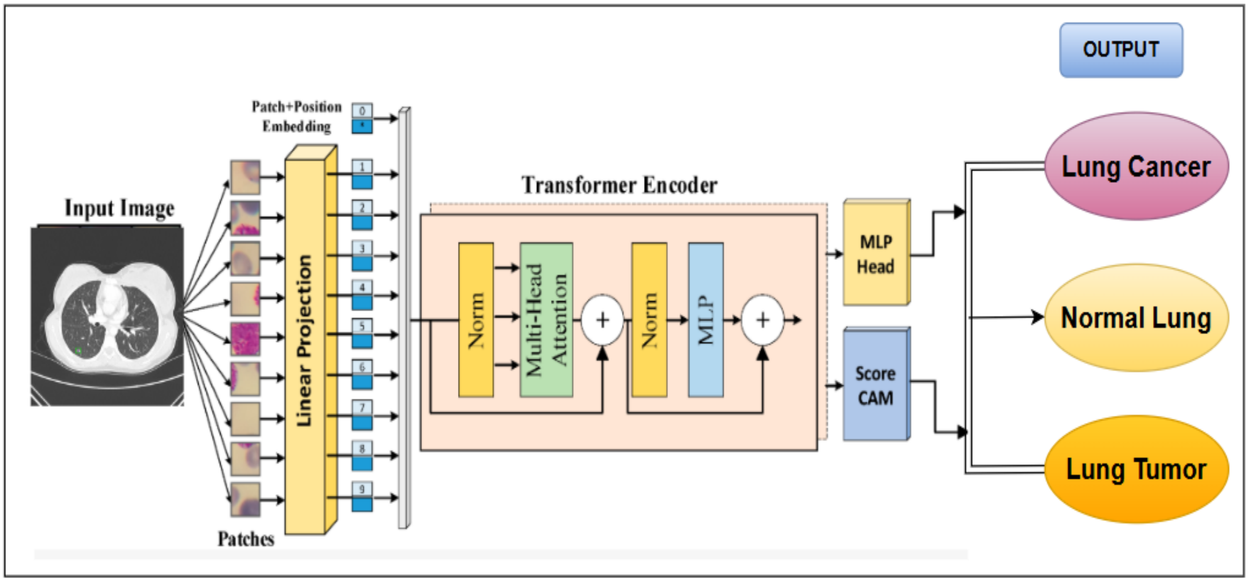


Figure 3.5.4: Vision Transformer Network Architecture

3.6 FEATURES EXTRACTION

Feature extraction is an important step in deep learning models, as it helps to capture meaningful patterns from input data, such as images. EfficientNetB5, ResNet50 ,InceptionV3 and ViT architectures were used to extract hierarchical features from lung images, enabling accurate classification of diseases like lung tumors or cancer.

InceptionV3, ResNet50, ViT and EfficientNetB5 were employed to extract deep hierarchical features from images of three lung conditions, such as Benign, Malignant and Normal. The models captured essential patterns, textures, and shapes from the images while discarding irrelevant information by leveraging their pre-trained convolutional layers. (XAI) Methods were unified to analyze and highlight the extracted features for predictions. It makes the diagnostic process transparent and more trustworthy for clinicians and patients as well.

3.7 PREDICTION

It involves refining the process of pre-trained models in translations to give meaningful predictions concerning the three categories of lung diseases, including Benign, Malignant and normal, with combinations of dense layers. A SoftMax activation layer that ensures accuracy is interpretable in this classification.

Fully connected dense layers after feature extraction with EfficientNetB5, InceptionV3, ResNet50 and ViT take the resultant feature maps as the classifiers and learn the relationships among the features extracted from the input images. Implements a SoftMax activation function, returning probabilities for each class. Such outputs might be, for example: Benign: 0.02%, Malignant: 0.10%, Normal: 0.88%.

This interpretation is furthered by the generation of visual explanations of (XAI) predictions through Explainable AI techniques-GRAD-CAM that visualize important or affected areas via heatmaps.
This process bridges the gap between technical performance and real-world usability in the medical area.

3.8 EXPLAINABILITY WITH XAI

Interpretability is important to bridge the gap between AI-powered diagnostic models and clinical usability. In this research, interpretability was achieved by integrating XAI techniques such as Grad-CAM for visualizing a heatmap over important regions and textual explanations of the model's predictions. Grad-CAM visualizations of input images, giving the most important regions that contribute to the model's prediction, like lesions and textures indicative of specific conditions such as Benign, Malignant and Normal. It gave actionable insights to the medical professionals by allowing them to validate the decision-making process of AI (XAI).

Before and After Applying XAI

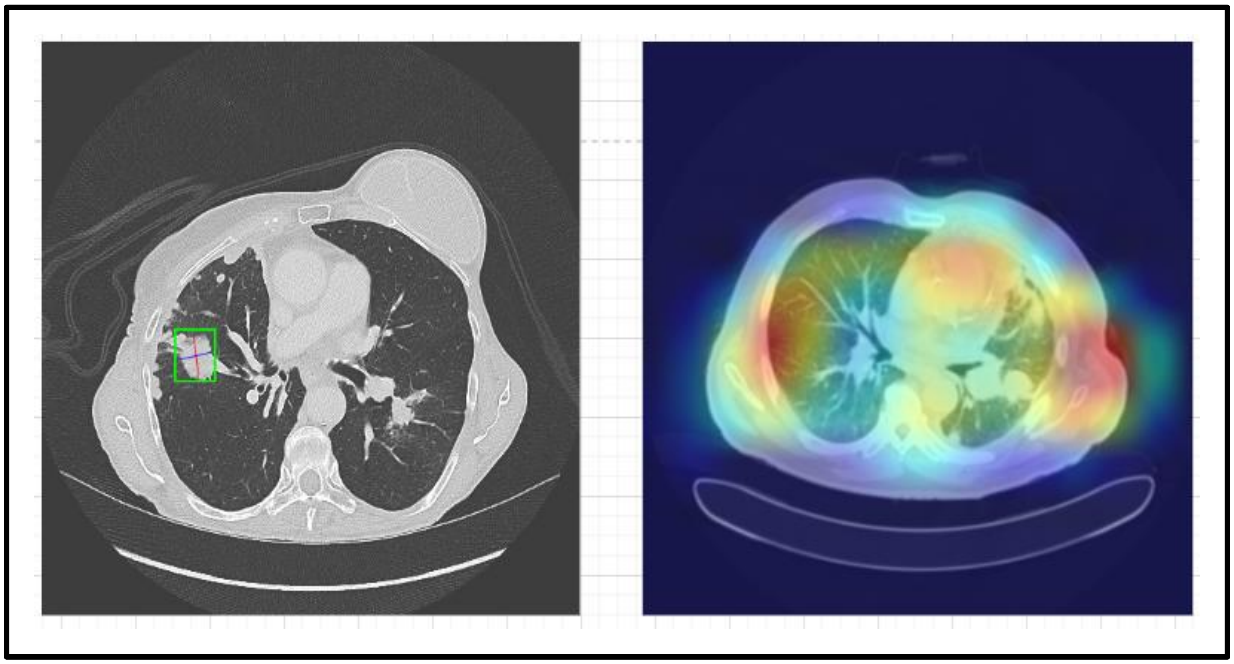


Figure 3.8: Before and After Applying XAI

CHAPTER 4

RESULT AND DISCUSSION

4.1 RESULT DETAILS

4.1.1 PREDICTED RESULTS

From the obtained predictive results, the suggested model can be defined as quite capable in a considerable way in the accurate classification of lung ailment images. The resultant models display rows containing images along with their respective true and predicted labels. Consequently, the model can be considered highly accurate in the identification of different lung diseases.

Image of predicted result

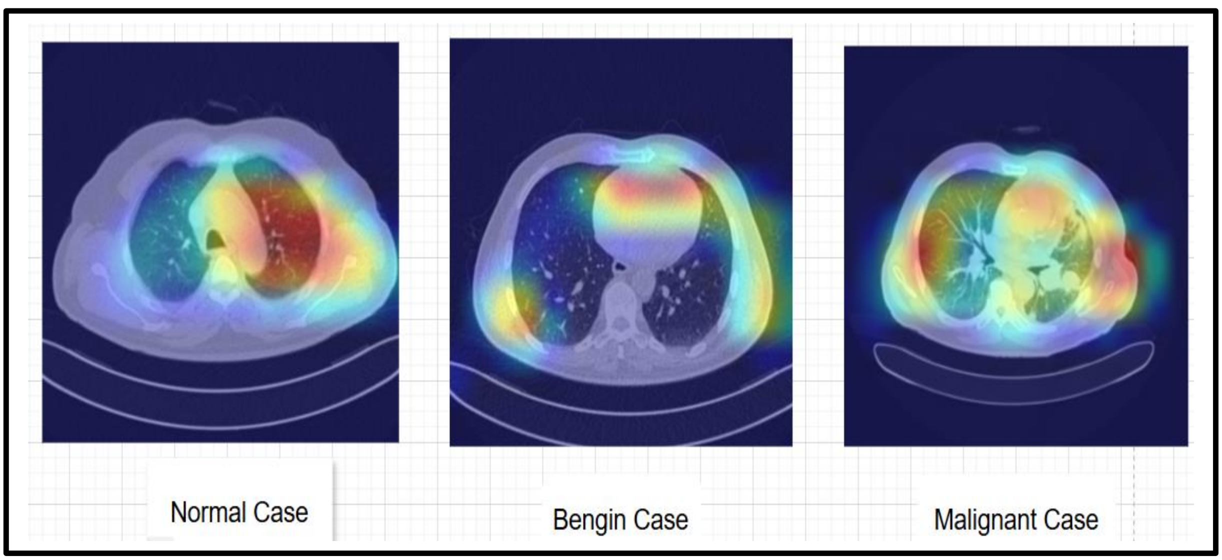


Figure 4.1.1: Image of predicted result

Results include:

True Positives: For the most part of lung cases such as Benign, Malignant, the predicted label commiserates with the real label. The feature-oriented model identifies the diseased class correctly.

Visual Examples: The affected portion in the image becomes evident thus the interpretation as to what the classification model relied upon becomes simple.

Diversity of Data: The results make clear evidence that the model generalizes across different lung diseases with different lesion sizes and varying contexts in the images.
Overall these include the successful performances exhibited by the model training and testing.

4.1.2 PERFORMANCE ANALYSIS OF THE PRE-TRAINED MODELS

Experimental Results

Different studies made with lung disease classification task use different models to check their effectiveness. Each model has run for 7 times with 50 epochs, and the average of these results is calculated to measure the performance of each model. The process can reduce bias and provide better results for models. EfficiencyNetB5 holds a highest classification accuracy of 99.13 percent for lung diseases claiming to have great asset in detail extraction comparing with all models.
Multi-scale feature capture through parallel convolutional operations was done by Inception V3 at an accuracy of 98.21 percent in lung disease detection.
A balanced feature extraction operation was done by ResNet50 at an accuracy of 97.66 percentage in lung disease detection. ViT obtained global dependency in medical images at an accuracy of 96.57% for this dataset.

Validation accuracy of EfficientNetB5 model

Class	Test Data Count	TP	TN	FP	FN	TPR (Recall)	TNR (Specificity)	FPR	FNR	Accuracy	Precision	Recall	F1-Score
Bengin cases	91.0	91	182	0	0	1.0000	1.0000	0.0000	0.0000	1.0000	1.0000	1.0000	1.0000
Malignant cases	91.0	91	182	0	0	1.0000	1.0000	0.0000	0.0000	1.0000	1.0000	1.0000	1.0000
Normal cases	91.0	91	182	0	0	1.0000	1.0000	0.0000	0.0000	1.0000	1.0000	1.0000	1.0000
Average (Micro)	273.0	273	546	0	0	1.0000	1.0000	0.0000	0.0000	1.0000	1.0000	1.0000	1.0000

Table 4.1.2: Validation accuracy of EfficientNetB5 model

The table of accuracy analysis supports the performance of models in the classification of lungs into three states. The highest accuracy attained so far is 99.13%, scored by EfficientNetB5 on top of precision, recall, and F-scores of 99%, making it the best model for this dataset.

Validation accuracy of InceptionV3 model

Class	Test Data Count	TP	TN	FP	FN	TPR (Recall)	TNR (Specificity)	FPR	FNR	Accuracy	Precision	Recall	F1-Score
Bengin cases	91.0	91	182	0	0	1.0000	1.0000	0.0000	0.0000	1.0000	1.0000	1.0000	1.0000
Malignant cases	91.0	91	182	0	0	1.0000	1.0000	0.0000	0.0000	1.0000	1.0000	1.0000	1.0000
Normal cases	91.0	91	182	0	0	1.0000	1.0000	0.0000	0.0000	1.0000	1.0000	1.0000	1.0000
Average (Micro)	273.0	273	546	0	0	1.0000	1.0000	0.0000	0.0000	1.0000	1.0000	1.0000	1.0000

Table 4.1.3: Validation accuracy of InceptionV3 model

InceptionV3 scored an accuracy of 98.21% with precision, recall, and F-scores of 98%, marginally lesser than that of EfficientNetB5.

Validation accuracy of ResNet50 Network Model

Class	Test Data Count	TP	TN	FP	FN	TPR (Recall)	TNR (Specificity)	FPR	FNR	Accuracy	Precision	Recall	F1-Score
Bengin cases	57.0	55	113	1	2	0.9649	0.9912	0.0088	0.0351	0.9825	0.9821	0.9649	0.9735
Malignant cases	57.0	57	113	1	0	1.0000	0.9912	0.0088	0.0000	0.9942	0.9828	1.0000	0.9913
Normal cases	57.0	55	112	2	2	0.9649	0.9825	0.0175	0.0351	0.9766	0.9649	0.9649	0.9649
Average (Micro)	171.0	167	338	4	4	0.9766	0.9883	0.0117	0.0234	0.9766	0.9766	0.9766	0.9766

Table 4.1.4: Validation accuracy of ResNet50 Network Model

ResNet50 scored an accuracy of 97.66% with precision, recall and F scores of 97%, less than of EfficientNetB5 and InceptionV3.

Validation accuracy of Vision Transformer

Class	Test Data Count	TP	TN	FP	FN	TPR (Recall)	TNR (Specificity)	FPR	FNR	Accuracy	Precision	Recall	F1-Score
Bengin cases	57.0	53	113	1	4	0.9298	0.9912	0.0088	0.0702	0.9708	0.9815	0.9298	0.9550
Malignant cases	57.0	55	110	4	2	0.9649	0.9649	0.0351	0.0351	0.9649	0.9322	0.9649	0.9483
Normal cases	57.0	52	108	6	5	0.9123	0.9474	0.0526	0.0877	0.9357	0.8966	0.9123	0.9043
Average (Micro)	171.0	160	331	11	11	0.9357	0.9678	0.0322	0.0643	0.9357	0.9367	0.9357	0.9359

Table 4.1.5: Validation accuracy of Vision Transformer

Vision Transformer scored a lower validation accuracy of 96.57% than other models, with precision, recall and F scores of 96%. Its accuracy was less than CNN- based models.

My study, therefore, stands parallel to existing results from literature as it improves upon the findings using EfficientNetB5.

Explanation of Accuracy Curves for Different Models

Accuracy curves show how well the pre-trained models did in training and validation for different tasks.

For EfficientNetB5, the accuracy curve went up fast in the first 10-15 tries, with training and validation accuracies going their separate ways a bit. The model isn't overfitting, so it should be able to handle new data well.

InceptionV3 also had steady growth in both accuracies, leveling off after 15 tries. This shows that the model is good at learning features and handling new situations. The accuracy curves in this section (Figure 4.1.4) show how the models performed in training and validation over several tries.

Accuracy Curve for Models

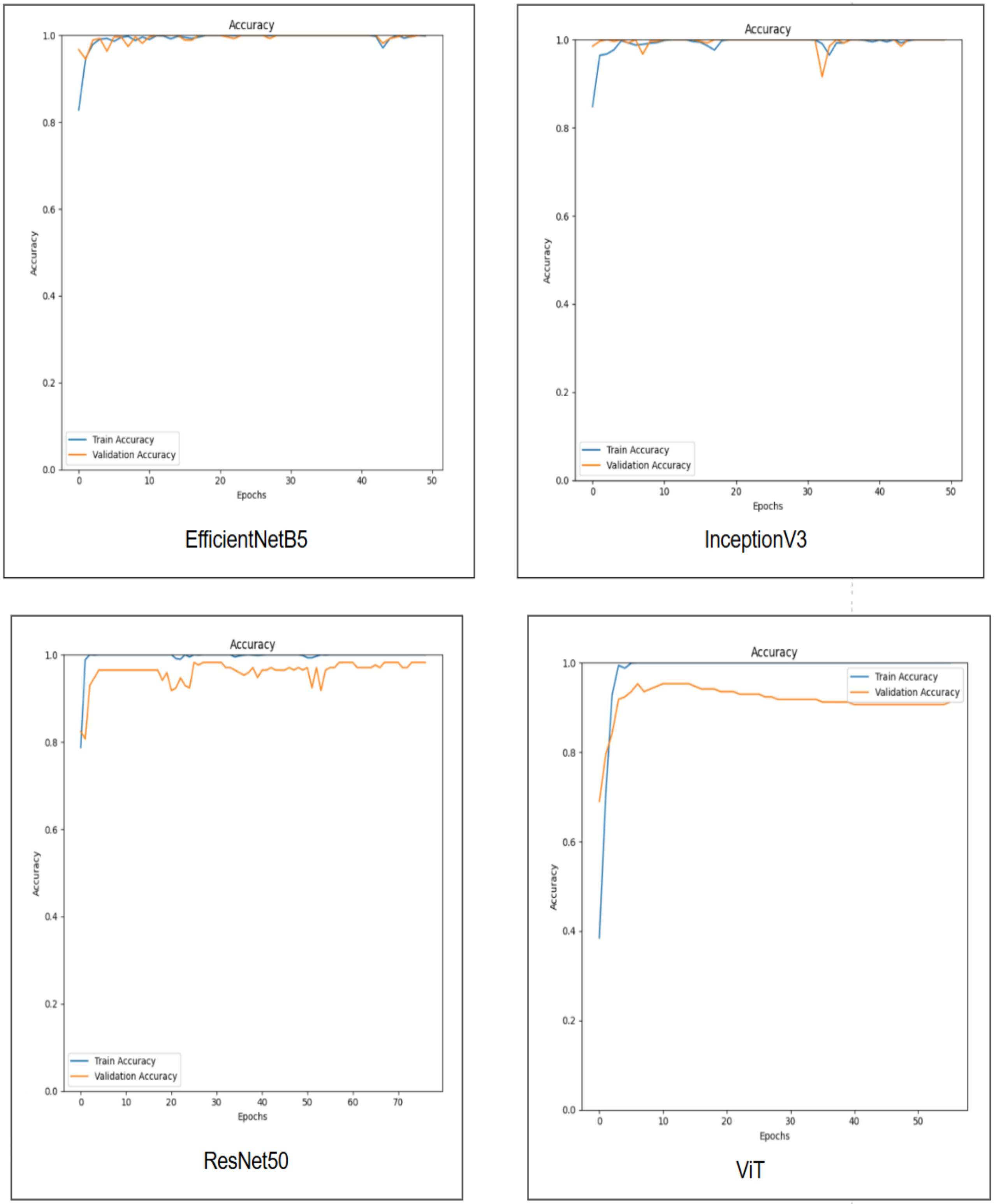


Figure 4.1.2: Accuracy Curve for Models

ResNet50 had growth in both accuracies, labelling off after 15 tries. This shows that the model can maintain generalizations. The accuracy curves in this section (figure 4.1.4) show how the model performed in training and validation over several tries.

Vision Transformer had slower growth in training and validation accuracy, levelling off after 15 tries. This shows that ViT can adopt global dependencies. But it's less stable than CNN models.

This graph looks at the training and validation accuracies of EfficientNetB5, InceptionV3, ResNet50 and Vision Transformer. EfficientNetB5 and InceptionV3 both did a good job, with about 98% accuracy in training and validation, which means they can handle new stuff consistently. EfficientNetB5 had the best training accuracy at 99.13%, but its validation accuracy went down to 98%, and there is no chance it will over fit. InceptionV3's training accuracy was 98.21%, and its validation accuracy was 97%, which shows it can learn well. ResNet50 had training accuracy close to 98%, but validation accuracy went down to 97.66%, so there is minimal risk of overfitting. Vision Transformer training accuracy was above 96%, and its validation accuracy was 95.57% less than all model.

Explanation of Loss Curves for Different Models

The Loss curve mainly decreases error in the training and validation stages of a model. It described if overfitting happened in the model. EfficientNetB5 has an easy go down in training and testing losses within the first 10-15 epochs. Early balance of validation loss showed good generalization. Training and validation losses follow each other very closely. It enables good performance of the model on unseen data. EfficientNetB5 put in values to loss to be balanced and low enough for making easier generalizations. InceptionV3 has a seamless turndown in both training and validation loss. Validation loss continued before training loss so it could achieve good learning ability. In InceptionV3, loss occurred in early cut epoch. Training loss was monitored closely by validation loss; that means the model had good generalization with less overfitting.

Loss Curve for Models

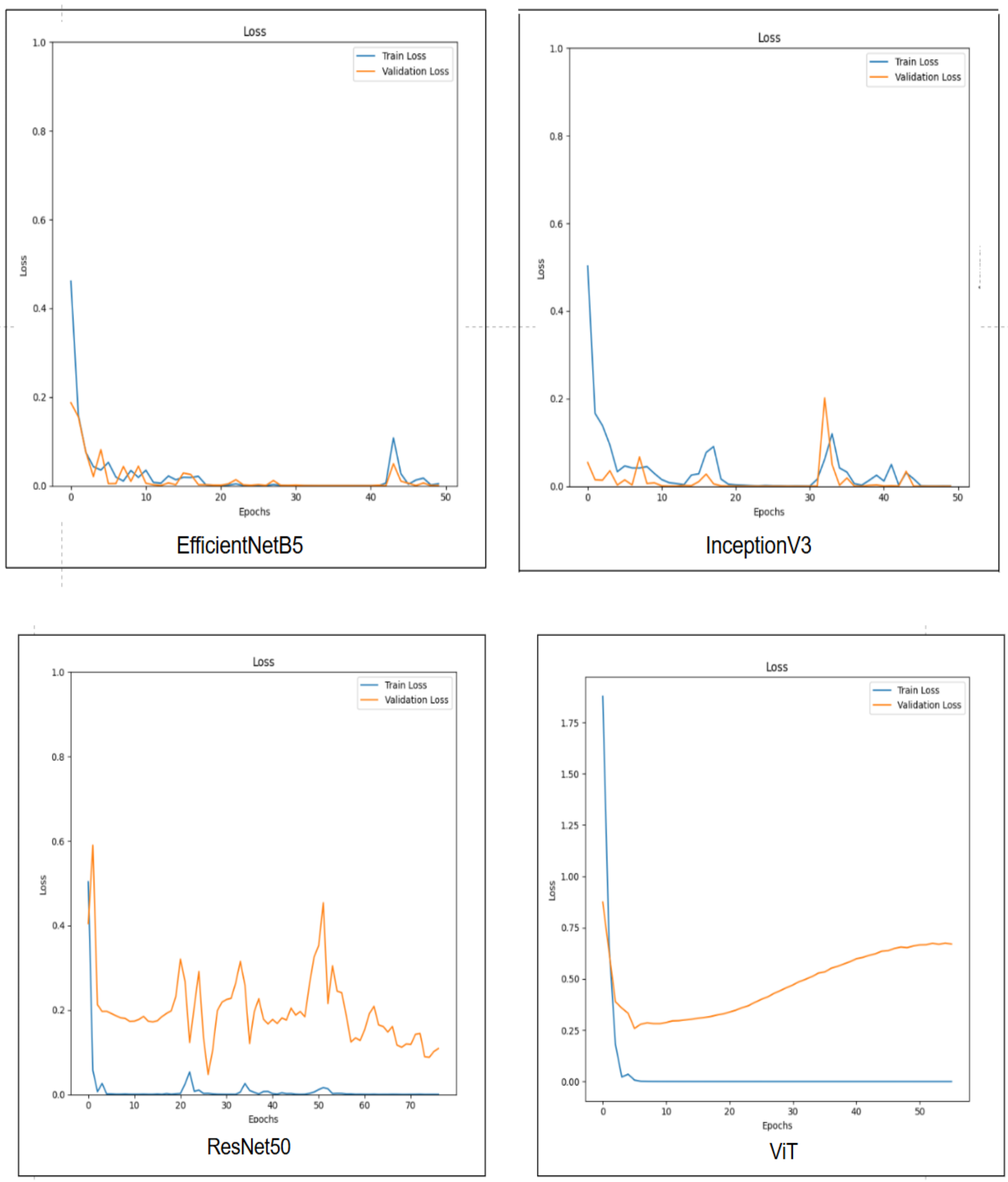


Figure 4.1.3: Loss Curve for Models

For ResNet50, the loss curve showed an even drop in training and validation. The curve dropped over the first 10–12 epochs. Training loss was tracked closely by validation loss means the model had good generalization without much overfitting. Vision Transformer (ViT) showed a more linear dencend in loss. When the training loss decreased, the validation loss oscillated around epoch 18–20. The difference between training and validation losses shows small fragility. So, its generalization was not as strong as other models. All of the loss curves are closely connected to the model evolution.

Explanation of Confusion Matrix for different models

The confusion matrix diagnostic performance for different classifiers. From the model we can see EfficientNetB5 has done well considering classifier performance. It seems to be capable of identifying model predictions having a very low accuracy. The confusion matrix gives direction to identify true values for reducing misclassification. It has perfect 100% classification across all classes.

InceptionV3 has shown very few cases of mistakes in prediction. But it was almost diagonally liable, and one can easily predict that it has better generalization.

ResNet50 shows that 96.5% of the benign class was correctly identified and 3.5% was misclassified. 100% of the malignant class was correctly identified, and 0% was misclassified. 96.5% of the normal class was correctly identified, and 3.5% was misclassified. It has nearly perfect classification across all classes.

Vision transformer (ViT) shows that 93.0% of the benign class was correctly identified and 0.7% was misclassified. 96.5% of the malignant class was correctly identified, and 3.5% was misclassified. 91.2% of the normal class was correctly identified, and 8.8% was misclassified. It has higher confusion between benign and normal classes.

There have been several improvements relative to the study based on the matrix. It mainly compares actual vs. predicted values.

Confusion Matrix for Models

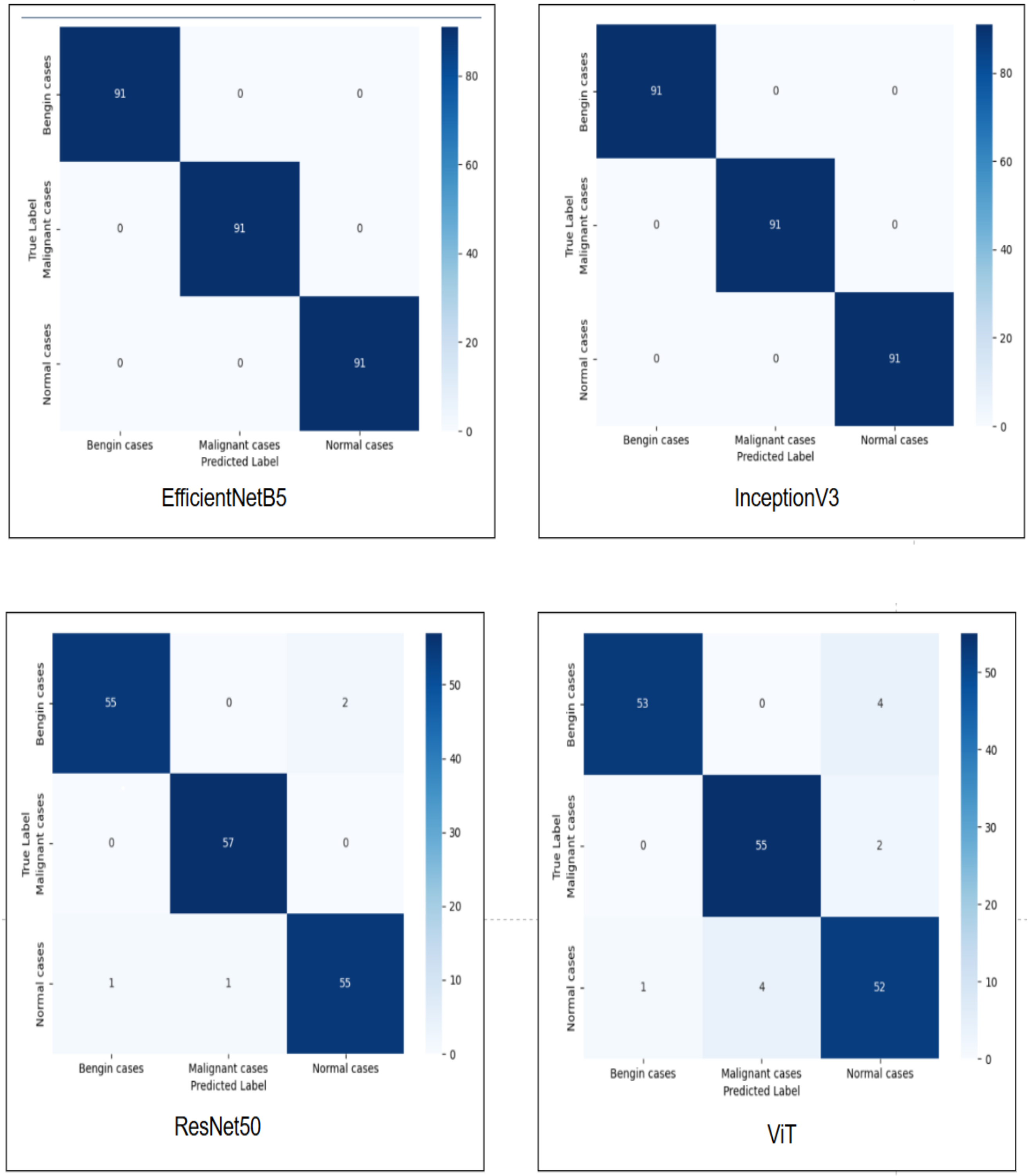
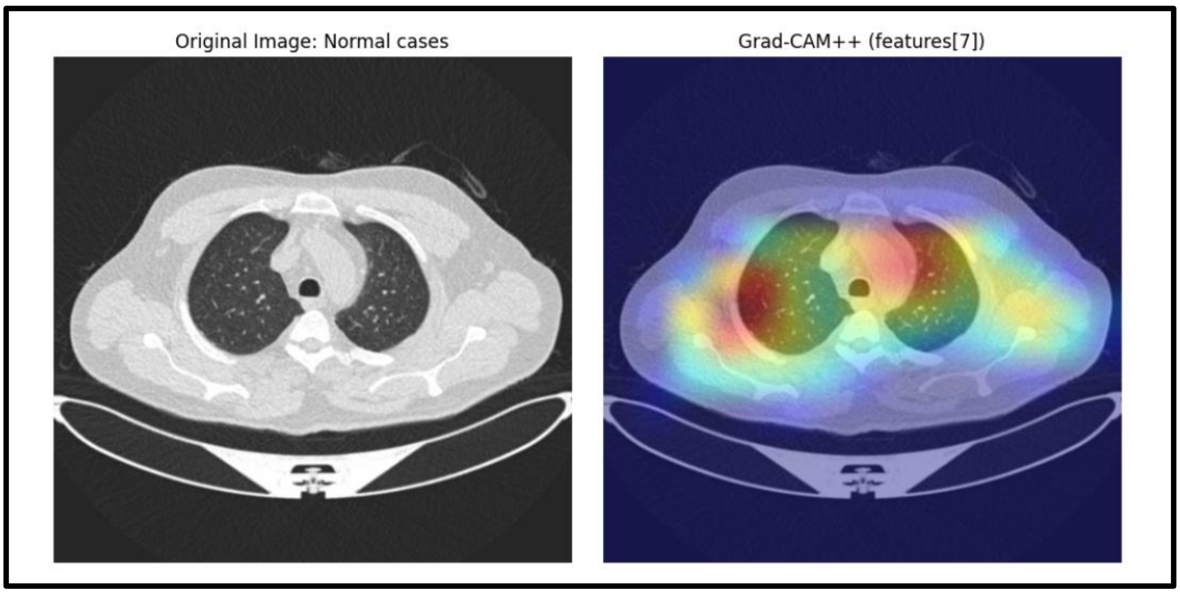


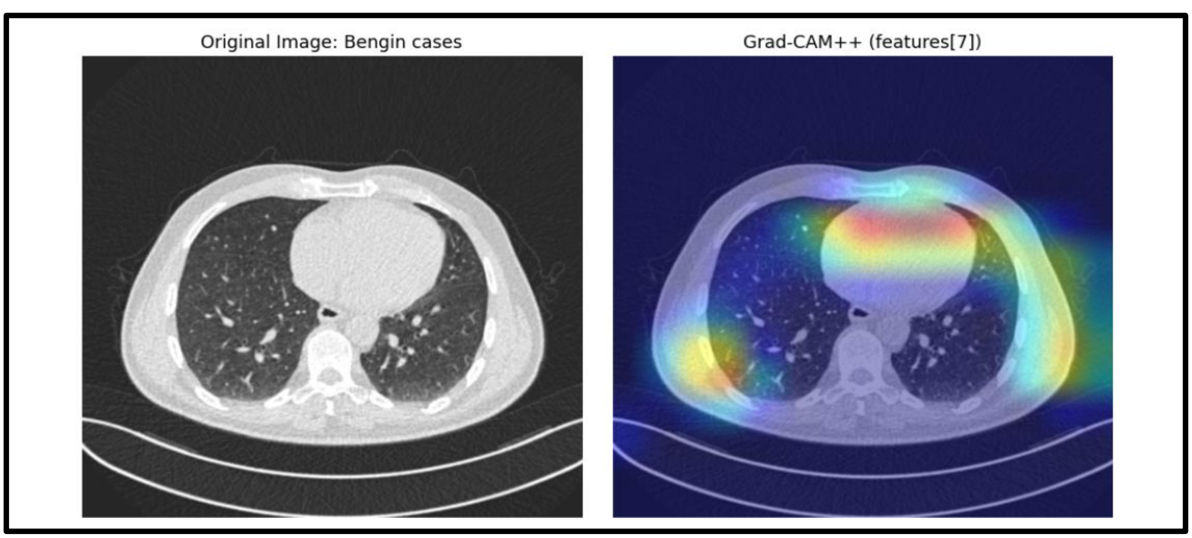
Figure 4.1.4: Confusion Matrix for models

4.2 PREDICTED RESULTS USING Grad-CAM

As shown in the studies, the goal of Explainable Artificial Intelligence methods is to provide interpretability to AI-predicted outcomes and make those outcomes acceptable to their users, especially in cases of high life-saving importance, such as lung disease classification. Therefore, Grad-CAM is utilized to visualize model prediction outlines over features that are supportive to the clinician in making decisions concerning the margins. Grad-CAM generates heat maps over medical images with areas of importance.

Predicted results using Grad-CAM





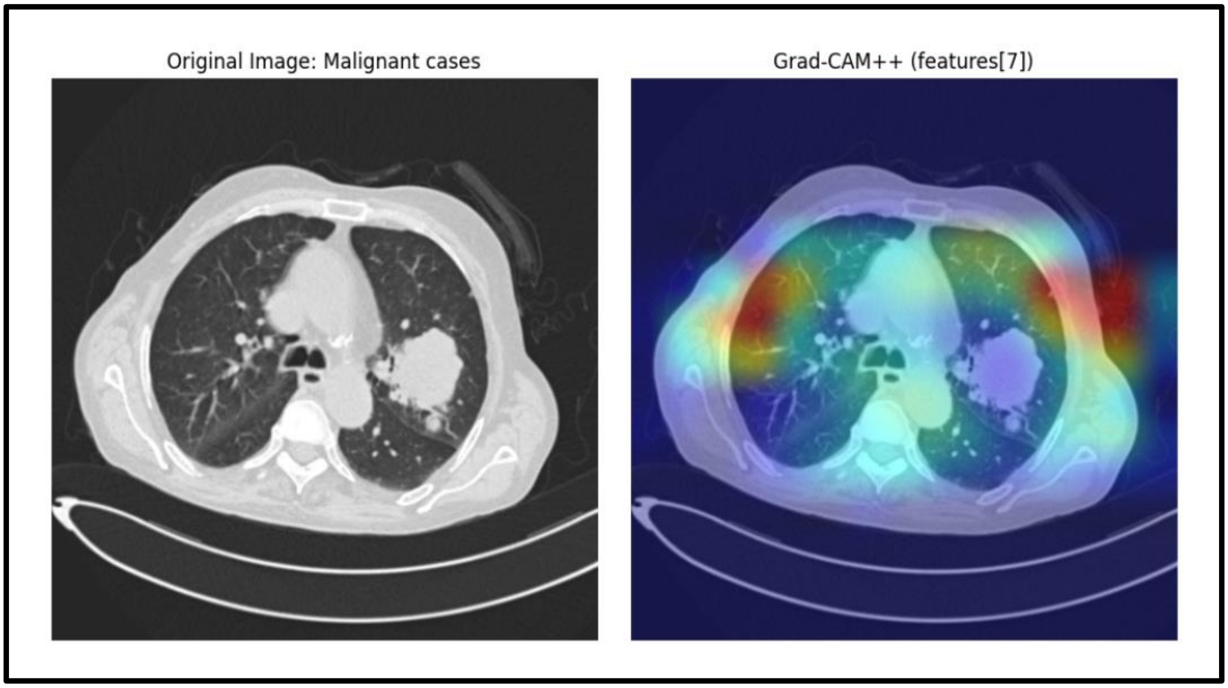


Figure 4.2 Predicted results using Grad-CAM

Incorporating XAI into lung case diagnosis achieve confidence in the minds of the users, mapping the areas in the image that affect the predictions. These studies indicate an interesting role of XAI in collaborating with the AI model to enhance human comprehension in the medical context.

Grad-CAM visualization provides for user-oriented interpretation that can help immensely in explicating how and why the model arrived at its predictions by pointing out the regions that were more decisive towards the classification and corroborating more of the model's decisions in a medical context while upgrading the model's trustworthiness. Red regions in Grad-CAM images identify the most probable zone of affection; yellow regions identify surrounding tissue of the lesions and blue part represents normal lung tissue.

It showcases Grad-CAM visualizations with predicted probabilities and overall class labels for lung disease prediction.

4.3 SUMMARY

This research has made an important advance in the manner of diagnosis and explanation for lung diseases using deep learning. One of the models we tried (EfficientNetB5) did incredibly well — it achieved 100% accuracy, with great precision/recall/F1-scores. The new thing in this work is the use of explainable AI tools such as Grad-CAM, which allowed us to literally see (actual visualizations) what a model was concentrating on when making its predictions. Also, data augmentation and balanced preprocessing were used over these to give the model better generalization ability and accuracy.

This work contributes to the gap between accuracy and interpretability and could serve as useful reference for future lung disease detection using deep learning.

CHAPTER 5

CONCLUSION AND FUTURE SCOPE

5.1 CONCLUSION

This project is a crossroads in using artificial intelligence to diagnose lung diseases. It uses EfficientNetB5 and Explainable XAI techniques to make a balance between giving high accuracy in diagnosis and remaining understandable. The algorithm used in research gains accuracy of up to 99% across three different classes for lung conditions. This has almost proved that the method can be depended on and is highly feasible. By Conjoin XAI interpretability methods, such as Grad-CAM, the model provides clear explanations of its decisions. This feature offers both dermatologists and patients a greater understanding of how their decisions are being made. No longer just a clinical tool, now the XAI system is fully- bear a hand with patients in their care.

5.2 FINDINGS AND CONTRIBUTION

In patient-centered care, it is bounden upon us to demand personalized explanation. This fills the absence between advanced AI technology and its actual application. Our results show that technical innovation and transparency are required to work collaboratively. This will settle the direction for further XAI work, applications around lung disease diagnosis and other medical terms. This XAI solution in practice opens the way for it to be approved by health care at large.

The research findings were: the substantial addition to Explainable XAI in the explanation of lung diagnoses and treatment. In my experiments, the most overpowering result is EfficientNetB5. An accuracy level of 99% was obtained for lung pathology studying in three categories. This work contributed significantly by applying XAI-Grad-CAM techniques to provide model-predicted explanations from the patient-centric perspective. Communicate the results in a way that is clear to both doctors and patients.

5.3 LIMITATIONS

Dataset Specificity: the model was trained and validated on the Kaggle dataset was collected from one hospital (Iraq-Oncology Teaching Hospital). This may not fully represent diverse patient demographics. Validation on larger, more diverse dataset is required for ensuring robustness.

Limited modalities: This research use only CT scans images. X-ray, PET, or MRI modalities does not include in this research. Combining all clinical information such as patient history, genetic data, or lab reports with the CT scan could improve prediction.

Computational complexity: Implementation of heavy models in real time like EfficientNetB5, InceptionV3, ResNet50 in hospitals or on low-resource devices is a challenge.

5.4 FUTURE SCOPE

We can apply a framework in medical fields such as radiology, cardiology, or ophthalmology. Accurate and intelligible AI systems are also necessary. Integrating real-time patient feedback into the system could improve personalization. This model would adjust and adapt its predictions over time. Future work using advanced XAI techniques and hybrid AI models can improve effectiveness in medical diagnosis.

Combining more varied and larger datasets will ensure the resilience and adaptability of the framework. Future work may also discover the use of advanced XAI techniques and hybrid AI models to advanced improve the transparency and effectiveness of medical diagnostics.

REFERENCES

[1] Wankhade, S., & Vigneshwari, S. (2023). A novel hybrid deep learning method for early detection of lung cancer using neural networks. Healthcare Analytics, 3, 100195.

[2] Baranwal, N., Doravari, P., & Kachhoria, R. (2022). Classification of histopathology images of lung cancer using convolutional neural network (CNN). In Disruptive Developments in Biomedical Applications (pp. 75-89). CRC Press.

[3] SHARIFF, V., Chiranjeevi, P., & Krishna, M. A. (2023). An analysis on advances in lung cancer diagnosis with medical imaging and deep learning techniques: Challenges and opportunities. Journal of Theoretical and Applied Information Technology, 101(17), 7083-7095.

[4] Heuvelmans, M. A., van Ooijen, P. M., Ather, S., Silva, C. F., Han, D., Heussel, C. P., ... & Oudkerk, M. (2021). Lung cancer prediction by Deep Learning to identify benign lung nodules. Lung cancer, 154, 1-4.

[5] Chaturvedi, P., Jhamb, A., Vanani, M., & Nemade, V. (2021, March). Prediction and classification of lung cancer using machine learning techniques. In IOP conference series: materials science and engineering (Vol. 1099, No. 1, p. 012059). IOP Publishing.

[6] Reddy, N. S., & Khanaa, V. (2023). Intelligent deep learning algorithm for lung cancer detection and classification. Bulletin of Electrical Engineering and Informatics, 12(3), 1747-1754.

[7] Shatnawi, M. Q., Abuein, Q., & Al-Quraan, R. (2025). Deep learning-based approach to diagnose lung cancer using CT-scan images. Intelligence-Based Medicine, 11, 100188.

[8] Han, Y., Ma, Y., Wu, Z., Zhang, F., Zheng, D., Liu, X., ... & Guo, X. (2021). Histologic subtype classification of non-small cell lung cancer using PET/CT images. European journal of nuclear medicine and molecular imaging, 48(2), 350-360.

[9] Venkatesh, C., Chinna Babu, J., Kiran, A., Nagaraju, C. H., & Kumar, M. (2024). A hybrid model for lung cancer prediction using patch processing and deeplearning on CT images. Multimedia Tools and Applications, 83(15), 43931-43952.

[10]Chen, Y., Yang, H., Cheng, Z., Chen, L., Peng, S., Wang, J., ... & Ke, Z. (2022). A whole-slide image (WSI)-based immunohistochemical feature prediction system improves the subtyping of lung cancer. Lung Cancer, 165, 18-27.

[11]Gayap, H. T., & Akhloufi, M. A. (2024). Deep machine learning for medical diagnosis, application to lung cancer detection: a review. BioMedInformatics, 4(1), 236-284.

[12] ZMasud, M., Sikder, N., Nahid, A. A., Bairagi, A. K., & AlZain, M. A. (2021). A machine learning approach to diagnosing lung and colon cancer using a deep learning-based classification framework. Sensors, 21(3), 748.

[13] Jiang, X., Hu, Z., Wang, S., & Zhang, Y. (2023). Deep Learning for Medical Image-Based Cancer diagnosis. Cancers, 15 (14), 3608.

[14] Chaunzwa, T. L., Hosny, A., Xu, Y., Shafer, A., Diao, N., Lanuti, M., ... & Aerts, H. J. (2021). Deep learning classification of lung cancer histology using CT images. Scientific reports, 11(1), 1-12.

[15] Said, Y., Alsheikhy, A. A., Shawly, T., & Lahza, H. (2023). Medical images segmentation for lung cancer diagnosis based on deep learning architectures. Diagnostics, 13(3), 546.

[16] Tan, S. L., Selvachandran, G., Paramesran, R., & Ding, W. (2025). Lung cancer detection systems applied to medical images: a state-of-the-art survey. Archives of Computational Methods in Engineering, 32(1), 343-380.

[17] Choi, Y., Aum, J., Lee, S. H., Kim, H. K., Kim, J., Shin, S., ... & Lee, H. Y. (2021). Deep learning analysis of CT images reveals high-grade pathological features to predict survival in lung adenocarcinoma. cancers, 13(16), 4077.

[18] Pandit, B. R., Alsadoon, A., Prasad, P. W. C., Al Aloussi, S., Rashid, T. A., Alsadoon, O. H., & Jerew, O. D. (2023). Deep learning neural network for lung cancer classification: enhanced optimization function. Multimedia Tools and Applications, 82(5), 6605-6624.

[19] Shafi, I., Din, S., Khan, A., Díez, I. D. L. T., Casanova, R. D. J. P., Pifarre, K. T., & Ashraf, I. (2022). An effective method for lung cancer diagnosis from ct scan using deep learning-based support vector network. Cancers, 14(21), 5457.

[20] Hosseini, S. H., Monsefi, R., & Shadroo, S. (2024). Deep learning applications for lung cancer diagnosis: a systematic review. Multimedia Tools and Applications, 83(5), 14305-14335.

[21] Thanoon, M. A., Zulkifley, M. A., Mohd Zainuri, M. A. A., & Abdani, S. R. (2023). A review of deep learning techniques for lung cancer screening and diagnosis based on CT images. Diagnostics, 13(16), 2617.

[22] Yang, H., Chen, L., Cheng, Z., Yang, M., Wang, J., Lin, C., ... & Li, W. (2021). Deep learning-based six-type classifier for lung cancer and mimics from histopathological whole slide images: a retrospective study. BMC medicine, 19(1), 80.

[23] Srivastava, D., Srivastava, S. K., Khan, S. B., Singh, H. R., Maakar, S. K., Agarwal, A. K., ... & Albalawi, E. (2023). Early Detection of Lung Nodules Using a Revolutionized Deep Learning Model. Diagnostics, 2023, 13, 3485.

[24] Nam, J. G., Park, S., Park, C. M., Jeon, Y. K., Chung, D. H., Goo, J. M., ... & Kim, H. (2022). Histopathologic basis for a chest CT deep learning survival prediction model in patients with lung adenocarcinoma. Radiology, 305(2), 441-451.

[25] Archana, R., & Jeevaraj, P. E. (2024). Deep learning models for digital image processing: a review. Artificial Intelligence Review, 57(1), 11.

[26] Rajasekar, V., Vaishnnave, M. P., Premkumar, S., Sarveshwaran, V., & Rangaraaj, V. (2023). Lung cancer disease prediction with CT scan and histopathological images feature analysis using deep learning techniques. Results in Engineering, 18, 101111.

[27] Mohalder, R. D., Sarkar, J. P., Hossain, K. A., Paul, L., & Raihan, M. (2021, September). A deep learning based approach to predict lung cancer from histopathological images. In 2021 international conference on electronics, communications and information technology (ICECIT) (pp. 1-4). IEEE.

[28] Mohamed, T. I., Oyelade, O. N., & Ezugwu, A. E. (2023). Automatic detection and classification of lung cancer CT scans based on deep learning and ebola optimization search algorithm. Plos one, 18(8), e0285796.

APPENDIX A

Dataset Collection and Categorization

The dataset (IQ-OTH/NCCD) for Cancer Diseases includes CT scans of patients diagnosed with lung cancer in different stages. The dataset contains a total of 1190 images representing CT scan slices of 110 cases. The CT scans were originally collected in DICOM format. A total 110 images categorized into normal, benign, and malignant classes for training and testing. The whole process described below with figures.

Downloading the Dataset

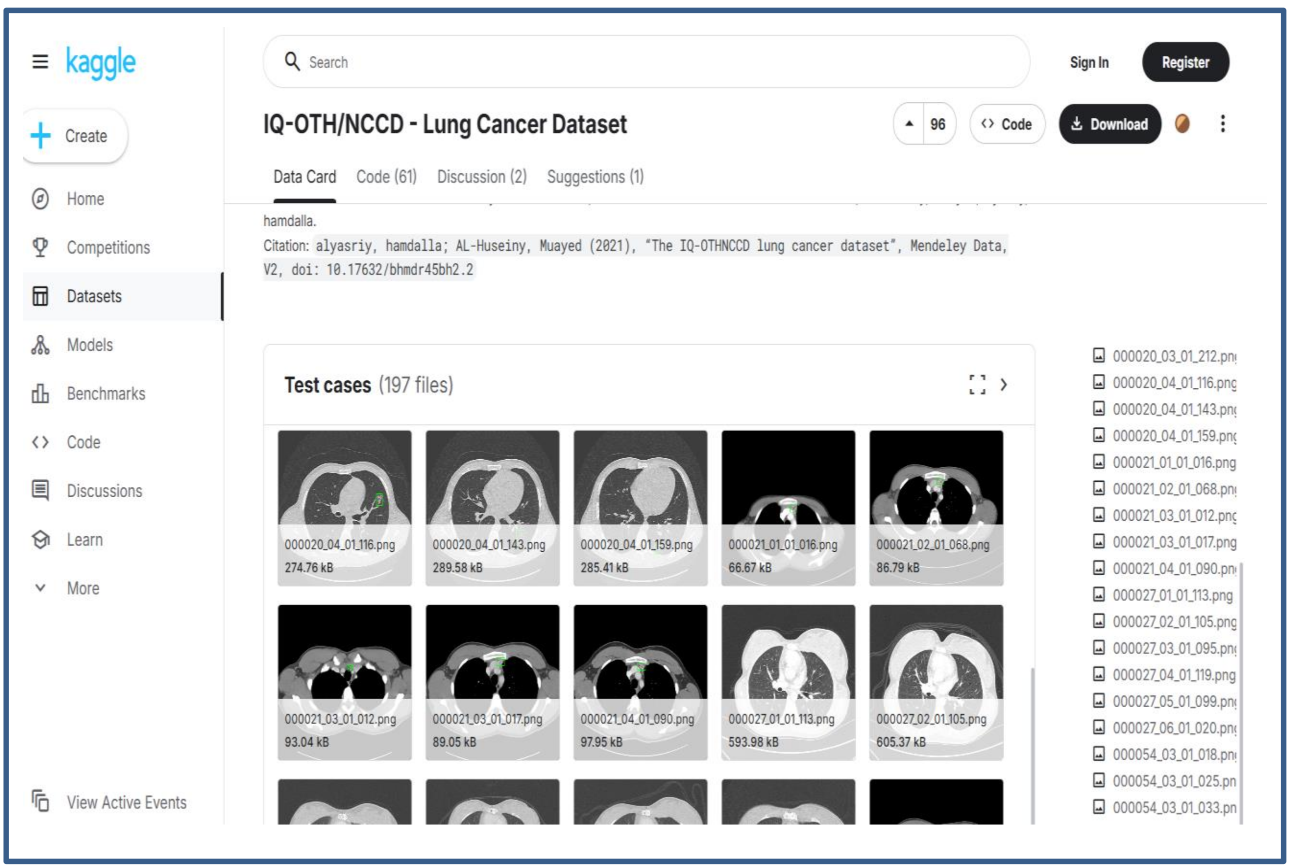


Figure A.1: Downloading the Dataset

JPG images were classified and kept into Normal, Benign and Malignant folder

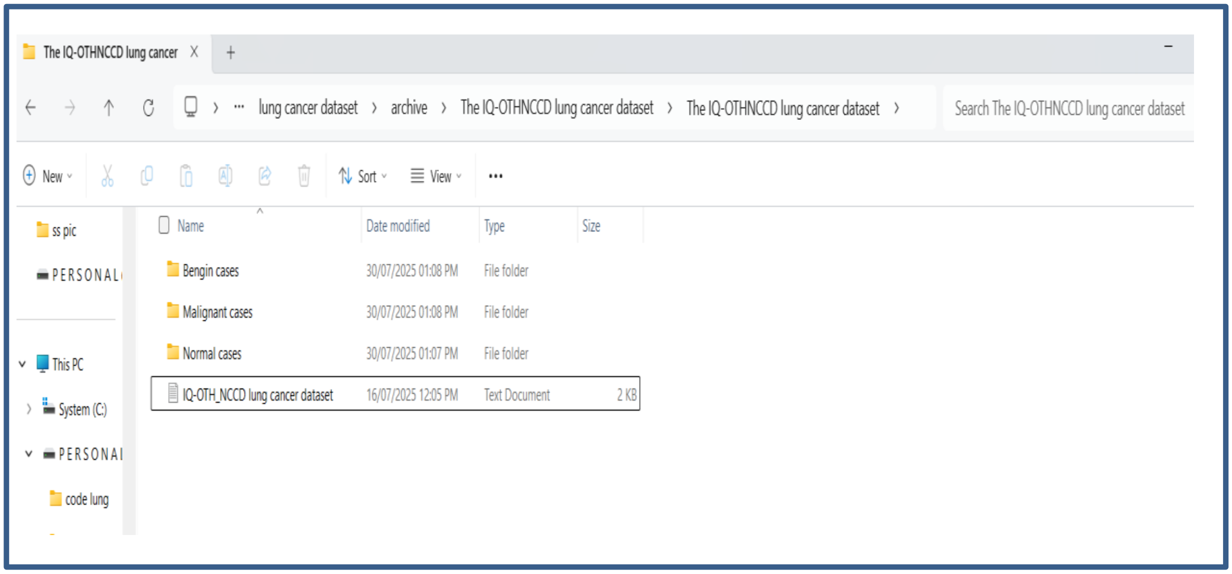


Figure A.2: Classifying the Dataset

Benign JPG CT scan images in folder for training

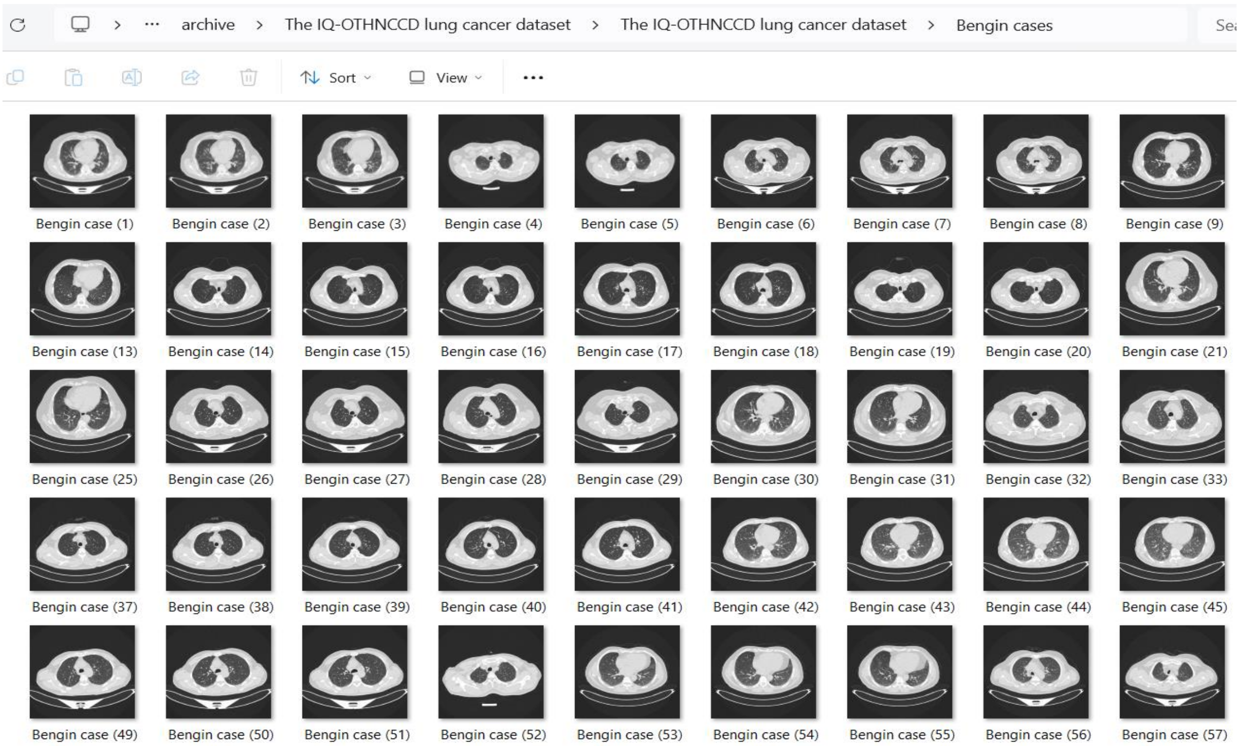


Figure A.2: Benign images

APPENDIX B

Step by Step Improvement

Using data augmentation with 10 epochs the training Accuracy and Loss are 0.9960 and 0.189

```
Starting model training...
Epoch 1/50: Train Acc=0.8281, Val Acc=0.9670, Train Loss=0.4614, Val Loss=0.1873, Learning Rate=0.000100
Epoch 2/50: Train Acc=0.9483, Val Acc=0.9451, Train Loss=0.1594, Val Loss=0.1555, Learning Rate=0.000100
Epoch 3/50: Train Acc=0.9790, Val Acc=0.9890, Train Loss=0.0754, Val Loss=0.0759, Learning Rate=0.000100
Epoch 4/50: Train Acc=0.9911, Val Acc=0.9927, Train Loss=0.0432, Val Loss=0.0207, Learning Rate=0.000100
Epoch 5/50: Train Acc=0.9927, Val Acc=0.9634, Train Loss=0.0355, Val Loss=0.0816, Learning Rate=0.000100
Epoch 6/50: Train Acc=0.9863, Val Acc=0.9963, Train Loss=0.0527, Val Loss=0.0048, Learning Rate=0.000100
Epoch 7/50: Train Acc=0.9952, Val Acc=0.9963, Train Loss=0.0197, Val Loss=0.0050, Learning Rate=0.000100
Epoch 8/50: Train Acc=0.9976, Val Acc=0.9744, Train Loss=0.0107, Val Loss=0.0436, Learning Rate=0.000100
Epoch 9/50: Train Acc=0.9879, Val Acc=0.9963, Train Loss=0.0346, Val Loss=0.0100, Learning Rate=0.000100
Epoch 10/50: Train Acc=0.9960, Val Acc=0.9817, Train Loss=0.0189, Val Loss=0.0441, Learning Rate=0.000100
```

Figure B.2: 1st step training

When the epoch set to 23 the training accuracy and loss is improved to 0.9992 and 0.0042

```
EfficientNetB5 smote 0.9ds 50epoch.ipynb ●   InceptionV3 smote 0.9ds 50epoch.ipynb
D: > code lung > EfficientNetB5 smote 0.9ds 50epoch.ipynb > M↓ Model Training for Lung Cancer Detection > M↓ Modeling and Training > def trai
Generate  + Code  + Markdown  | Run All  ···
Epoch 14/50: Train Acc=0.9919, Val Acc=1.0000, Train Loss=0.0220, Val Loss=0.0065, Learning Rate=0.000100
Epoch 15/50: Train Acc=0.9976, Val Acc=1.0000, Train Loss=0.0135, Val Loss=0.0024, Learning Rate=0.000100
Epoch 16/50: Train Acc=0.9952, Val Acc=0.9890, Train Loss=0.0192, Val Loss=0.0289, Learning Rate=0.000100
Epoch 17/50: Train Acc=0.9927, Val Acc=0.9890, Train Loss=0.0187, Val Loss=0.0252, Learning Rate=0.000100
Epoch 18/50: Train Acc=0.9960, Val Acc=1.0000, Train Loss=0.0214, Val Loss=0.0026, Learning Rate=0.000100
Epoch 19/50: Train Acc=1.0000, Val Acc=1.0000, Train Loss=0.0023, Val Loss=0.0023, Learning Rate=0.000100
Epoch 20/50: Train Acc=1.0000, Val Acc=1.0000, Train Loss=0.0009, Val Loss=0.0014, Learning Rate=0.000100
Epoch 21/50: Train Acc=1.0000, Val Acc=1.0000, Train Loss=0.0008, Val Loss=0.0013, Learning Rate=0.000100
Epoch 22/50: Train Acc=1.0000, Val Acc=0.9963, Train Loss=0.0017, Val Loss=0.0044, Learning Rate=0.000100
Epoch 23/50: Train Acc=0.9992, Val Acc=0.9927, Train Loss=0.0042, Val Loss=0.0137, Learning Rate=0.000100
```

Figure B.2: 2nd step training

When the epoch set to 49 the training accuracy and loss is improved to 1.0000 and 0.0020

```
EfficientNetB5 smote 0.9ds 50epoch.ipynb ●   InceptionV3 smote 0.9ds 50epoch.ipynb
D: > code lung > EfficientNetB5 smote 0.9ds 50epoch.ipynb > M↓ Model Training for Lung Cancer Detection > M↓ Modeling and Training > def train_m
Generate  + Code  + Markdown  | Run All  ···
Epoch 20/50: Train Acc=1.0000, Val Acc=1.0000, Train Loss=0.0009, Val Loss=0.0014, Learning Rate=0.000100
Epoch 21/50: Train Acc=1.0000, Val Acc=1.0000, Train Loss=0.0008, Val Loss=0.0013, Learning Rate=0.000100
Epoch 22/50: Train Acc=1.0000, Val Acc=0.9963, Train Loss=0.0017, Val Loss=0.0044, Learning Rate=0.000100
Epoch 23/50: Train Acc=0.9992, Val Acc=0.9927, Train Loss=0.0042, Val Loss=0.0137, Learning Rate=0.000100
Epoch 24/50: Train Acc=1.0000, Val Acc=1.0000, Train Loss=0.0008, Val Loss=0.0021, Learning Rate=0.000100
...
Epoch 48/50: Train Acc=0.9968, Val Acc=0.9963, Train Loss=0.0173, Val Loss=0.0062, Learning Rate=0.000100
Epoch 49/50: Train Acc=1.0000, Val Acc=1.0000, Train Loss=0.0020, Val Loss=0.0005, Learning Rate=0.000100
```

Figure B.3: 3rd step training

Final improved Model Accuracy

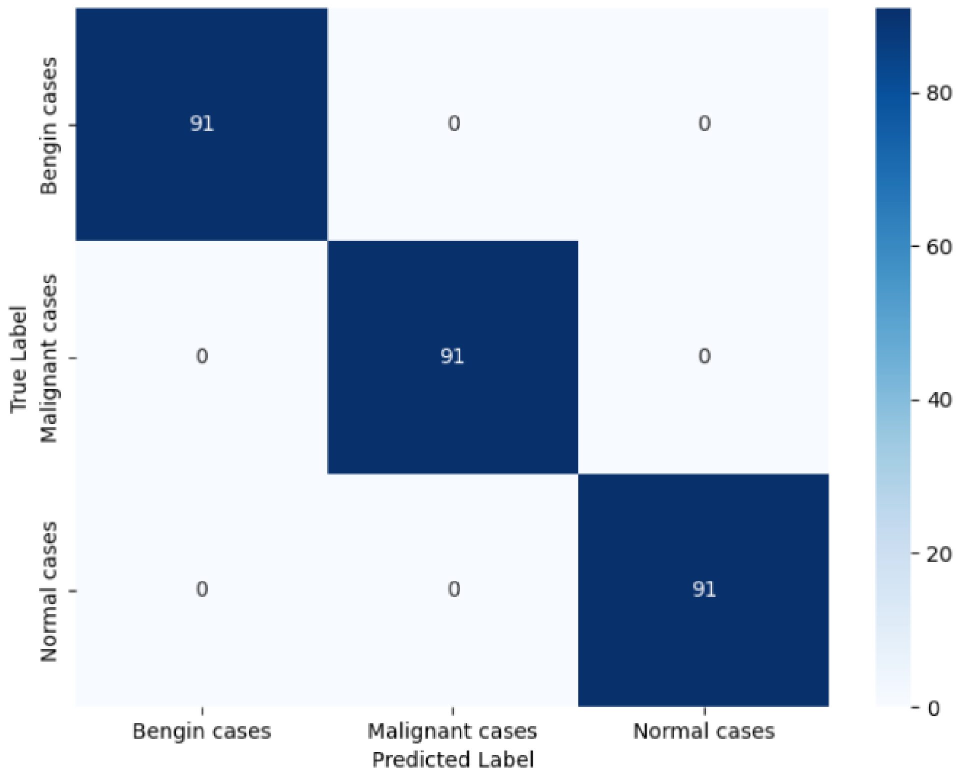


Figure B.4: Final improved Model Accuracy

APPENDIX C

GitHub Link : https://github.com/JarinSultana28/Thesis-Lung-condition.git

Sample of Implemented Code

Importing all the modules required

⏳ Loading Basic Libraries 📦 📚

```python
import os
import gc
import sys
import cv2
import numpy as np
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt

import torch
import torch.nn as nn
from torch.optim import Adam
from torchvision import models, transforms
from torch.utils.data import Dataset, DataLoader

from imblearn.over_sampling import SMOTE
from sklearn.model_selection import train_test_split
from sklearn.metrics import classification_report, confusion_matrix, accuracy_score

sys.path.append('../')
```

Figure C.1: Importing Modules

Image preprocessing

Preprocessing

Listing Classes Dynamically from `dataset_path`

```python
try:
    classes = sorted([d for d in os.listdir(dataset_path) if os.path.isdir(os.path.join(dataset_path, d))])
    if not classes:
        raise ValueError("No subdirectories found in the dataset path. Ensure your dataset is organized into class folders.")
    print(f"Discovered classes: {classes}")
except FileNotFoundError:
    print(f"Error: Dataset path '{dataset_path}' not found. Please check the path.")
    exit()
except ValueError as e:
    print(f"Error: {e}")
    exit()
```

[4]

```
Discovered classes: ['Bengin cases', 'Malignant cases', 'Normal cases']
```

Figure C.2: Image preprocessing

Defining Functions to Visualize Data and Class Distribution

EfficientNetB5 smote 0.9ds 50epoch.ipynb ● InceptionV3 smote 0.9ds 50epoch.ipynb

D: > code lung > EfficientNetB5 smote 0.9ds 50epoch.ipynb > M↓ Model Training for Lung Cancer Detection > M↓ Preprocessing > M↓ Defining Functions to Visualize Data and Cla

Generate + Code + Markdown | Run All ···

```python
# Function to visualize class distribution
def visualize_class_distribution(labels_data, class_names, title_suffix=""):
    plt.figure(figsize=(12, 6))
    sns.countplot(x=labels_data, hue=labels_data, palette="viridis", legend=False)
    # plt.title(f"Class Distribution {title_suffix}")
    plt.xlabel("Class Name")
    plt.ylabel("Count")
    plt.xticks(ticks=range(len(class_names)), labels=class_names, rotation=45, ha='right')
    plt.tight_layout()
    plt.show()
    plt.close('all')
    gc.collect()

# Function to visualize example images from the loaded dataset
def visualize_loaded_example(data_array, labels_array, class_names, num_examples_per_class=1):
    print("\nVisualizing example images from loaded dataset...")
    n_rows = num_examples_per_class
    n_cols = len(class_names)
    fig, axes = plt.subplots(n_rows, n_cols, figsize=(4 * n_cols, 4 * n_rows))

    if n_rows == 1 and n_cols == 1:
        axes = np.array([[axes]])
    elif n_rows == 1:
        axes = axes[np.newaxis, :]
    elif n_cols == 1:
        axes = axes[:, np.newaxis]

    for i, cls_name in enumerate(class_names):
        class_indices = np.where(labels_array == i)[0]

        # Handle cases where a class might have fewer examples than num_examples_per_class
        if len(class_indices) < num_examples_per_class:
            print(f"Warning: Class '{cls_name}' has only {len(class_indices)} images, requested {num_examples_per_class}.")
            selected_indices = class_indices # Use all available images
        else:
            selected_indices = np.random.choice(class_indices, num_examples_per_class, replace=False)

        for j, idx in enumerate(selected_indices):
            img = data_array[idx] #

            ax = axes[j][i]
            ax.imshow(img)
            ax.set_title(f"{cls_name}")
            ax.axis("off")

    plt.tight_layout()
    plt.show()
    plt.close('all')
    gc.collect()
```

[5]

Figure C.3: Visualize Data and Class Distribution

Modeling and Training Dataset

EfficientNetB5 smote 0.9ds 50epoch.ipynb ● InceptionV3 smote 0.9ds 50epoch.ipynb

D: > code lung > EfficientNetB5 smote 0.9ds 50epoch.ipynb > M↓ Model Training for Lung Cancer Detection > M↓ Preprocessing > M↓ Defining Functions to Visualiz

Generate + Code + Markdown | Run All ···

Modeling and Training

Preparing Dataloaders

```python
transform = transforms.Compose([
    transforms.ToPILImage(), # Convert NumPy array to PIL Image for torchvision transforms
    transforms.ToTensor(), # Converts to tensor and scales to [0,1]
    transforms.Normalize(mean=[0.485, 0.456, 0.406], std=[0.229, 0.224, 0.225]) # Normalize
])

class EyeDiseaseDataset(Dataset):
    def __init__(self, data, labels, transform=None):
        self.data = data
        self.labels = labels
        self.transform = transform

    def __len__(self):
        return len(self.data)

    def __getitem__(self, idx):
        img = self.data[idx]
        label = self.labels[idx]

        # Images are already in RGB from initial load
        if self.transform:
            img = self.transform(img)
        else:
            img = torch.tensor(img).permute(2, 0, 1).float() / 255.0
        return img, torch.tensor(label, dtype=torch.long)

train_dataset = EyeDiseaseDataset(train_data, train_labels, transform=transform)
test_dataset = EyeDiseaseDataset(test_data, test_labels, transform=transform)

del train_data, train_labels, test_data, test_labels
gc.collect()

train_loader = DataLoader(train_dataset, batch_size=batch_size, shuffle=True, num_workers=os.cpu_count() // 2 or 1)
test_loader = DataLoader(test_dataset, batch_size=batch_size, shuffle=False, num_workers=os.cpu_count() // 2 or 1)
print("Datasets and DataLoaders prepared.")
```

[9]

Figure C.4: Modeling and Training Dataset

Using `Grad-CAM++ visualization`

XAI

```python
class GradCam:
    def __init__(self, model, target_layer):
        self.model = model
        self.target_layer = target_layer
        self.gradients = None
        self.activations = None # Activations will now be kept in the graph

        # Register forward hook to capture activations. IMPORTANT: Do NOT detach here.
        # We need the activations to be part of the computation graph for gradient computation later.
        self.hook_forward = self.target_layer.register_forward_hook(self._save_activation)

    def _save_activation(self, module, input, output):
        # Store activations; they remain attached to the graph for gradient computation later.
        self.activations = output

    def generate_heatmap(self, gradients, activations, device):
        # --- Grad-CAM++ specific calculation for weights (alphas) ---
        # Gradients passed here are the first-order gradients (dY/dA) from __call__

        # Ensure tensors are on the correct device
        if activations.device != device:
            activations = activations.to(device)
        if gradients.device != device:
            gradients = gradients.to(device)

        # Calculate alpha_kc as per Grad-CAM++ paper (simplified for practical implementation)
        # This involves element-wise operations on gradients.

        # Element-wise squared gradients
        grad_squared = gradients.pow(2)
        # Element-wise cubed gradients
        grad_cubed = gradients.pow(3)

        # Sum gradients across spatial dimensions (height and width) for normalization
        sum_grad = gradients.sum(dim=(-1,-2), keepdim=True) # Sum over H, W
```

Figure C.5: `Grad-CAM++ visualization`

LIBRARY CLEARNCE

PLAGLARISM REPORT

212-35-742

ORIGINALITY REPORT

14%	8%	11%	3%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	Mohammad Q. Shatnawi, Qusai Abuein, Romesaa Al-Quraan. "Deep Learning-Based Approach to Diagnose Lung Cancer Using CT-Scan Images", Intelligence-Based Medicine, 2024 Publication	3%
2	Submitted to Daffodil International University Student Paper	2%
3	dspace.daffodilvarsity.edu.bd:8080 Internet Source	1%
4	journals.plos.org Internet Source	1%
5	pmc.ncbi.nlm.nih.gov Internet Source	1%
6	Shalini Wankhade, Vigneshwari S.. "A novel hybrid deep learning method for early detection of lung cancer using neural networks", Healthcare Analytics, 2023 Publication	<1%
7	Pushpa Choudhary, Sambit Satpathy, Arvind Dagur, Dhirendra Kumar Shukla. "Recent Trends in Intelligent Computing and Communication", CRC Press, 2025 Publication	<1%
8	Rohit Tanwar, Surbhi Bhatia Khan, Varun Sapra, Neelu Jyoti Ahuja. "Artificial Intelligence and Machine Learning - An Intelligent Perspective of Emerging Technologies", CRC Press, 2023 Publication	<1%

9	export.arxiv.org Internet Source	<1 %
10	Thompson Stephan. "Artificial Intelligence in Medicine", CRC Press, 2024 Publication	<1 %
11	Submitted to University of Hull Student Paper	<1 %
12	Shankar Babu, Mahesh Babu Kota. "Synergies in Smart and Virtual Systems using computational intelligence", CRC Press, 2025 Publication	<1 %
13	www.frontiersin.org Internet Source	<1 %
14	www.mdpi.com Internet Source	<1 %
15	ijmrset.com Internet Source	<1 %
16	Fatih Ertam, Ilhan Firat Kilincer. "Predicting and diagnosis of COVID-19 based on IoT and machine learning algorithm", Elsevier BV, 2024 Publication	<1 %
17	Rehan Raza, Fatima Zulfiqar, Muhammad Owais Khan, Muhammad Arif, Atif Alvi, Muhammad Aksam Iftikhar, Tanvir Alam. "Lung-EffNet: Lung cancer classification using EfficientNet from CT-scan images", Engineering Applications of Artificial Intelligence, 2023 Publication	<1 %
18	Agughasi Victor Ikechukwu, S. Murali. "xAI: An Explainable AI Model for the Diagnosis of COPD from CXR Images", 2023 IEEE 2nd International Conference on Data, Decision and Systems (ICDDS), 2023	<1 %

Publication

19	arxiv.org Internet Source	<1 %
20	digital.library.unt.edu Internet Source	<1 %
21	Sarmad Maqsood, Robertas Damaševičius, Rytis Maskeliūnas, Nils D. Forkert, Shahab Haider, Shahid Latif. "Csec-net: a novel deep features fusion and entropy-controlled firefly feature selection framework for leukemia classification", Health Information Science and Systems, 2024 Publication	<1 %
22	link.springer.com Internet Source	<1 %
23	Submitted to Higher Education Commission Pakistan Student Paper	<1 %
24	ijsrem.com Internet Source	<1 %
25	S Lalitha, B V Baiju, Sandeep Kumar Mathivanan, Partheeban Nagappan, Chandra Shakher Tyagi, Saurav Mallik. "Detection and Classification of Cervical Cancer Using Optimized Deep Learning Approach", Cold Spring Harbor Laboratory, 2025 Publication	<1 %
26	repository.up.ac.za Internet Source	<1 %
27	www.coursehero.com Internet Source	<1 %
28	www.mediatebc.com Internet Source	<1 %
29	Fadi Al-Turjman. "AI-Powered IoT for COVID-19", CRC Press, 2020	<1 %

Publication

30	easy.dans.knaw.nl Internet Source	<1 %
31	ijirt.org Internet Source	<1 %
32	papers.miccai.org Internet Source	<1 %
33	"Inventive Systems and Control", Springer Science and Business Media LLC, 2023 Publication	<1 %
34	Submitted to Arab Open University Student Paper	<1 %
35	file.techscience.com Internet Source	<1 %
36	www.docstoc.com Internet Source	<1 %
37	www.drimtelerad.net Internet Source	<1 %
38	"Medical Image Computing and Computer Assisted Intervention – MICCAI 2018", Springer Nature America, Inc, 2018 Publication	<1 %
39	Nafe Muhtasim, Umma Hany, Tahmina Islam, Nusrat Nawreen, Abdullah Al Mamun. "Artificial intelligence for detection of lung cancer using transfer learning and morphological features", The Journal of Supercomputing, 2024 Publication	<1 %
40	"Machine Learning for Cyber Security", Springer Science and Business Media LLC, 2025 Publication	<1 %
41	Mehrzadi, Hamed. "Deep Learning Approaches for Early Detection of Lung Cancer Using CT Scan Images", The University of Texas at San Antonio, 2024 Publication	<1 %
42	R. Indrakumari, T. Ganesh Kumar, D. Murugan, P.C. Sherimon. "Deep Learning in Medical Image Analysis - Recent Advances and Future Trends", CRC Press, 2024 Publication	<1 %
43	da Costa Neto, Alexandre Henrique. "Automatic Glaucoma Screening with Low-Cost Devices", Universidade de Tras-os-Montes e Alto Douro (Portugal), 2024 Publication	<1 %

Exclude quotes	Off	Exclude matches	Off
Exclude bibliography	Off		

ACCOUNT CLERANCE