Interpretable Deep Learning Model for Classification of Lung Nodules for Early Detection of Lung Cancer

Besufekad Negussie Gebregziabher

A Thesis Submitted to

The Department of Computer Science and Engineering

School Of Electrical Engineering and Computing (SOEEC)

Presented in Partial Fulfillment of the Requirement for the Degree of Master's in Computer Science and Engineering

Office of Graduate Studies

Adama Science and Technology University

September, 2024

Adama, Ethiopia

Interpretable Deep Learning Model for Classification of Lung Nodules for Early Detection of Lung Cancer

Besufekad Negussie Gebregziabher

Advisor: Dr. Sintayehu Hirpasa

A Thesis Submitted to

The Department of Computer Science and Engineering

School Of Electrical Engineering and Computing (SOEEC)

Presented in Partial Fulfillment of the Requirement for the Degree of Master's in Computer Science and Engineering

Office of Graduate Studies

Adama Science and Technology University

September, 2024

Adama, Ethiopia

DECLARATION

I declare that this thesis/dissertation proposal entitled "Interpretable Deep Learning Model for Classification of Lung Nodules for Early Detection of Lung Cancer" is my own work. That is, it has not been submitted for the award of any academic degree, diploma or certificate in any other university. All sources of materials that are used for this thesis have been duly acknowledged through citation.

Besufekad Negussie Gebregziabher	_____	_____
Name of student	Signature	Date

RECOMMENDATION

I/we, the advisor(s) of this thesis, hereby certify that I/we have read the revised version of the thesis entitled "Interpretable Deep Learning Model for Classification of Lung Nodules for Early Detection of Lung Cancer" prepared under my/our guidance by Besufekad Negussie Gebregziabher submitted in partial fulfillment of the requirements for the degree of Mater's of Science in Computer Science and Engineering. Therefore, I/we recommend the submission of revised version of the thesis to the department following the applicable procedures.

Dr. Sintayehu Hirpasa	_____	_____
Major Advisor/Supervisor	Signature	Date
_____	_____	_____
Co-advisor/Co-supervisor	Signature	Date

APPROVAL PAGE OF M.SC. THESIS

I/we, the advisors of the thesis entitled "Interpretable Deep Learning Model for Classification of Lung Nodules for Early Detection of Lung Cancer" and developed by Besufekad Negussie Gebregziabher, hereby certify that the recommendation and suggestions made by the board of examiners are appropriately incorporated into the final version of the thesis.

Dr. Sintayehu Hirpasa	_____	_____
Major Advisor	Signature	Date
_____	_____	_____
Co-advisor	Signature	Date

We, the undersigned, members of the Board of Examiners of the thesis by Besufekad Negussie Gebregziabher have read and evaluated the thesis entitled "Interpretable Deep Learning Model for Classification of Lung Nodules for Early Detection of Lung Cancer" and examined the candidate during open defense. This is, therefore, to certify that the thesis is accepted for partial fulfillment of the requirement of the degree of Mater's of Science in Computer Science and Engineering.

_____	_____	_____
Chairperson	Signature	Date
_____	_____	_____
Internal Examiner	Signature	Date
_____	_____	_____
External Examiner	Signature	Date

Finally, approval and acceptance of the thesis is contingent upon submission of its final copy to the Office of Postgraduate Studies (OPGS) through the Department Graduate Council (DGC) and School Graduate Committee (SGC).

_____	_____	_____
Department Head	Signature	Date
_____	_____	_____
School Dean	Signature	Date
_____	_____	_____
Office of Postgraduate Studies, Dean	Signature	Date

ACKNOWLEDGEMENT

First and foremost, I would like to express my deepest gratitude to God for His unwavering guidance and strength throughout this research journey. I also want to dedicate this work to the loving memory of my parents, my mother Azalech Mengesha and my father Nigussie Gebregziabher. Their love and support have been the foundation upon which I stand. I am forever grateful for their belief in me. My heartfelt appreciation goes to my dear siblings, Abel Nigussie, Kidest Nigussie, and Simegn Nigussie, and their families. Their encouragement and unwavering support have been a source of immense strength during this challenging but rewarding endeavor.

I am deeply indebted to my advisor, Dr. Sintayehu Hirpasa, as well as my instructors at ASTU for their invaluable guidance, mentorship, and support throughout this research project. Their expertise and dedication have been instrumental in shaping this work. I would also like to express my sincere thanks to S/r Askale Yacob, who assisted me in the data collection phase at the local hospital.

Furthermore, I am grateful to my friends for their companionship and support throughout this journey. Their encouragement and camaraderie have been a true blessing. Finally, I would like to acknowledge Adama Science and Technology University and all the resources that were made available to me during this research. These resources have been essential in facilitating the completion of this project.

This research would not have been possible without the contributions of all the individuals and institutions mentioned above. I am truly grateful for their support.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ACRONYMS AND ABBREVIATIONS

AI: Artificial intelligence

ANN: Artificial Neural Networks

AUC: Area Under the Curve

CAD: Computer Aided Diagnosis

CAT: Computed Axial Tomography

CBAM: Convolutional Block Attention Module

CIP: Cancer Imaging Program

CNN: Convolutional Neural Networks

CT: Computed Tomography

CV: Computer Vision

DataViz: Data Visualization

DICOM: Digital Imaging and Communication in Medicine

DL: Deep Learning

DSB2017: Data Science Bowl 2017

EDA: Exploratory Data Analysis

FN: False Negative

FP: False Positive

FNLCR: Frederick National Laboratory for Cancer Research

Grad-CAM: Gradient-weighted Class Activation Mapping

HU: Hounsfield Unit

IARC: International Agency for Research on Cancer

IML: Interpretable Machine Learning

LAC: Latin America and the Caribbean

LIDC: Lung Image Database Consortium

LIDC-IDRI: Lung Imaging Database Consortium-Image Database Resource Initiative

LIME: Local Interpretable Model-agnostic Explanations

LDCT: Low-Dose Computed Tomography

LIME: Local Interpretable Model-Agnostic Explanations

LRP: Layer-Wise Relevance Propagation

LUNA16: Lung Nodule Analysis 2016

LUNA22-ISMI: Lung Nodule Analysis 2022 - Intelligent Systems in Medical Imaging

ML: Machine learning

MRN: Medical Record Number

NCI: National Cancer Institute

NIfTI: Neuroimaging Informatics Technology Initiative

NLST: National Lung Screening Trial

NN: Neural Networks

NSCLC: Non-Small Cell Lung Cancer

OOM: Out Of Memory

PFI: Permutation Feature Importance

RLU: Rectified Linear Unit

RN-PDMP: Residual Neural and Partial Derivative Multilayer Perceptron

ROC-AUC: Receiver Operating Characteristic Curve's - Area Under the Curve

SCLC: Small Cell Lung Cancer

SHAP: Shapley Additive explanations

WHO: World Health Organization

XAI: Explainable Artificial Intelligence

ABSTRACT

Lung cancer is a leading cause of cancer-related deaths worldwide. Early detection through low-dose chest computed tomography (CT) scans is crucial for improving patient survival rates. This study investigates the application of interpretable deep learning models for classifying lung nodules in CT scans as benign or malignant, aiding in the early detection of lung cancer. We leverage 3D Convolutional Neural Networks (CNNs) trained on the LUNA22 ISMI dataset, which has 1,176 lung nodules with a collection of lung nodule annotations from anonymized CT scans. We compared four 3D CNN models with different architectures. One as a baseline model, the second one is based on 3D adaptations of well-established architecture known as AlexNet, the third one is our proposed 3D CNN model and the fourth is our proposed 3D CNN with CBAM integration. The CBAM module is inserted after the last convolutional block in a deep learning model. This allows it to leverage the rich feature representations learned by the preceding layers and apply attention mechanisms to enhance the most informative features. Our 3D CNN with CBAM model outperformed the other three with an accuracy of 94.06%, AUC of 98.84% and F1-Score of 95.56%. Our results demonstrate promising performance in binary classification of lung nodules. Considering the criticality of explainability in medical applications, we employed 3D Grad-CAM, a technique for interpreting the predictions of our 3D CNNs. This approach provides visual insights into the regions within lung nodule images that contribute most significantly to the model's classification decision. This interpretability fosters trust and facilitates the adoption of deep learning models in clinical settings. Our findings suggest that interpretable deep learning holds promise for improving the accuracy and efficiency of early lung cancer detection through CT scans. By leveraging interpretable models, we can potentially enhance clinical decision-making and ultimately contribute to improved patient outcomes.

Keywords: *3D Convolutional Neural Network (3D CNN), Explainable AI (XAI), 3D Gradient Class Activation Maps (3D Grad-CAM), Convolutional Block Attention Module (CBAM), LUNA22 ISMI*

CHAPTER ONE

INTRODUCTION

1.1 Background of the Study

1.1.1 Medical Background

1.1.1.1 Cancer

Cancer occurs when some of the body's cells grow out of control and invade other bodily regions. With trillions of cells making up the human body, cancer can begin practically anywhere. Human cells typically divide to create new cells as needed by the body by growing and multiplying. New cells replace old ones when they die due to aging or injury. This controlled process can occasionally malfunction, causing damaged or aberrant cells to multiply and expand when they shouldn't. Tumors are lumps of tissue that can be formed by these cells. Cancerous/Malignant or healthy/benign tumors can both occur. Malignant tumors can metastasize, or spread into, neighboring tissues, and can also generate new tumors by spreading to far-off regions of the body (National Cancer Institute et al., 2021).

Cancer comes in more than a hundred varieties. The organs or tissues where tumors originate are typically used to name different types of cancer. For example, brain cancer begins in the brain, but lung cancer begins in the lung. Cancers can also be classified according to the kind of cell that gave rise to them, such as squamous or epithelial cells (National Cancer Institute et al., 2021).

1.1.1.2 Lung Cancer

The lungs are the primary site for lung cancer development. Lung cancer can be broadly classified into two primary categories; Small Cell Lung Cancer (SCLC) and Non-Small Cell Lung Cancer (NSCLC). In rare instances, a lung cancer may exhibit characteristics of both SCLC and NSCLC. This subtype is known as combined small cell/large cell carcinoma (American Cancer Society., 2007). Figure 1.1 shows the different parts of the lung.

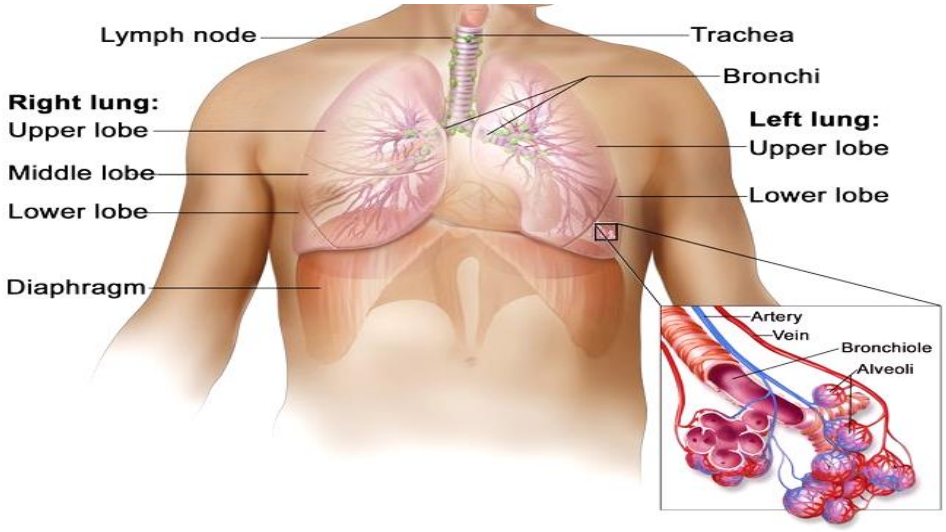


Figure 1.1 An illustration of the respiratory system's anatomy

1.1.1.3 Lung Cancer Screening

The prognosis for various cancers is significantly improved when the disease is detected and treated at an early stage. Lung cancer is one such cancer that can be identified early through screening methods. The term "screening" describes the process of using diagnostic procedures or tests to find a disease in people who have not yet shown any symptoms. Nonetheless, the majority of lung malignancies are discovered as a result of symptoms. It's important to remember that your doctor may not think you have cancer just because they advise a screening test. Examining a sample of lung cells in the laboratory allows for the precise diagnosis of lung cancer (National Cancer Institute et al., 2021).

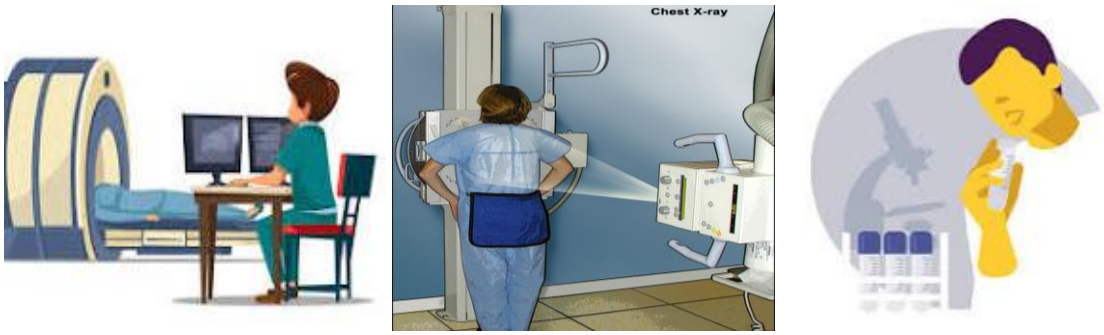
The most widely used screening tests for early identification of lung cancer are the three listed below.

Low-dose computed tomography (LDCT) also referred to as low-dose CT scan. It is a process that produces a series of exquisitely detailed pictures of different bodily sections utilizing an x-ray machine that emits very little radiation and connected to a computer. Three-dimensional views of tissues and organs are produced using the various camera angles. LDCT scan should be taken into consideration as a screening test for adults who have a high risk of lung cancer because of their age and previous tobacco use.

Chest X-ray: also known as a chest radiograph, is a common imaging test that uses a focused beam of radiation to create images of the organs and bones within the chest cavity. This includes the heart, lungs, blood vessels, airways, and the bones of the spine and ribs.

Chest X-rays are a relatively painless and quick procedure, making them a valuable tool for initial evaluation and monitoring of chest-related conditions.

Sputum cytology is a diagnostic procedure in which cells extracted from sputum—the mucus secreted from the lungs—are examined under a microscope. The purpose of this examination is to find any abnormal cells, especially ones that might be signs of lung cancer. Figure 1.2 shows examples of lung cancer screening methods.



LDCT-Scans Chest X-rays Sputum cytology

Figure 1.2 Examples of Lung cancer screening methods

There are also Positron Emission Tomography (PET Scans) and Magnetic Resonance Imaging (MRI Scans).

Frequent chest x-rays have been investigated as a lung cancer screening test for individuals who are at higher risk, but the results have not demonstrated that the majority of patients benefit from the test, and as a result, it is not advised for lung cancer screening. Studies using LDCT scans in individuals at greater risk of lung cancer have revealed that, in contrast to chest x-rays, yearly LDCT scans to screen those at higher risk of lung cancer can save lives. Getting annual lung cancer screenings before symptoms appear helps reduce the mortality risk from lung cancer for these individuals (Mayekar et al., 2022).

The primary objective of lung cancer screening and detection methods is to identify lung nodules. These are small, abnormal areas visible on CT scans. While most lung nodules are benign, stemming from factors like scar tissue or past infections, further investigation is often necessary to rule out malignancy. LDCT scans play a crucial role in this process. Not only can they detect lung nodules, but they also provide valuable information about enlarged lymph nodes (these may indicate the spread of cancer), tumor characteristics (LDCT scans reveal the size, shape, and location of any lung tumors) aiding in diagnosis and treatment planning.

By identifying lung nodules and offering detailed insights into potential tumors, LDCT scans significantly contribute to early lung cancer detection and improved patient outcomes.

1.1.2 Research Background

Artificial intelligence (AI) empowers computers to see and analyze their visual environment, known as computer vision (CV). This technology, specifically deep learning, leverages neural networks to "learn" from vast amounts of data. These networks mimic the human brain's decision-making process, enabling them to analyze images and identify features, making them valuable tools in lung cancer early detection. AI systems have diverse real-world applications, including: Speech Recognition: Converting spoken language to text. Anomaly Detection: Identifying unusual patterns in large datasets, potentially indicating equipment failure, human error, or security breaches. And Computer Vision: Enabling machines to understand and analyze visual data like images and videos.

While AI won't replace healthcare professionals, it's transforming the field with personalized care, outcome prediction, and improved diagnostics. Real-time data analysis can aid in disease treatment, cancer risk assessment, and even drug recommendations (Panesar, 2019).

As AI systems become increasingly sophisticated, understanding their decision-making processes is crucial, especially in fields like law, medicine, and defense. While early AI systems were transparent, complex models like Deep Neural Networks (DNNs) present challenges in understanding their reasoning.(Barredo Arrieta et al., 2020).

Deep Learning (DL) models, like DNNs, have demonstrated empirical success as a combination of their large parametric space and effective learning methods. Because of the latter space's millions of parameters and hundreds of layers, DNNs are regarded to be complex black-box models (Barredo Arrieta et al., 2020).

For proper adoption of ML and DL models specially in the healthcare sector, we need to answer at least this question: “Do we know the factors behind the models output?” Or “Do we know the conditions or cases that the model is successful and when it is not?” There is a new sub discipline of AI that deals with answering these types of questions and making black box models white, and it is called Explainability AI or XAI in short. Its scope spans in every field of AI. Figure 1.3 shows the scope of XAI.

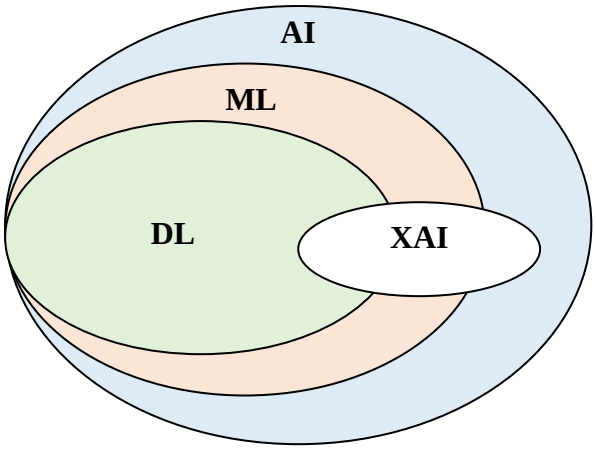


Figure 1.3 Scope of XAI

The aim of XAI, or interpretable machine learning (IML), is to create machine learning models that are comprehensible to people. It is a subject of study as well as the collection of current instruments and approaches created by that field. This covers techniques for interpreting black box models and modeling techniques applied to create easily interpreted models. “We can think of XAI as both interpreting models and making models more interpretable” (Conor O’Sullivan, n.d.).

There are different taxonomies of XAI and they will be discussed in chapter two in detail. Let see on the next section what we are trying to accomplish and the aim of this research.

1.2 Motivation of the Study

Lung cancer, which affects both men and women, is the primary cause of cancer-related fatalities worldwide. According to the latest data by the International Agency for Research on Cancer (IARC), approximately 19 million new cancer cases and 9 million cancer deaths occurred globally in 2022 (WHO, 2022). Cancer remains a leading cause of death worldwide, with lung cancer holding the top spot. Figure 1.4 (a) and (b) show the incidences for both sexes and (c) and (d) show the mortalities for both sexes (IARC, 2022).

Most countries predict lung cancer incidence and mortality to rise through 2050, making lung cancer a significant global public health concern. The continual shift in the epidemiology of lung cancer emphasizes the necessity of redistributing resources and enhancing lung cancer control strategies to lessen the burden of lung cancer globally for the future (Luo et al., 2023). Figure 1.5 shows forecasted number of new cases and deaths from 2022 to 2050, for males and females, age [0-85+] for Trachea, bronchus and lung cancers (Ferlay J et al., 2024) (IARC, 2024).

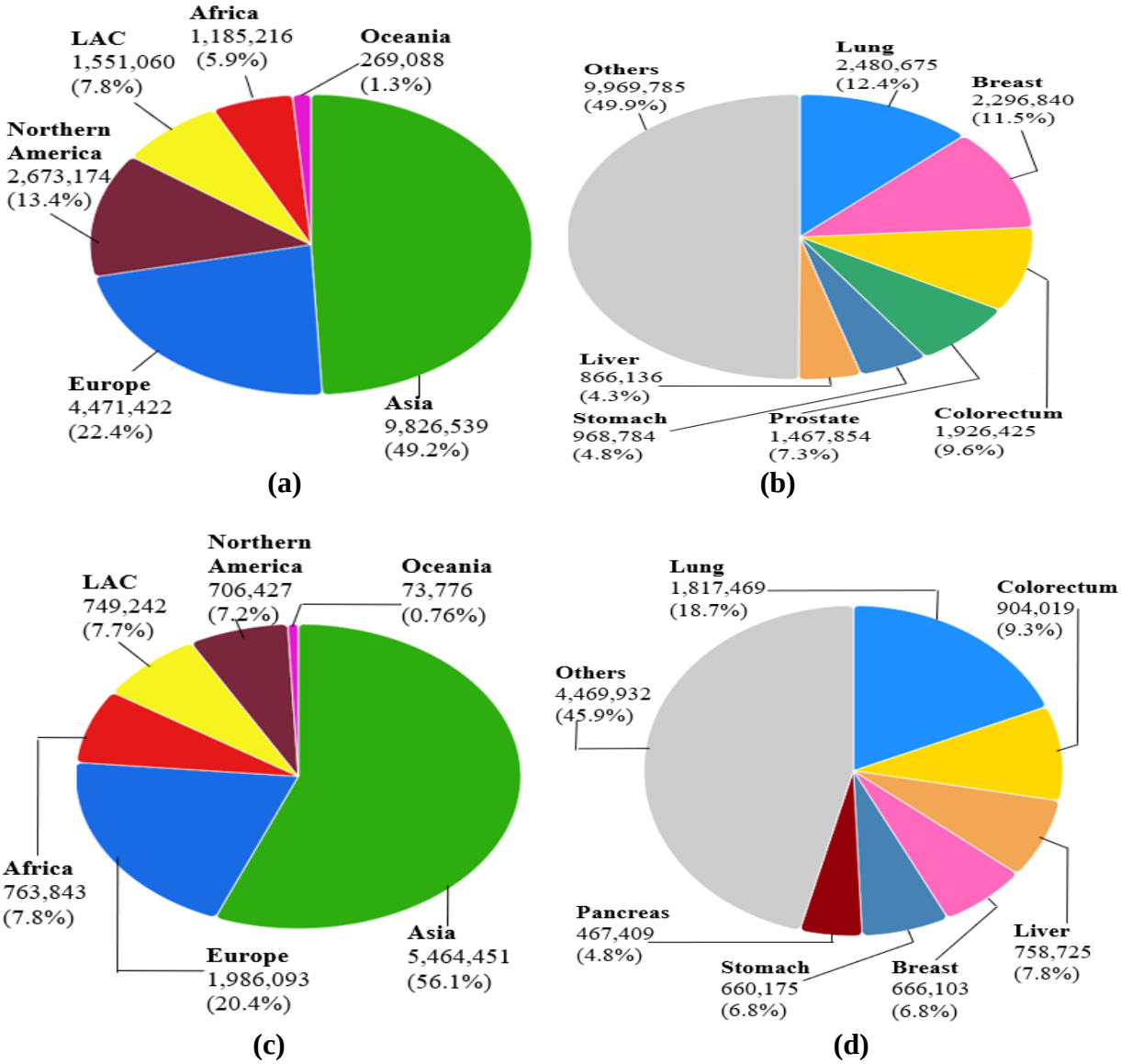


Figure 1.4 DataViz for Cancer Incidence and Mortality (IARC, 2022)

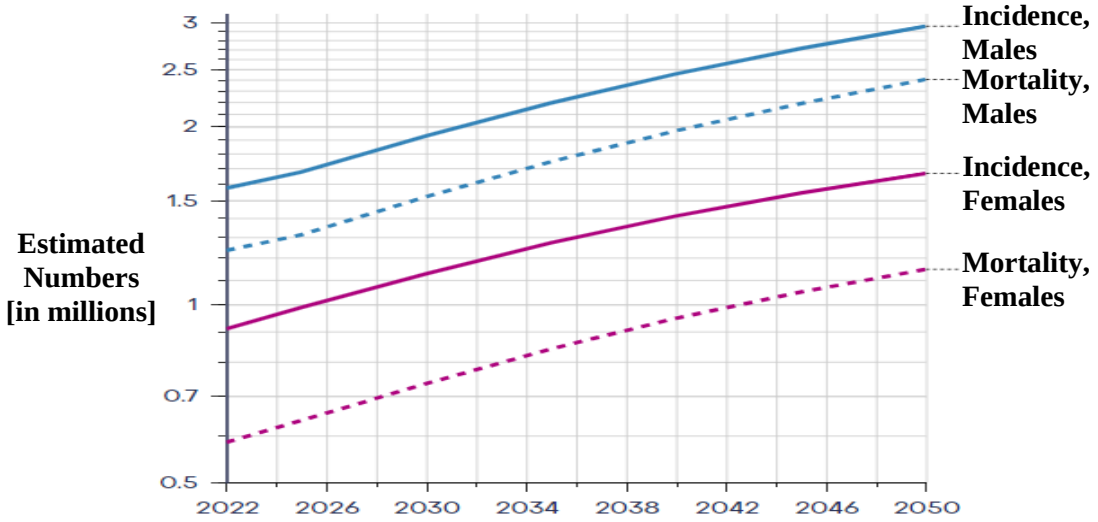


Figure 1.5 Projected number of new cases and mortality for trachea, bronchus and lung cancers

While treatment options exist, late-stage diagnoses are common. Early detection through screening programs for high-risk individuals is crucial for improving survival rates. LDCT scans significantly reduce lung cancer mortality in both current and past heavy smokers. Yearly screening for three years is superior to chest X-rays for early detection.

Although the LDCT scans are effective in detecting lung cancer at early stage, there are risks as well. One of them is identifying someone as having cancer when it is actually cancer negative. This could prompt unnecessary invasive operations. These risks happen because of the radiologist mistakes that are raised during the screening process. The workload for radiologists has grown dramatically over the previous thirty years (Markotić et al., 2021). Radiologists are feeling burnout, as there is too much critical data and too little time.

The data captured by medical imaging equipment has detailed information and has been used for AI models as AI is data driven programming. If used wisely with advanced techniques in the ML and DL, can assist the radiologists by minimizing the risks related to screening of lung cancer at early stage. As our models become more complex, they become opaque to the domain experts in our case the medical doctors. Using XAI techniques we can make our AI models more interpretable and transparent to the clinicians.

1.3 Statement of the Problem

The best and effective way of controlling cancer related deaths is with early detection and treatment. In recent years LDCT scans showed a more effective screening for lung cancer early detection. However, for properly screening individuals with high risk of lung cancer we need highly experienced radiologist and we need a lot of them as there will be too much data and to little physicians to examine them.

AI, particularly deep learning, shows promise in assisting radiologists with lung cancer screening. However, integration into clinical workflows requires careful consideration due to potential drawbacks.

There have been a lot of researches done on diagnosis of lung cancer and other types of cancers especially in detection and classification of the tumors using different DL techniques. However, there are numerous challenges facing the previous attempts. The most common is the interpretability of the models, which is important for medical professionals to comprehend the logic of the prediction. The interpretation of the results of the models will improve the transparency of the results produced by these models.

For instance, the work (Essaf et al., 2020) demonstrated high performance results for lung cancer detection using CT-scans, but the interpretation of these results and deep learning model is not available. Because these models don't reveal the reasoning behind their decisions, it may be challenging for people, including medical professionals, to comprehend why specific predictions are being made. This can be challenging in the healthcare industry, as patient safety and technology trust-building depend heavily on explainability.

Consequently, more study is required to determine how well DL models and prediction outcomes may be interpreted. This research can help both the AI developers and radiologists in understanding how these models work and make decisions and provide insights into how they can be improved. In conclusion, the interpretation of deep learning models for lung cancer diagnosis is essential for improving the reliability and trustworthiness of the results produced by these models.

1.4 Research Questions

The research questions guiding this work are:

RQ1: Which Deep learning techniques are performing best for early screening of lung cancer using CT scan images?

RQ2: Which XAI methods are used to interpret the results of the developed lung cancer detection model?

1.5 General and Specific Objectives

1.5.1 General Objective

This study is to build deep learning models for lung cancer diagnosis that incorporate interpretability of their results.

1.5.2 Specific Objectives

This study is to achieve the following specific objectives

- To prepare CT scan lung cancer medical image data for AI model building.
- To implement selected XAI technique to make deep learning models explainable and interpretable, for improving reliability and transparency.

- To optimize deep learning models that incorporate explainability and interpretability for supporting decision-making processes in lung cancer diagnosis.
- Develop 3D CNNs model using lung cancer CT scan raw medical image data and using XAI methods.

1.6 Significance of the Study

Lung cancer is a major public health concern, and early detection is vital for improving patient outcomes. Deep learning, particularly Convolutional Neural Networks (CNNs), has emerged as a powerful tool for analyzing medical images like CT scans and identifying lung nodules. Here's how the study can assist radiologists in early lung cancer detection:

3D CNNs can be trained to automatically scan CT scans and identify potential lung nodules, significantly assisting the radiologist in the screening process. This allows them to focus on analyzing suspicious nodules and making informed diagnoses. DL models can achieve high performance in detecting and classifying lung nodules, potentially exceeding human performance in certain cases. This can improve early detection of lung cancer and improve

Furthermore, these models can reduce the incidence of false positives, which occur when patients are mistakenly diagnosed with lung cancer. By minimizing unnecessary treatments which can help to optimize healthcare resource allocation, trained 3D CNN models can enhance the overall patient experience. Deep learning models are capable of analyzing vast amounts of data quickly and efficiently, enabling radiologists to review more scans in a shorter timeframe. This ultimately leads to improved patient throughput and reduced waiting times for other true cancer positive patients' diagnosis.

Addressing the "black box" issue; Our research focuses on developing explainable deep learning models for lung cancer diagnosis. By using explainable AI techniques, we aim to make these models more transparent and trustworthy, paving the way for their safe and effective integration into clinical workflows. Overall, integrating deep learning into lung cancer screening workflows can significantly assist radiologists in improving detection accuracy, leading to improved patient outcomes.

1.7 Scope and Limitations of the Study

The scope of this research is to build 3D CNNs models for lung cancer diagnosis and make these models transparent and understandable for the medical professionals for improving their usage in the medical industry. As we mentioned before there are various successful deep learning models built for lung cancer detection previously. However, the interpretability of the models, which is important for medical professionals to understand the reasoning behind the predictions is still a research gap.

The delimitation scope of XAI part are the set of boundaries and limitations that are placed on the concept of explainable AI. This includes the specific areas or domains in which explainable AI is relevant, as well as the characteristics and properties that define what is meant by "explainable AI". There are several key aspects that define the delimitation scope of explainable AI. Explainable AI is primarily relevant when decisions are made based on AI systems in domains, such as healthcare, finance, and criminal justice.

Explainable AI is focused on creating AI systems that are transparent and explainable to human users. This allows users to understand how decisions are being made by the system and to detect biases or errors. These aspects together define the scope of explainable AI, where decisions are made based on AI, while also emphasizing human interaction and alignment with human values and goals.

1.8 Organization of the Thesis

There are six chapters in the thesis. The second chapter is dedicated to understanding the fundamentals of medical imaging specifically of CT-scans, Convolutional Neural Networks of DL and related works done for lung cancer screening, finally XAI and related works done in the healthcare sector.

The third chapter discusses about the materials and methods used for the research work. Chapter four is devoted on describing the implementation of the proposed model.

Chapter five is all about the discussion of results found in every step of the implementation phase including the performance metrices used for evaluation of the DL models. Last but not least the last chapter is everything regarding conclusions and recommendations for future work.

CHAPTER TWO

LITERATURE REVIEW

2.1 Introduction

This chapter gives explicit reviews on literatures written on topics directly related to this research paper. As we have said before the main goal of this study is to develop an interpretable deep learning model that can detect and classify lung cancer from a lung CT-Scan data. In the following sections, we will delve into fundamental concepts related to medical imaging and the data associated with medical images particularly, CT scans. Subsequently, we will see topics related to CNNs, their applications in computer vision, and some related works done to lung cancer detection and classification. Finally, we will explore the field of XAI, its taxonomies, and some related XAI applications in healthcare.

2.2 Medical Imaging

Medical Imaging or Radiology describes the non-invasive methods and procedures used to provide illustrations of the human body's interior organs and tissues. These images, or visual representations, have the ability to identify and diagnose a wide range of illnesses, direct the development of treatment plans for those illnesses, and track the effectiveness of those plans. In particular, medical imaging is widely and frequently used to examine and visualize different parts of the body, including bones, muscles, organs, blood vessels, and other internal structures (Islam et al., 2023).

2.2.1 General Workflow of Radiologists

A radiologist workflow includes the following, first a doctor refers a patient for imaging, and the radiology department schedules the appointment. Depending on the scan, patients may need to fast or take contrast material. Then, the radiologist positions the patient, adjusts machine settings, and ensures comfort and safety. After that the technician captures images and adds relevant metadata. Finally, the radiologist analyzes images, compares them to prior studies, and writes a report with observations, conclusions, and recommendations (Giard, 2023). Figure 2.1 illustrates the summary of the above steps.

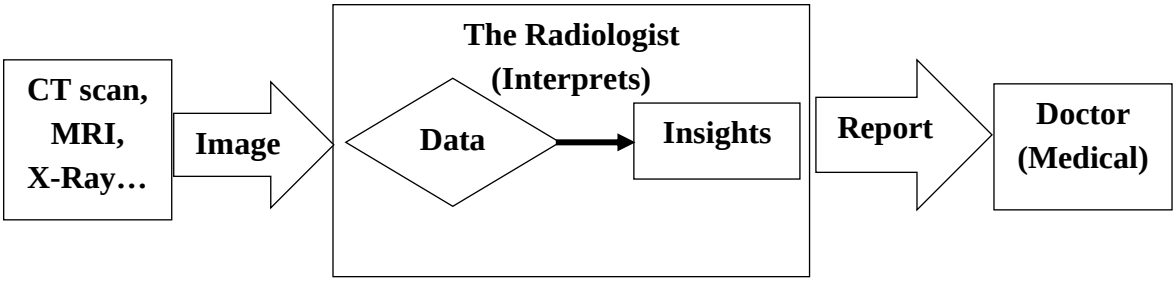


Figure 2.1 Basic work flow of Radiologists

2.2.2 CT-scan

Medical imaging modalities are techniques that use various tools to create visual representations of the human body for diagnosing and treating diseases (Islam et al., 2023). Popular medical imaging modalities include: Radiography (X-Ray imaging): include projection radiography or X-rays, produce 2D images. e.g. Chest X-ray imaging, Computed Tomography (CT) produce finely detailed cross-sectional pictures of the body using X-rays. produce 3D images. Magnetic Resonance Imaging (MRI): 3D images, Ultrasound Imaging: 2D/ 3D images, Positron Emission Tomography (PET): 3D images

The name CT-Scan also known as CAT-Scan is an abbreviation for Computed Axial Tomography. The term 'tomography ' derives from the Greek words 'tomos' (meaning 'slice') and 'graphein' (meaning 'to write'). With the use of detectors to measure the intensity of X-rays that pass through the body at various angles of incidence, CT scans use radiation to produce detailed images of internal structures. The details of CT Scan imaging are available at Appendix C.

2.2.3 Hounsfield Units

The physical density of a tissue or an organ is directly correlated with photon attenuation, or photon absorption. The image processor stores the data as bytes and transforms the values from the CT detectors, which quantify the degree to which scanned tissues attenuate photons (i.e., their density). This allows the pixels to be presented with pixel brightness that is appropriately assigned. Every value on the scale is referred to as a Hounsfield unit (HU), and the scale's range of values is named for Hounsfield (Shady Hermena & Michael Young, 2023).

Hounsfield units are used to describe the absorption rate of X-rays that was measured for each voxel when the CT- Scan was taken. It is a unit of radiodensity and is proportional to

the degree of X-ray attenuation and it is allocated to each pixel to show the image that represents density of the tissue or organ. (HU is named after Sir Godfrey Hounsfield)

CT Scan Density Measurement: **Attenuation Coefficients:** Different materials have assigned values called attenuation coefficients, reflecting their X-ray absorption. **Density Calculation:** Based on these coefficients, a linear transformation estimates the density of tissues and implants within and around the patient. **Reference Values:** Distilled water (0 HU) and air (-1000 HU) serve as standard reference points for density calculations.(Shady Hermena & Michael Young, 2023). Different publications define different ranges for certain tissues and substances, figure 2.2 shows HU scales of different substances or materials.

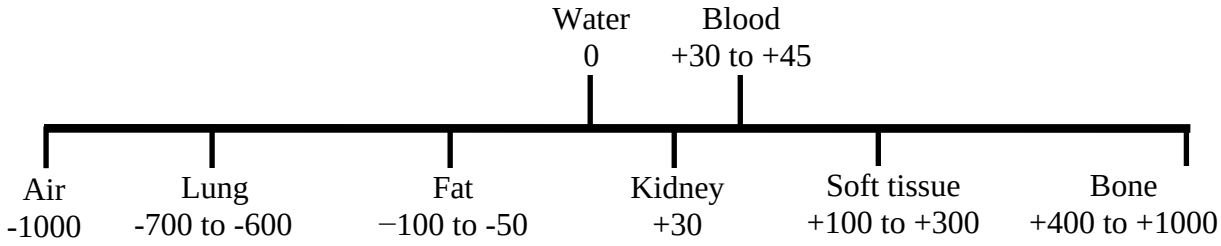


Figure 2.2 Hounsfield Unit Scale

In CT scans, the Hounsfield Unit (HU) values for various tissues are calculated using eq. (2.1)

$$HU = 1000 * \frac{\mu_{tissue} - \mu_{water}}{\mu_{water} - \mu_{air}} \qquad eq.\ (2.1)$$

Where: μ_{tissue} is the linear attenuation coefficient of the tissue or organ and μ_{water} and μ_{air} are the linear attenuation coefficients of water and air respectively (Shady Hermena & Michael Young, 2023).

Hounsfield unit is a pixel-by-pixel representation of the tissue density and is pixel-level measure of tissue density based on X-ray attenuation. Higher HU value indicates denser tissues (e.g., bone). lower HU value indicates less dense tissues (e.g., air)

Linear Attenuation Coefficient (μ) quantifies how X-rays interact with tissues. Higher μ means more X-ray absorption (denser tissue) and lower μ means less X-ray absorption (less dense tissue)

HU is derived from the linear attenuation coefficient of a tissue compared to water (reference point). in a CT scan that is proportionate to the degree of x-ray attenuation (Al-Zahrani, 2017). The conversion from pixel data to HU is a linear transformation:

$$HU = pixel\ value * slope + intercept \qquad eq.\ (2.2)$$

The formula in eq. (2.2) requires two key values stored within the CT Scan image files. Rescale slope (slope) scales the pixel intensity to physical units, and Rescale intercept (intercept) adjusts the baseline for the HU scale.

2.2.4 Data Formats in Medical Imaging

The most popular data or image formats used in medical imaging are DICOM and NIFTI. There are also other formats used in medical imaging and these formats including the above two are not used only for CT-Scans they are used for other modalities as well, such as MRI, X-Ray, Ultrasound, PET scans etc.

DICOM: “Digital Imaging and Communications in Medicine is the international standard for medical images and related information. It defines the formats for medical images that can be exchanged with the data and quality necessary for clinical use.” (NEMA PS3 / ISO 12052 et al., 2024). It is used for storing and sharing medical images in hospitals. DICOM files have file extension of “.dcm” or “.dicm”. It has two parts: Header: Contains patient information, study details, and image metadata. And Body: Stores the actual image data (2D, 3D, or 4D).

Primarily used in clinical settings, but raw scanner data often comes in DICOM format. Tools exist to handle DICOM files, like the “PYDICOM” library, their complexity often leads to conversion to other formats for research and machine learning (Li et al., 2016). They have Key-value pairs for easy access to data using unique tags. Example shown on figure 2.3

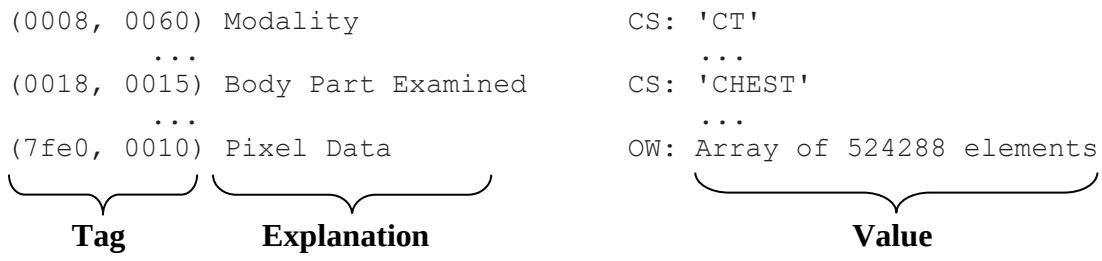


Figure 2.3 Sample DICOM file key value pairs to access the data

NIfTI: Neuroimaging Informatics Technology Initiative Open file format for medical images, often used in neuroimaging research. Simpler and more interoperable than DICOM. Smaller file size, easier for storage and sharing. It has two parts: Header contains essential information about image geometry, not patient or scanner details. Body stores the actual

image data (Heindl, 2022; Li et al., 2016). Similar to the DICOM files the data can be accessed using libraries with header index.

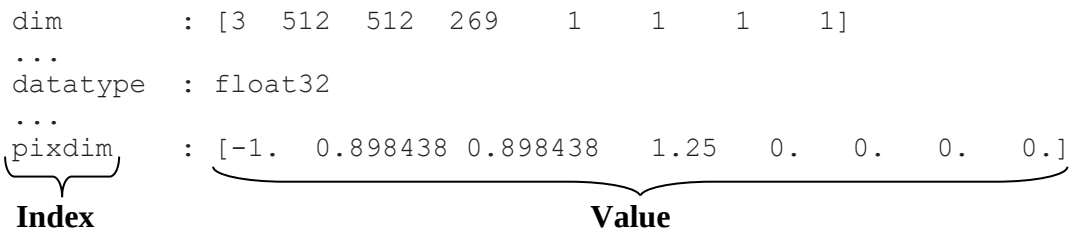


Figure 2.4 Sample meta data of NIFTI formats

NIfTI and DICOM differ from each other in the following ways: while DICOM works with 2D layers, NIfTI can display 3D detail (with NIfTI files, images, and other data saved in a 3D format). NIfTI can take longer to load, while DICOM allows users to display one layer at a time. Less metadata is included in NIfTI files. More information can be contained in DICOM files.

2.3 Deep Learning

DL is a sub field of ML which uses models that employed Artificial Neural Networks (ANN). DL algorithms have extensive applications in healthcare especially in informed clinical decision support systems. Since, compared to DL, classical ML has not been effective in working on problems like Natural Language Processing (NLP), image detection, image recognition, and others. The proper choice for this research study is DL and its different architectures and tools, especially for feature extraction.

Powerful computational models learn complex representations from data through multiple processing layers. Achieved state-of-the-art results in various domains like speech recognition, image recognition, and object detection. They use backpropagation to adjust internal parameters and discover intricate patterns in large datasets. There are different types of deep learning models: Convolutional Neural Networks (CNNs) which exceled in image, video, and audio processing, Recurrent Neural Networks (RNNs) that handle sequential data like text and speech, and there are also other types (Lecun et al., 2015).

The fundamental components of deep learning systems are Neural Networks (NN) or Artificial Neural Networks (ANN). Since "network" refers to a structure like a graph and "neural" is the adjective form of "neuron," an "Artificial Neural Network" is a computing

system that aims to imitate or at the very least, be inspired by the neural connections found in human nervous system (Dr. Rosebrock, 2017).

A system needs to have a directed, labeled graph structure with simple computations performed by each node in the graph in order to be classified as a NN. Each node or a single neuron in the NN is called a "perceptron/ artificial neuron" (Dr. Rosebrock, 2017).

Every node carries out a basic calculation. The signal, or the computation's output, is then sent from one node to another via each link, marked with a weight that denotes how much the signal is amplified or attenuated. Certain connections have significant positive weights that enhance the signal, suggesting the signal's criticality in the categorization process. Others have negative weights, which diminish the signal and indicate that the node's output is not as significant in the final categorization. If such a system has a graph structure with adjustable connection weights through the use of a learning algorithm, we refer to it as an ANN (Dr. Rosebrock, 2017). Figure 2.5 shows a single artificial neuron or perceptron.

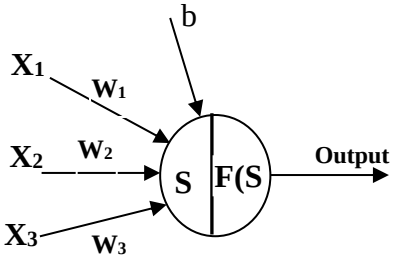


Figure 2.5 Perceptron (Artificial Single Neuron)

It receives all the inputs (X_1, X_2, X_3) and the weights (W_1, W_2, W_3) and the summation function sums all the inputs multiplied with the weights then add the bias:

$$S = [\sum_{i=1}^{n} w_i * x_i] + b \qquad \textit{eq. (2.3)}$$

F(S) is an activation (transformation) function; the output of the summation function will be the input to the activation function. After we apply activation function on it, it will give us Output.

$$Output = F(S) \qquad \textit{eq. (2.4)}$$

There are different types of activation functions some of them are Step function, Signum function, Linear function, ReLu function, Leaky-ReLu function, Sigmoid function, Softmax function and Hyperbolic Tangent function. Figure 2.6 shows a multi-layered feed forward NN.

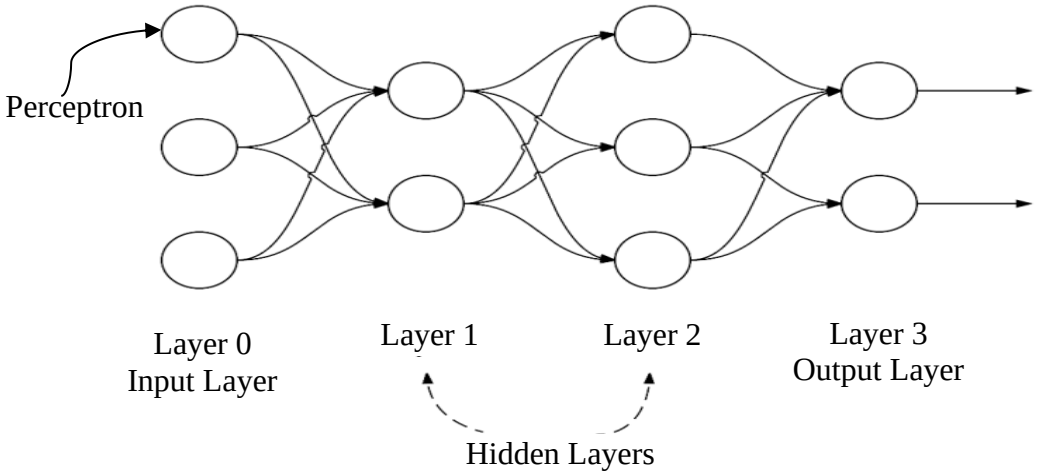


Figure 2.6 A multi-layer feed forward ANN

Most commonly there are three types of DL models supervised, unsupervised and semi supervised; supervised DL models use labeled data or examples (i.e. where the target variable or the desired output is known). Convolutional Neural Networks (CNNs) and Recurrent Neural Networks (RNNs) are widely used supervised DL models. Unsupervised DL models extract structure from data. Autoencoders are best examples of unsupervised NN. Semi-supervised models are used when we have labels for part of our data but not for the rest (Dr. Rosebrock, 2017). CNNs perform higher with image, speech, or audio signal inputs than other types of neural networks. CNNs are mostly used and performed well in the applications of image classification, object detection, feature extraction and image segmentation etc.

There are three main kinds of layers in CNN. The first one is Convolutional Layer, it is a core component Responsible for most computations in CNNs. Key elements include; Input data which is the image to be analyzed. Filters are small matrices that detect specific features in the image. Feature maps that are outputs of filters, indicating where features are found.

The Convolution process is as follows the filter slides across the image, computing dot products with input regions. Then, output feature map shows locations where the filter detects the desired feature. Hyperparameters included are; number of filters which controls the number of feature maps produced. Stride which controls how much the filter shifts, affecting output size. Padding a parameter that adds zeros to the image border if needed to maintain output size. After each convolution, an activation function is applied to introduce non-linearity into the CNN model. This helps the model learn more complex relationships in the data. Figure 2.7 shows an example how the convolution layer operates and the formula to calculate the size of feature maps.

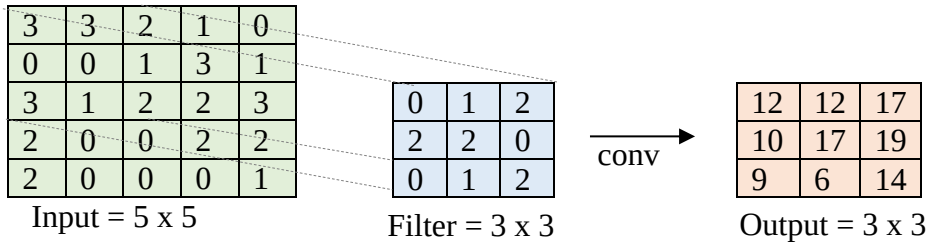


Figure 2.7 Illustration of how the convolution is done

$$output\ size = \frac{input\ size - filter\ size + (2 * padding)}{strid} + 1 \qquad eq.\ (2.5)$$

Convolution layers may be followed by another convolution layer. When this occurs, the CNN's structure may become hierarchical because the pixels in the receptive fields of earlier layers are visible to the layers that come after them.

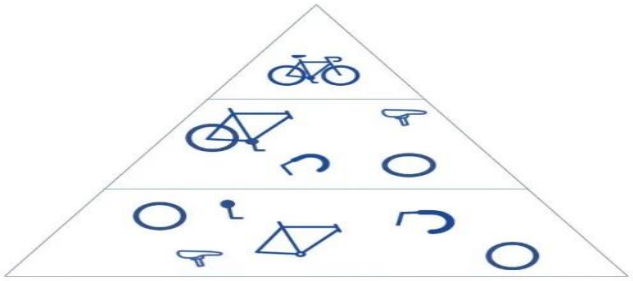


Figure 2.8 hierarchical CNN structure

(Source: http://viraai.com/wp-content/uploads/2021/10/5.jpg)

The second type of layer is Pooling Layer which downsizes feature maps to reduce the number of parameters and spatial dimensions (width and height) for computational efficiency. There are two types; Max Pooling which selects the maximum value within a local region. And Average Pooling which selects the average value within a local region. Makes the model more robust to small image variations (translation, distortion) and also Reduces overfitting.

The third type of layer is Fully Connected Layer (FC- layer); Analyzes features from previous layers and classify the data. Each output node connects to all other nodes. Typically use SoftMax for multiclass classification, Sigmoid for binary classification, and it has probability outputs (0-1).

CNNs have witnessed significant architectural advancements over time. Pioneering models like LeNet, introduced in 1989, laid the groundwork for deeper and more complex architectures. LeNet, designed for handwritten digit recognition, utilized a relatively shallow architecture with seven layers and operated on greyscale images of size 32x32 pixels.

Following LeNet, the field saw a surge in the development of increasingly powerful CNN architectures. The 2012 ImageNet competition winner, AlexNet, marked a significant leap by demonstrating the effectiveness of deeper architectures with eight layers. This success spurred further exploration of deeper models, leading to the development of various prominent architectures like: VGG, The VGG family of architectures, introduced in 2014, explored the concept of increasing depth by stacking numerous convolutional layers with small filter sizes. Other architectures include: ResNet (2015), Inception (2014), DenseNet (2016), EfficientNet (2019), etc.

The training of CNNs is done through back propagation algorithm. Back propagation is the process of calculating weight and biases that best fit the model. It is the method of updating the weights and biases of the network to minimize the error which is the difference between the predicted and actual value.

2.4 Related Works

Our initial information retrieval targeted relevant databases encompassing both general scientific publishing platforms and domain-specific medical imaging resources. General databases included national repositories like the National Academic Digital Repository of Ethiopia (NADRE) alongside internationally recognized platforms such as PubMed, ScienceDirect, arXiv, and academic publisher databases (Frontiers, Hindawi, Taylor & Francis, etc.). Additionally, we consulted domain-specific resources like ResearchGate, ACM Digital Library (for computer science), and IEEE Xplore for potentially relevant publications.

Then we developed a search string using keywords related to our research topic. Examples include "interpretable deep learning," "lung cancer detection," "lung cancer classification," "CT scans," "explainable AI (XAI)," etc. Boolean operators (AND, OR, NOT) were used to refine our search and focus on relevant articles.

Inclusion/Exclusion Criteria: we established a clear criterion for selecting relevant articles for our review. this involved: publication date (focusing on recent advancements, not more

than five years), peer-reviewed journals or reputable conference proceedings, studies directly addressing issues related to our study and studies only written in English.

Selection Process: we used a two-stage article screening; title and abstract screening and full-text evaluation. In the first stage, titles and abstracts of the retrieved articles were evaluated. Articles deemed clearly irrelevant to the research topic based on their titles and abstracts were excluded. This initial screening helped to efficiently reduce the number of articles for further consideration. Articles passing the initial title and abstract screening stage proceeded to a more in-depth evaluation. Full-text versions of these articles were obtained and thoroughly examined to assess their alignment with the research question, methodology, and inclusion/exclusion criteria.

After the above steps we critically evaluated the selected articles to assess their quality, relevance, and methodological rigor. Consider aspects like: Study design (retrospective, prospective). Dataset characteristics (size, source, type of CT scans). Deep learning model architecture and if there are interpretability techniques employed. Evaluation metrics used (accuracy, interpretability measures). Strengths and limitations of the research presented. Finally, we have used reference management software Mendeley to organize and manage our retrieved articles and references.

As we have discussed previously, DL algorithms have extensive applications in healthcare especially in informed clinical decision making which is when patients are able to weigh the pros, disadvantages, limitations, alternatives, and uncertainties of the clinical care they are considering for a certain disease or condition. There are a lot of successful applications of DL in detecting and classifying lung cancer in terms of performance but most of them are done on 2D images or by converting the CT-Scans to ‘jpeg’ and ‘png’ formats. Some of them include (Sori et al., 2019), (B. C. & K. B., 2023), (Shafi et al., 2022), (Tandon et al., 2022), (Bushara & Kumar, 2022), (Ramana et al., 2022) and many more. Since this research study used 3D CNNs for the raw 3D medical image data we will focus on studies that employed 3D deep learning models, let’s see some of them next.

A research work on (Jiang et al., 2022) demonstrated the effectiveness of a deep learning model for accurate and fast lung cancer prediction using CT scans. The model comprised three stages of work. The first one is lung nodule detection, which is identifying potential cancerous nodules. The second is Benign vs. Malignant classification that is differentiating between Benign and Malignant nodules. The third is false positive reduction, which is

filtering out non-cancerous nodules. The researchers employed various network architectures and loss functions for optimal performance. Notably, they introduced "Nodule-Net," a detection network combining U-Net and RPN (Region Proposal Network) for improved accuracy in lung nodule detection.

(Causey et al., 2022) proposed "DeepScreener," a deep learning model for predicting lung cancer using volumetric CT scans. It combines CNNs for whole-image analysis and leverages Spatial Pyramid Pooling and 3D convolutions to capture spatial information at various scales, enhancing feature extraction. Trained on carefully curated datasets, DeepScreener achieved promising results on an independent test set, demonstrating its potential as a valuable tool for lung cancer detection and risk prediction.

A study in (Essaf et al., 2020) investigated the categorization and identification of CT lung images using convolutional neural networks. It used four different consecutive stages of work steps: (1) Using the database's description file CT images of malignant pulmonary nodules and normal nodules was located; (2) Using the maximum inter-class variance method CT image was Segmented to create a binary image; (3) A suitable CNN was designed; (4) CCN was trained and CT image of malignant lung nodules were classified. This study employed the LIDC dataset and the study's findings indicated that a CNN received the processed CT scans as input for both training and recognition. 95.85% was the average classification accuracy rate following six-fold cross-validation.

(Sun et al., 2020) utilized a two-stage approach to classify lung nodules using Generative Adversarial Networks (GAN) and 3D CNN. In the first stage, the model employed transfer learning based on Deep Convolutional Generative Adversarial Networks (DCGAN) to preliminarily classify pulmonary nodules. The GAN model generated a dataset similar to the original data, which was then used to pre-train the nodule detection network based on AlexNet. The pre-trained network is fine-tuned using the original data to obtain a neural network with high accuracy. In the second stage, a 3D CNN was introduced to remove false positives in the classification process. They stated that this two-stage approach leveraged the strengths of GAN for data generation and transfer learning in the first stage, followed by the use of 3D CNN for accurate classification and false positives removal in the second stage. The accuracy of the test set in the first stage is reported to be 94.50% and the highest accuracy rate of false positives removal is reported to be 95.30%.

Lei Bao and his colleagues proposed a simple residual network for lung nodule classification. First, they suggested employing the Self-Attention technique to identify global features. Subsequently, a module called Inception-like was presented in order to extract multi-scale local features The identified local features were regarded as complementary part of global features with residual connection. By combining the self-attention mechanism for global feature detection and the inception-like module for multi-scale local feature extraction, the network was able to effectively capture both global and local features essential for accurate lung nodule classification. The network achieved an AUC of 96.07%, indicating its high discriminatory power in distinguishing between benign and malignant lung nodules. The accuracy of the model was reported as 90.45%, and the precision and recall values were 0.8993 and 0.9015, respectively (Bao et al., 2020).

Zhifeng Lin and his partners proposed a 3D CNNs with shortcut connections for improved lung nodules classification. As 3D CNN has many parameters, which leads to low model efficiency and to mine the vertical information of tumor CT images and increase the model's training efficiency they proposed a VGG based 3D residual connection network, called 'VGG+ResCon' or 'VGG plus ResCon'. The VGG+ResCon network utilizes shortcut connections to improve the classification of lung nodules in CT images by addressing the issue of gradient disappearance, simplifying the learning process, and accelerating the convergence rate of the network. Additionally, the use of shortcut connections in the VGG+ResCon network enables the model to mine vertical information from tumor CT images more effectively. This helps in capturing important spatial features of pulmonary nodules in 3D CT data, leading to improved accuracy in distinguishing between benign and malignant nodules. The authors stated that their methodology was evaluated on the LUNA16 dataset, with the best precision of 93.62%, recall of 92.48%, specificity of 96.83% and f1-score of 93.04% (Lin et al., 2020).

There are many other deep learning applications in detection and classification of lung cancer; we just selected some of them. Table 2.1 summarizes the aforementioned literatures. We included research gaps on each literature even though we focused on the interpretability aspect of these models, it's important to note that there are various other research gaps, as highlighted in the table.

Table 2.1 Summary of Related Works of DL on Lung Cancer

Ref.	Methodology used	Dataset used	Results (%)	Gap found
(Jiang et al., 2022)	Lung region segmentation**;** candidate region feature extraction**;** lung nodule recognition and classification. Nodule net for lung nodule detection (3D U-Net).	LIDC IDRI, and LUNA16	FROC = 0.876 (Free-Response Receiver Operating Characteristic Curve)	▪ Lacks detail on the real-time clinical implementation of the proposed models, although the paper mentions fast detection speeds including integration and impact on radiologists. ▪ No interpretation of the results and this reduce their trustworthiness in clinical settings ▪ The research primarily addresses detection and classification at a single time point, lacking insights into the progression of lung nodules over time.
(Causey et al., 2022)	Deep screener base on spatial pyramid pooling with 3D convolution	LIDC, DSB2017, LUNA16, and NLST	Accuracy = 78.2 AUC = 85.8 AUPRC = 78.8	▪ Further refinement of the DeepScreener model to improve its accuracy and reduce false negatives (fine-tuning the architecture and hyperparameters of the model to optimize performance). ▪ High False Positive Rates ▪ No interpretation of the results and this reduces their trustworthiness in clinical settings
(Essaf et al., 2020)	The CT scans were segmented using Otsu method, and the features are extracted using CNN then classified the CT scan images as benign and malignant.	LIDC IDRI	Accuracy = 95.85 AUC = 82.16	▪ Insufficient evaluation metrics ▪ No interpretation of the results and this reduces their trustworthiness in clinical settings

Ref.	Methodology used	Dataset used	Results (%)	Gap found
(Sun et al., 2020)	Generative Adversarial Networks (GAN) and 3D CNN	LIDC	Accuracy= 94.50	▪ The data augmentation needs improvement. This could involve exploring different architectures or training strategies for GANs to better capture the variability of lung nodules. ▪ No interpretation of the results to understand the decision-making process of the models, thereby increasing trust and usability. ▪ Insufficient evaluation metrics
(Bao et al., 2020)	Simple residual network for lung nodule classification. Self-Attention technique to identify global features	LIDC	AUC = 96.07 Accuracy = 90.4 Precision = 89.93 Recall = 90.15	▪ The feature extraction modules need improvement. Building better local and global feature extraction modules could strengthen the model's performance (a need for more advanced architectures/techniques that can enhance the extraction of relevant features from lung nodule images). ▪ No interpretation of the results and this reduces their trustworthiness in clinical settings.
(Lin et al., 2020)	VGG based 3D residual connection network, called VGG+ResCon	LUNA16	Accuracy = 93.6 Precision = 93.6 Recall = 92.48 Specificity = 96.83 F1-score = 93.04	▪ No interpretation of the results and methods to visualize and explain the decision-making process of the network. This could help clinicians trust and understand the model's outputs better.

2.5 Explainable AI (XAI)

2.5.1 Introduction to XAI

The field of XAI (Explainable Artificial Intelligence) aims to make machine learning models more understandable and interpretable. This is particularly important in sensitive areas like healthcare and finance, where users need to trust and understand the decisions made by these models. The terminologies used and goals of XAI are as follows as given by (Ali et al., 2023; Barredo Arrieta et al., 2020; Conor O'Sullivan, n.d.)

Terminologies include: Explainability; the active process of making a model's inner workings clear and easy to understand. Interpretability; model's inherent characteristic of being understandable to humans. Understandability; ability of a model to be understood without detailed explanations. Comprehensibility; The ability of a model to express its knowledge in a way humans can understand. Transparency; The ability of a model to be understood on its own.

Goals of XAI include: Trustworthiness; building trust with domain experts and users affected by the model's decisions. Causality; understanding the causal relationships between inputs and outputs. Transferability; ensuring the model's performance generalizes to new data. Informativeness; providing evidence to support the model's predictions. Confidence; ensuring the model's predictions are reliable and stable. Fairness; Mitigating biases and ensuring fair outcomes for all. Accessibility; Making the model's explanations accessible to different stakeholders. Interactivity; Allowing users to interact with the model and explore its explanations. Privacy; Protecting user privacy while providing explanations.

XAI research seeks to achieve these goals through various techniques, ultimately aiming to improve the trust, reliability, and responsible use of machine learning models.

Black Box vs. White Box vs. Gray Box Models: (Kenton, 2023)

- Black Box: Opaque inner workings. Difficult to understand and trust its decisions.
- White Box: Fully interpretable, but often less accurate.
- Gray Box: Offers a balance between interpretability and accuracy, allowing partial understanding of the model's reasoning.

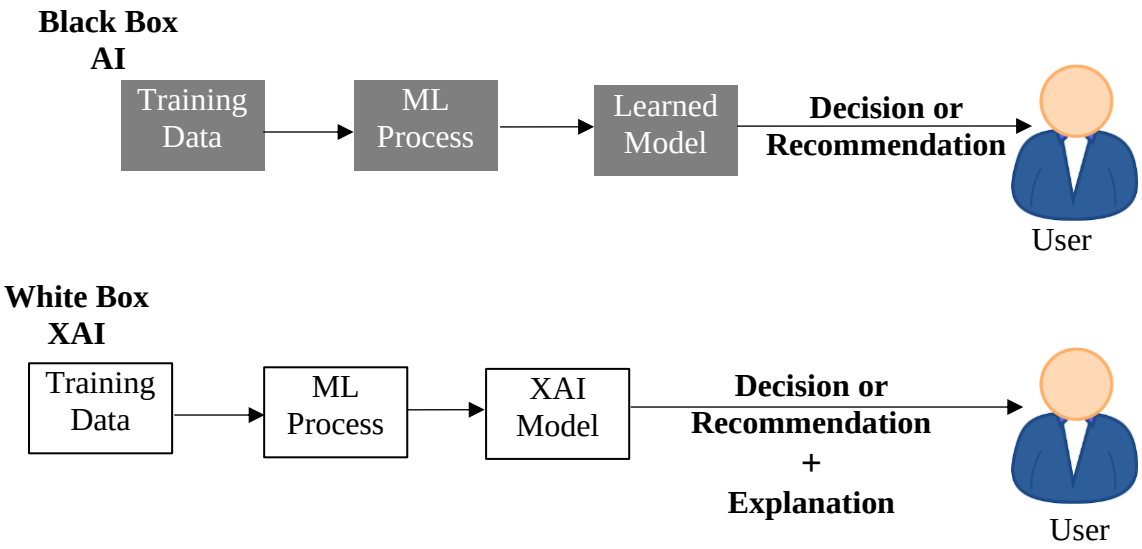


Figure 2.9 Black Box AI vs. White Box XAI

XAI research suggests that it might be possible to overcome the traditional trade-off between model complexity (and accuracy) and interpretability. This means we could potentially have models that are both accurate and understandable (Barredo Arrieta et al., 2020) .

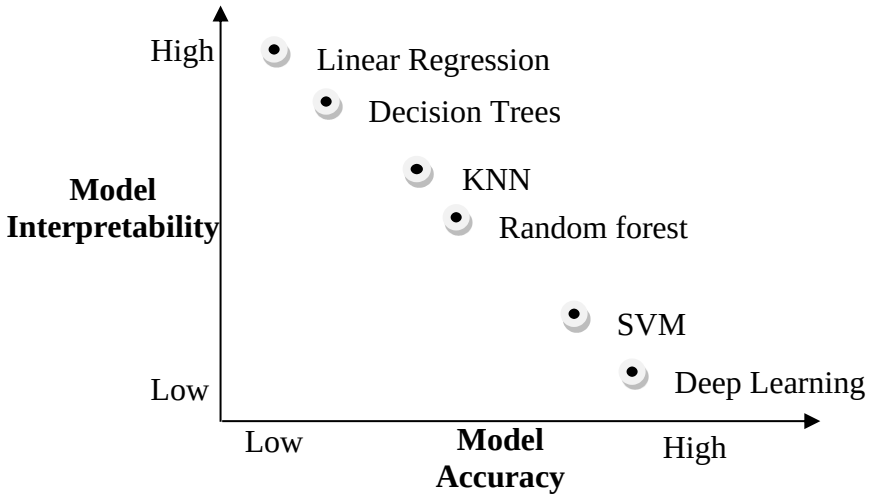


Figure 2.10 Interpretability vs Accuracy

Similarly, (Conor O'Sullivan, n.d.) has put the interpretability spectrum as shown in figure 2.11 below. As models become more complex, they become less interpretable. A ML model is a function. The input consists of the model's features, while the output is its predictions. A function that is too complex for a human to comprehend is a black-box model. To be able to peek inside the black box and comprehend how the model functions, we require an extra approach.

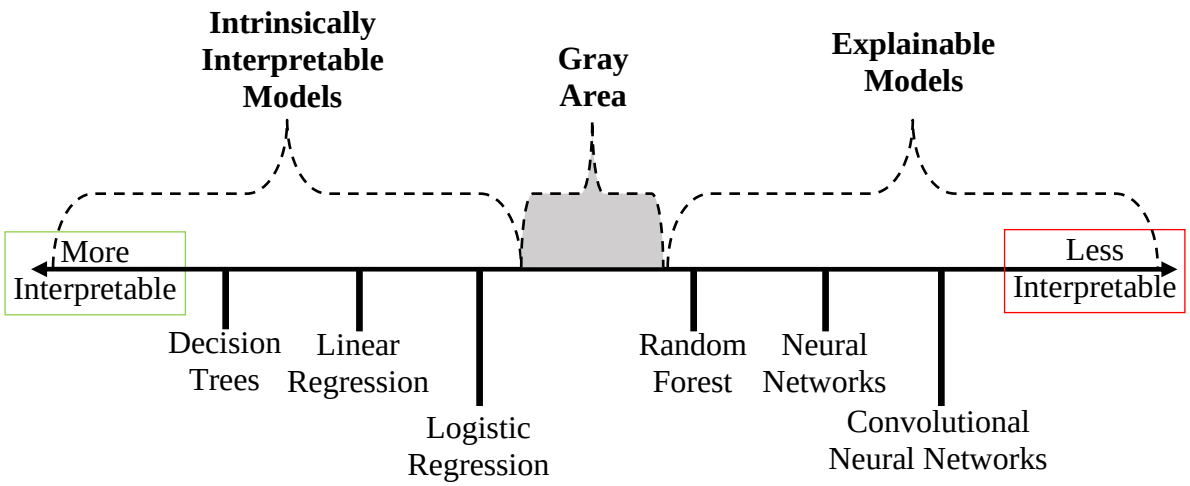


Figure 2.11 The Interpretability Spectrum (Interpretability vs Explainability)

Intrinsic interpretability focuses on building simple models (like decision trees or regression) that are inherently understandable due to their structure. However, these models may struggle to capture complex non-linear relationships in the data (Stiglic et al., 2020).

2.5.2 Taxonomy of Explainability Methods

Taxonomies are a helpful tool for structuring discussions, but there are just too many different explainability techniques for there to be a single, practically applicable taxonomy due to their scope and complexity (Speith, 2022). There is various classification criteria used to categorize techniques for ML interpretability. Some of them are as follows; as described by (Molnar, 2022).

Local vs Global Explanations: This categorization criteria asks the question Does the XAI method explain a particular sample or the whole model? Does the model's overall behavior or a specific prediction get explained by the interpretation method? Local methods provide explanations for individual samples. Global methods provide explanations for the entire model or group of samples. Local methods result per sample can be averaged across samples. Global methods will weight input parameters the same way regardless the individual prediction (Molnar, 2022).

In global interpretation, the overall view of the model, along with data predictions and explanations, the data exploration, which displays an overview of the data set along with the prediction’s values, the global importance, these aggregates, features, importance values of individual data points, to show the models’ overall tap key and the explanation demonstrates how a feature affects the change in the model prediction values(Molnar, 2022).

In local interpretability, the local importance highlights the importance features for any individual prediction. It illustrates the local behavior, of the underlying model at a specific data point. The data collection exploration, is a sort of what id analysis. These observations allow changes to feature values of the selected data points, and observers outing changes to the prediction value. Local explanations can illustrate how a prediction would change when a feature change(Molnar, 2022).

Model Agnostic vs Model Specific Explanations: This classification asks the question does the interpretation depend on a particular model? Model specific interpretation tools are limited to specific models while, model agnostic tools can be used on any ML model. Agnostic methods usually work by analyzing feature input and output pairs. These methods cannot have access to model architecture such as layer weights or structural information (Molnar, 2022). Figure 2.12 shows some explainability methods in their class.

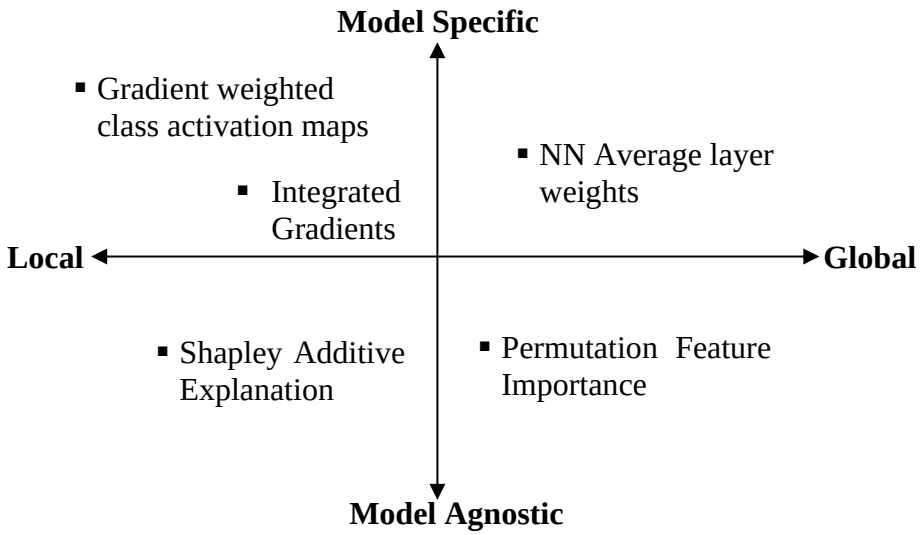


Figure 2.12 Examples of XAI methods in their class

Ad-hoc vs Post-hoc Explanations ask the question when does the explanation occur? In ad-hoc explanations the model has been designed to be intrinsically explainable (Molnar, 2022).

Data modality specific vs Data modality agnostic are based on the data type; it can be tabular data, time-series data, or imaging data. E.g. Grad CAM is only applicable to data related to CNNs such as images. Data modality specific refers to explanation methods that are only applicable to a certain type of data. Data modality agnostic methods are commonly coupled to explanations that are model-neutral or model agnostic, they refer to explanations that work for any data type (Molnar, 2022).

Surrogate Models vs Attribution/ visualization methods: Surrogate models use an interpretable model or a simpler one that simulates a black box's behavior. Attribution/ visualization methods attempt to visualize certain aspects of the model to allow explanation on why and how the model reach a decision (Molnar, 2022).

On the other hand, a work in (Speith, 2022) reviewed recent approaches and classified them into four approaches: The Functioning-Based Approach which is the way the explainability methods extract information from the model, The Result-Based Approach, The Conceptual Approach which is to partition the field into several conceptual dimensions, and The Mixed approach which is a hybrid of the above 3 approaches. The paper's thorough taxonomy approaches are comprehensive and beginner friendly the illustration below showed the mixed approach which is likely good for new comers in the field. It has Top level: scop distinction. Middle level: applicability distinction. And Bottom level: elements of other approaches.

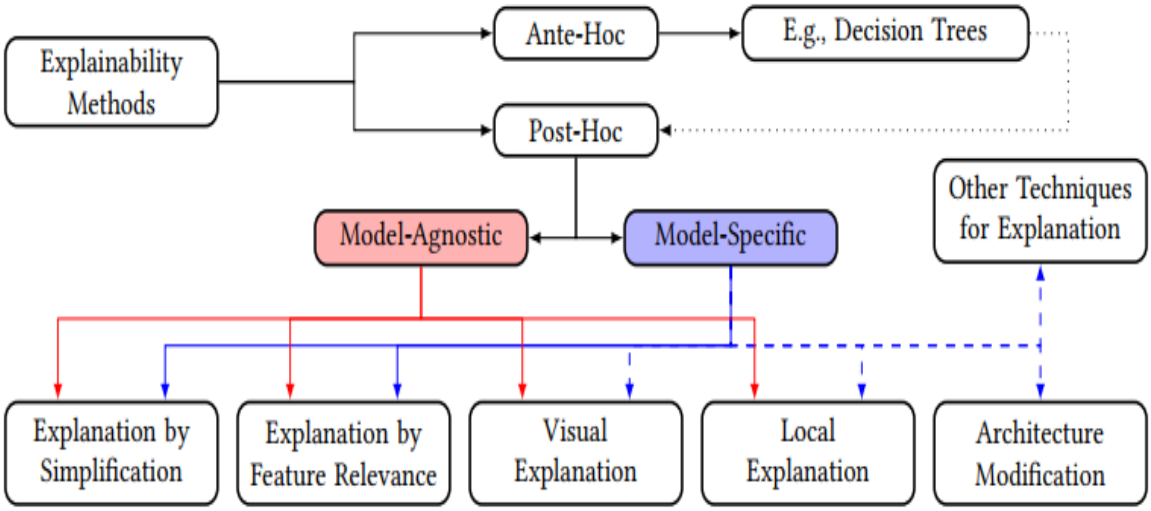


Figure 2.13 General taxonomy of the hybrid approach

2.5.3 Overview of Popular XAI Methods

LIME (Local Interpretable Model-Agnostic Explanations): Creates simplified models around specific data points to understand their influence on predictions (Ribeiro et al., 2016).

SHAP (SHapley Additive exPlanations): Uses game theory concepts to attribute credit for predictions to individual features, providing detailed insights into their impact (Lundberg & Lee, 2017).

PFI (Permutation Feature Importance): Assesses feature importance by shuffling feature values and observing the model's performance change (Mi et al., 2021).

LRP (Layer-wise Relevance Propagation): Backpropagates the prediction score through the model, assigning relevance scores to features and activations in each layer, providing a more granular understanding of the decision-making process (Bach et al., 2015; Sarker, 2021).

Grad-CAM, which stands for Gradient-weighted Class Activation Mapping, is a technique used in deep learning, especially with convolutional neural networks (CNNs). It helps us understand what parts of an image are most important to the CNN's prediction for a particular class. According to (Selvaraju et al., 2017) it works with an input image and model (pre-trained CNN model). The model analyzes the image and calculates the gradients of the final classification score concerning the activations of the last convolutional layer. Gradients indicate how much each activation in the layer influences the final score (Grad Calculation).

Then, gradients are used to weigh the activations in the convolutional layer, creating a class activation map. This map highlights the image regions that have the most influence on the classification (Class Activation Map). The class activation map is overlaid on the original image, typically as a heatmap. Brighter areas represent image regions the model focused on for its prediction (Visualization). Grad-CAM offers a way to interpret the "black box" nature of deep learning models, particularly CNNs used in image recognition. By visualizing the important regions, we gain insights into why the model made a specific prediction.

Grad-CAM is versatile and can be applied to various CNN architectures without modifying the model itself. Grad-CAM is particularly useful for diagnosing model predictions and improving trust in their decisions. There are advanced libraries available, like Pytorch-grad-cam, that provide functionalities for implementing Grad-CAM (Selvaraju et al., 2017).

There are many other XAI techniques as well, XAI is a rapidly evolving field with a variety of techniques to understand and explain AI models. By choosing the right XAI method for your specific needs, you can unlock the full potential of AI while ensuring transparency and responsible use

2.5.4 Explainability in Healthcare Applications

Explainable Artificial Intelligence (XAI) has emerged as a critical area of research in healthcare applications that leverage deep learning models. While deep learning models can achieve impressive accuracy in tasks like medical image analysis and disease prediction, their "black-box" nature raises concerns about transparency, accountability, and trust from both medical professionals and patients. The significance of explainability in healthcare applications are the following.

Clinicians need to understand the rationale behind an AI model's recommendations to trust its output and integrate it into their decision-making process. Without explanations, clinicians might hesitate to adopt AI tools, hindering their potential benefits for patient care (McKelvey et al., 2018).

Explainable models can help identify and mitigate potential biases in the training data that could lead to unfair or discriminatory outcomes for certain patient groups. By understanding how the model arrives at its decision, developers can address these biases and ensure fair treatment for all patients (Whittaker, 2019).

Patients have a right to understand the reasoning behind AI-driven diagnoses or treatment recommendations. Explainability can facilitate better communication between patients and healthcare providers, fostering patient trust and engagement in their care (Guerra-Manzanares et al., 2023).

A study by (Alghamdi, 2022) employed LIME to explain a deep learning model’s prediction for diabetic retinopathy from retinal fundus photographs. LIME identified the most relevant image regions (e.g., microvascular abnormalities, hemorrhages) influencing the model's decision, aiding ophthalmologists in understanding its reasoning.

A research project by (Jiang et al., 2021) developed an explainable AI system for personalized treatment recommendations in lung cancer. The system combined a deep learning model with SHAP to explain how factors like a patient's genomics and medical history contributed to the suggested treatment plan. This transparency empowered oncologists to discuss treatment options with patients in a more informed way.

Investigators in a study by (McKelvey et al., 2018) built an explainable machine learning model to predict hospital readmission risk for heart failure patients. The model used rule-based explanations, highlighting key factors like a patient's medications and lab test results that contributed to the predicted risk. This information helped healthcare providers tailor discharge plans and interventions for patients at higher risk.

These examples showcase how explainable AI can be implemented in various healthcare applications, fostering trust, improving communication, and ultimately leading to better patient care.

But there are challenges, the challenges and future directions for XAI in healthcare applications are: Developing effective and user-friendly explanations for complex deep learning models remains an ongoing challenge. Balancing the level of detail in explanations with user needs (clinicians vs. patients) is crucial. Regulatory frameworks for explainability in healthcare AI are still evolving.

Explainability is now a must or a necessity for the successful implementation of AI in healthcare. By fostering trust, reducing bias, and improving communication, explainable AI can unlock the full potential of AI for improving patient care and outcomes.

CHAPTER THREE

MATERIALS AND METHODS

This chapter details the materials and methods used for the whole DL workflow, including data acquisition, pre-processing, the design of the interpretable deep learning model architecture, the training process, and the evaluation metrics used to assess classification performance.

3.1 Materials

As the core materials for AI models are the **platform** to run our code, the **data** for teaching our models, and the different **libraries** and **packages** used for different operations, we will see them in the next sub-section.

3.1.1 Platform

This entire study was carried out using both Kaggle and Google Colab. They are cloud-based platforms that lets us run Python code within our web browser. This means we can use them for ML, data analysis, and other computationally intensive tasks without needing a powerful computer ourselves. One of their key features is they access to hardware accelerators, which are specialized processors that can significantly speed up certain types of computations. There are two main types of hardware accelerators available in both Colab and Kaggle: Graphics Processing Units (GPUs): GPUs are particularly well-suited for tasks that involve a lot of parallel processing, such as training deep learning models. Kaggle offers two different types of GPUs GPU T4 x 2 and GPU P100 and also a TPU type of TPU VM v3-8. And Colab also offers several different types of GPUs, including Tesla A100s, V100s, and T4s. Each type of GPU has different capabilities and performance characteristics. TPUs are custom-designed by Google for machine learning tasks. They are generally even faster than GPUs for specific machine learning workloads, but they may not be suitable for all types of computations.

3.1.2 Data

Data can be classified as local or public data based on scope, accessibility, and origin. Public data is information freely available from government agencies or public institutions, but local data is information specific to a particular geographic area, often collected by local authorities or organizations.

Only 20% of lung cancer cases are reported to occur in low- and middle-income countries. An estimated 1.5% of all Ethiopian cancers involved the lung; however, no nationwide cancer registry exists in Ethiopia. Thus, accurate data on clinical history, histopathology, molecular characteristics, and risk factors for lung cancer are not available (Gebremariam et al., 2021).

We opted for leveraging publicly available dataset curated for research purposes. Some of the datasets we have explored included LIDC-IDRI, LUNA16, Kaggle Data Science Bowl, and LUNA22-ISMI. After careful data exploratory analysis of some of the above and other additional datasets we have selected the most recent and computational inexpensive dataset for our study which is the LUNA22-ISMI dataset.

The LUNA22-ISMI (Lung Nodule Analysis 2022 - Intelligent Systems in Medical Imaging) was an educational challenge focused on classifying lung nodules in chest CT scans. It aimed to provide a platform for researchers and students to develop and test their algorithms for nodule malignancy prediction and nodule type classification [1.7 GB compressed files of 3D patches of nodules] (Venkadesh & Jacobs, 2022).

The LUNA22-ISMI dataset contains 1176 lung nodule CT scan patches derived from the LIDC-IDRI dataset for the educational LUNA22-ISMI challenge. This dataset only comprises 3D patches of nodules (size 128 x 128 x 64 in x, y, and z dimensions) that have been labeled by at least three out of four (using LUNA16 criteria, yielding 1186 nodules). The nodules are always situated in the middle of the 3D patch. The information includes radiologist scores for nodule kind and malignancy (Venkadesh & Jacobs, 2022).

The images are formatted in a compressed “nifti” format as “.nii.gz”. The labels for each nodule can be found from the given Numpy file named “*LIDC-IDRI_1176.npy”*.

3.1.3 Software

Deep learning models require specialized software libraries and frameworks. These are lists of the software tools we used for: data pre-processing, deep learning model development and model interpretation. In addition, we have used a software called ITK-SNAP for data exploration or viewing the medical images.

Data exploration and Data pre-processing:

ITK-SNAP: is an interactive software application that enables users to manually identify anatomical regions of interest and browse through or to navigate three-dimensional medical images like a CT scan.

During the data preprocessing step, we have used the following python libraries for exploratory data analysis of different datasets; let us see them one by one.

Pydicom: Python library for handling DICOM medical images. PyDicom is a free and open-source Python library specifically designed to work with DICOM files. PyDicom offers a comprehensive suite of functionalities for: Reading DICOM files: PyDicom allows you to access and parse the metadata associated with medical images, such as patient information, acquisition parameters, and modality details. Writing DICOM files: You can create and manipulate DICOM files using PyDicom, enabling data exchange and archiving. Processing medical images: PyDicom provides tools for basic image processing tasks on DICOM data, facilitating further analysis (Mason et al., 2023).

NiBabel: Python library for working with neuroimaging data. Nibabel is a free and open-source Python library that provides tools for interacting with various neuroimaging data formats. It acts as a bridge between different neuroimaging file formats and analysis tools, streamlining your workflow. Key functionalities of Nibabel include: Loading and saving neuroimaging data: Nibabel supports a wide range of neuroimaging file formats, including fMRI (NIFTI, MGH), EEG (BrainVision), and PET (ANALYZE). We can effortlessly load data from these formats for further analysis in Python. Data manipulation: Nibabel offers basic functionalities for manipulating neuroimaging data, such as cropping, masking, and reorientation. Interaction with other neuroimaging libraries: Nibabel integrates seamlessly with other popular neuroimaging libraries in Python, allowing for a more comprehensive analysis pipeline (Brett et al., 2024).

dicom2nifti: Python library for converting DICOM to Nifti format dicom2nifti is a free and open-source Python library specifically designed to convert DICOM files to Nifti format.

There are also additional python packages and modules that we have used for different computational operations.

Os: module provides a powerful set of functions for interacting with the underlying operating system. It allows your Python programs to perform various tasks related to file and directory management, process control, and environment variables.

Shutil: The shutil module in Python provides a collection of high-level functions for working with files and collections of files. It offers functionalities that go beyond the basic file operations available in the built-in os module, simplifying common tasks like copying, moving, and removing files and directories.

Numpy: short for Numerical Python, is a fundamental library for scientific computing in Python. It offers a powerful set of tools for working with multidimensional arrays and matrices, along with mathematical functions for efficient operations on numerical data**.**

Pandas: is a powerful and popular open-source library in Python specifically designed for data analysis and manipulation. It excels at working with tabular data, similar to spreadsheets or SQL tables, making it a go-to tool for data scientists and analysts.

Scipy: pronounced "Sigh-pi" is another cornerstone library in the scientific Python ecosystem. Building on top of NumPy, SciPy extends its functionality by providing a vast collection of algorithms and tools for various scientific computing tasks. We used it for data augmentation and other Numpy array operations.

Matplotlib: is a fundamental library for creating static, animated, and interactive visualizations in Python. It offers a versatile and user-friendly toolkit for generating various plot types to explore and communicate data insights effectively.

Deep learning model development:

TensorFlow: An Open-Source Framework for Machine Learning. TensorFlow is a free and open-source software library developed by Google for numerical computation and large-scale machine learning. It provides a flexible architecture that allows you to define, train, and deploy machine learning models efficiently. Key functionalities of TensorFlow include: Dataflow Programming: TensorFlow utilizes dataflow graphs to represent computations. These graphs define the relationships between data elements and the operations performed on them. This approach facilitates efficient execution on various hardware platforms, including CPUs, GPUs, and TPUs. Automatic Differentiation: TensorFlow offers automatic differentiation capabilities, which are crucial for training machine learning models.

It calculates the gradients of a function with respect to its inputs, allowing optimization algorithms to adjust the model's parameters effectively. Tensor Operations: A core concept in TensorFlow is the "tensor," a multidimensional array of numerical data. TensorFlow provides a rich set of operations for manipulating and transforming tensors, forming the building blocks for machine learning algorithms. Pre-built Libraries and Tools: TensorFlow offers a wide range of pre-built libraries for specific tasks like natural language processing, computer vision, and recommender systems. Additionally, it provides tools for model deployment, serving, and visualization (TensorFlow, 2019).

Keras: High-level API for Deep Learning with TensorFlow. Keras is a high-level API designed to simplify the development of deep learning models. It sits on top of frameworks like TensorFlow, providing a user-friendly interface for building, training, and deploying deep neural networks. Key functionalities of Keras include: Model Building Blocks: Keras offers a set of pre-built layers that serve as the building blocks for deep learning models. These layers include convolutional layers (CNNs) for image processing, recurrent layers (LSTMs) for sequence data, and dense layers for fully-connected networks. Sequential vs. Functional API: Keras provides two primary approaches for constructing deep learning models: Sequential API: This simpler API is suitable for building linear stacks of layers, often used for basic deep learning models. Functional API: This more flexible API allows for defining complex model architectures with branching, merging, and custom layers, providing greater control for advanced deep learning tasks. Loss Functions and Optimizers: Keras incorporates various pre-defined loss functions (like mean squared error or categorical cross-entropy) to evaluate model performance during training. Additionally, it offers a suite of optimizers (like gradient descent variants) that automatically adjust model parameters based on the loss function. Ease of Use: Compared to directly using TensorFlow, Keras simplifies deep learning development with its intuitive API and focus on essential model building components (Keras, 2019).

3.2 Methods

Our methodological approach encompassed four key stages of the work and they are explained on the next sub sections.

3.2.1 Exploratory Data Analysis

Our initial stage involved a comprehensive exploratory data analysis (EDA) of potential datasets. This included data acquisition and the EDA focused on understanding the data characteristics, including medical imaging data characteristics and how the data is labeled.

After exploring and viewing the CT scan images using ITK Snap software; we used Nibable to access the pixel data and the header information. Below is a typical histogram of pixel values or Hounsfield units of a single 3D patch of nodule.

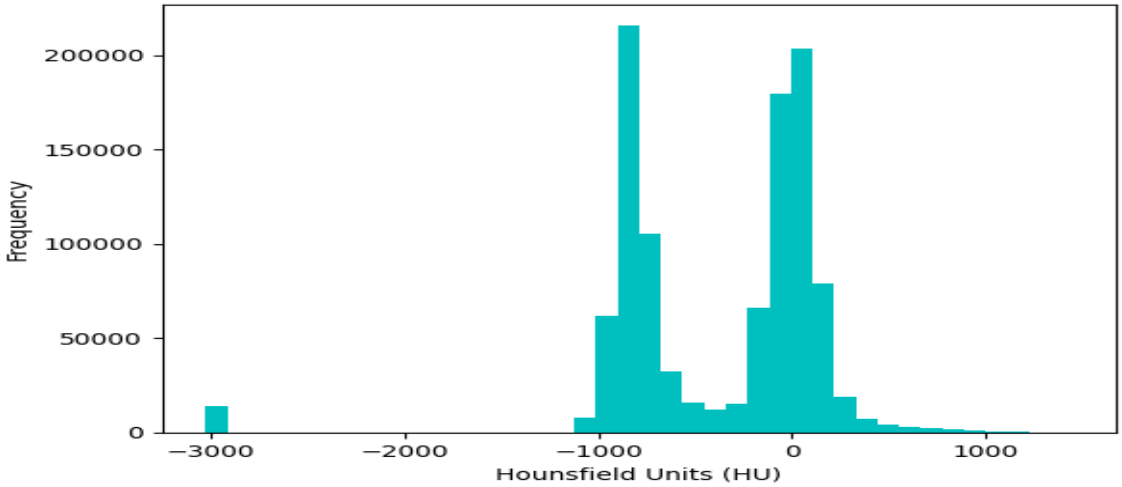


Figure 3.1 Histograms of Pixel values of a single 3D patch of nodule

As we can see from the above figure the pixel values for this sample CT scan range from more than +1000 to less than -3000 and each patch has a size of 128x128x64 in x, y, z directions. After going through all pixel values of each 3D nodule patches, we have found that the pixel values range from (-3,024.0) to (+6,054.0).

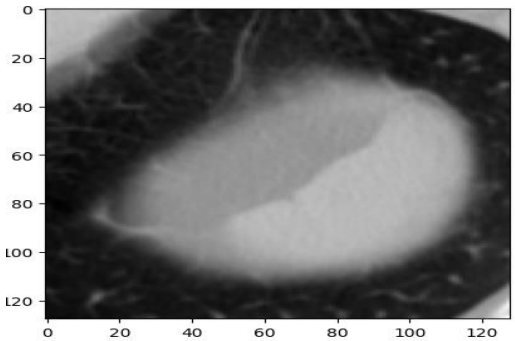


Figure 3.2 A CT scan slice of a sample 3D patch nodule

The labels were obtained from the “LIDC-IDRI_1176.npy” file. This file stores information for each nodule in a structured format using Python dictionaries. The file contains a NumPy array, where each element represents a single nodule. Each nodule dictionary includes the following key-value pairs relevant for label extraction and preprocessing:

```
{
'SeriesInstanceUID':
'1.3.6.1.4.1.14519.5.2.1.6279.6001.100225287222365663678666836860',
 'VoxelCoordX': 45,
 'VoxelCoordY': 211,
 'VoxelCoordZ': 77,
 'Diameter': [6.97167141, 6.97167141, 7.34878692, 5.94228451],
 'Texture': [5, 5, 5, 5],
 'Malignancy': [4, 2, 4, 2],
 'Calcification': [6, 6, 6, 6],
 'Filename':
'1.3.6.1.4.1.14519.5.2.1.6279.6001.100225287222365663678666836860_45_21
1_77_0000.nii.gz'
}
```

Where, SeriesInstanceUID: Unique identifier for the patient CT scan series, VoxelCoordX, VoxelCoordY, VoxelCoordZ: 3D coordinates of the nodule center within the CT scan volume, Diameter: An array containing the nodule diameter measurements along different axes, Texture (1-5): Categorical variable representing the radiographic solidity (internal texture) of the nodule, ranging from non-solid (1) to solid (5), Malignancy (1-5): Categorical variable representing the subjective assessment of malignancy likelihood, considering a 60-year-old male smoker, with 1 being highly unlikely and 5 being highly suspicious, Calcification (1-6): Categorical variable indicating the presence and extent of calcification within the nodule (Venkadesh & Jacobs, 2022).

We used the malignancy score for Malignancy Binarization: The original five-class malignancy labels were transformed into a binary classification problem. This simplifies the model's objective and focuses on distinguishing between malignant and benign nodules. Labels 1 and 2 were combined to represent benign (low malignancy suspicion), while labels 3, 4, and 5 were grouped to represent malignant (high malignancy suspicion). Following label extraction from the "malignancy" key-value pair, a median score was calculated to represent the overall malignancy assessment. Nodules with a median score below a predefined threshold of 3 were classified as benign, while those with a score of 3 or above were classified as malignant. This approach aimed to consolidate potentially subjective assessments from multiple radiologists into a single, representative label for each nodule.

However, this classification process revealed a class imbalance within the dataset. Out of the 1,176 labeled 3D nodule patches, 380 were categorized as benign, and 796 belonged to the malignant class. To address this imbalance and ensure the model learns effectively from both classes, appropriate data augmentation techniques were implemented during the preprocessing stage.

3.2.2 Data Preprocessing

In this stage, we have done data splitting and employed some preprocessing techniques on the dataset. To prevent data leakage, we split the dataset before augmentation.

Data Splitting: we split the dataset in a ratio of 80% for training and 10% for validation used for testing the models with unseen data during training, and 10% for testing after training. This split ensured the model was trained on a representative portion of the data, evaluated on a separate set to prevent overfitting, and ultimately tested on unseen data for robust performance assessment. After splitting we have: 304 and 637 3D CT scan patches of nodules for benign and for malignant classes respectively, for training set. For validation set we have 38 and 79 3D CT scan patches of nodules for benign and for malignant classes respectively. And for test set we have 38 and 80 3D CT scan patches of nodules for benign and for malignant classes respectively.

Data Augmentation is used to original imbalanced dataset and make equal number of 3D CT scan patches of nodules for both classes. And also, to improve the generalizability of the model and overcome the problem of overfitting during training. we employed data augmentation techniques on the training set. This process artificially increases the training data by introducing controlled variations of existing samples, enhancing the model's ability to learn features robust to these variations. There were different augmentation techniques that we have come up with, some of them are: Flipping, Scaling, Translation, Noise addition, Brightness variation, Contrast Variation, Elastic Deformation, Median filtering, Rotating by some angles and a combination of three or two of the techniques. All of those techniques were done using the SciPy library.

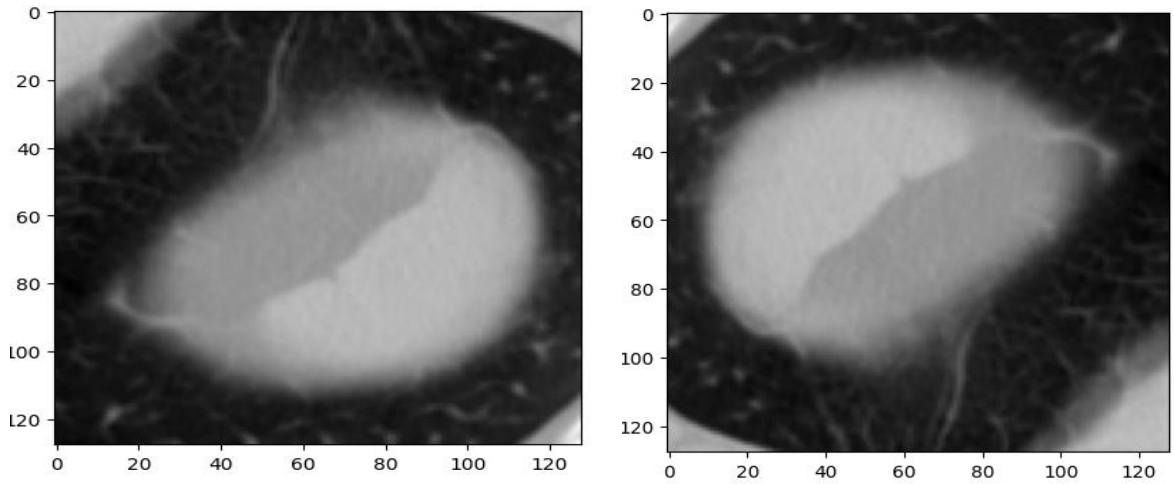
But even if we employed up to ten augmentation techniques, we only used the data that we have gotten from only seven techniques. This is done because of shortage of RAM during training.

We encountered RAM out of memory or “your notebook tried to allocate more memory than is available. It has restarted.” during training. The seven augmentation techniques used are as follows with their visualizations.

Flipping which is one of geometric transformations types. The 3D nodule patches were flipped along two randomly selected axes out of the three dimensions. This simulates potential variations in nodule orientation within the lung scans. Scaling; scales the image by a random factor between 0.8 and 1.2 along each axis independently. Translation; translates or shifts the image by a random amount within a range of -10 to 10 pixels along each axis.

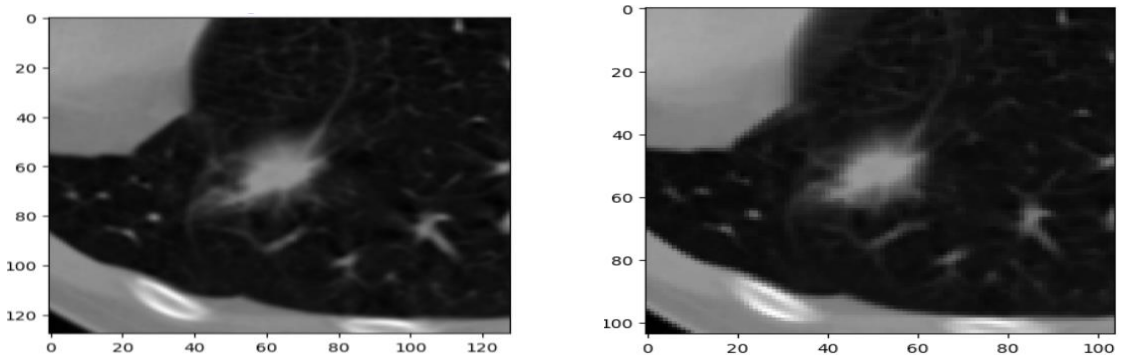
Noise addition; adds Gaussian noise to the image with a random standard deviation between 0.05 and 0.2. Contrast adjustment; applies shear transformation to the image with random shear factors between -0.2 and 0.2 along each axis. Brightness adjustment, adjusts the brightness of the image by a random factor. Shifts the intensity values of a 3D image by -30 to 20 factor to make it brighter or darker. Elastic Distortion or spline filtering, Elastic distortion introduces random, localized deformations to the 3D nodule patches. This technique mimics slight anatomical variations or scanner imperfections that might occur in real-world CT scans, improving the model's resilience to such artifacts.

The SciPy library provided the necessary functionalities for implementing these data augmentation techniques on the pixel intensity arrays of the 3D nodule patches. Figure 3.4 and 3.5 illustrate the effects of these augmentation techniques on sample nodule slices with how the original pixel intensity values changed.



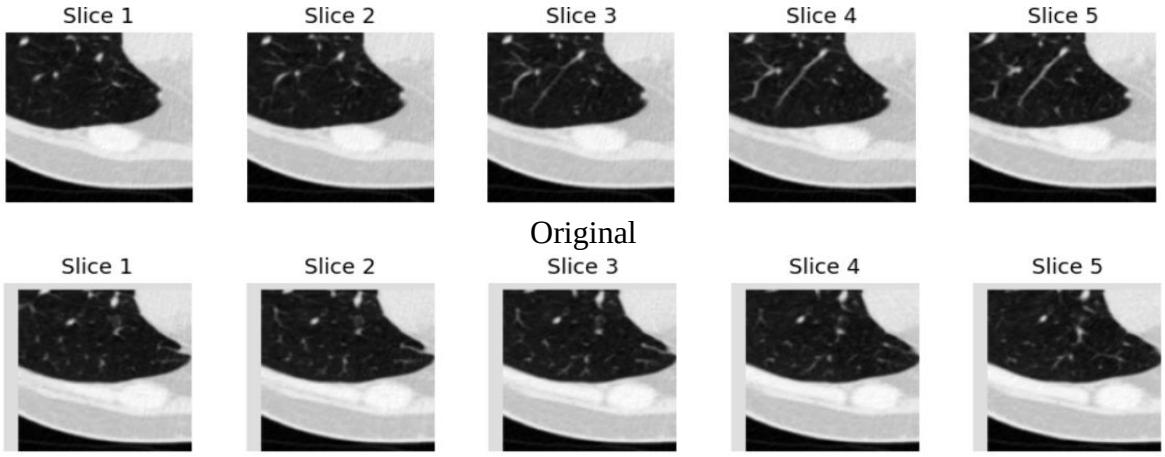
Original Flipped

(a) Randomly Flipped nodule patch CT scan slice



Original Scaled

(b) Randomly Scaled nodule patch CT scan slices and the size changed to (104, 104, 52)

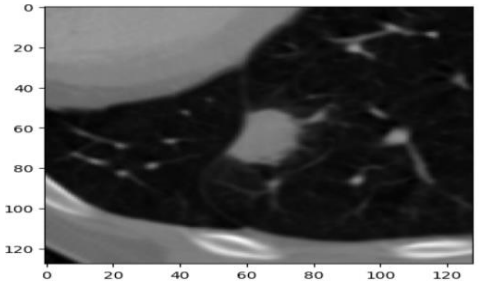


Original

Translated

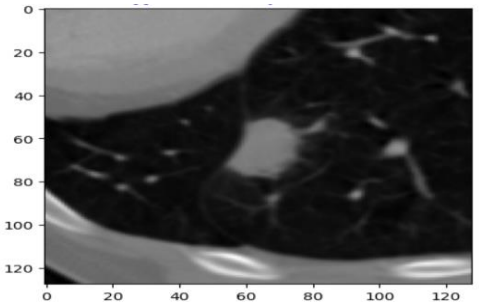
(c) Randomly translated nodule patch CT scan slices

Figure 3.3 The effect of Flipping, Scaling and Translation on a sample CT scan of a nodule patch from the Luna22 dataset



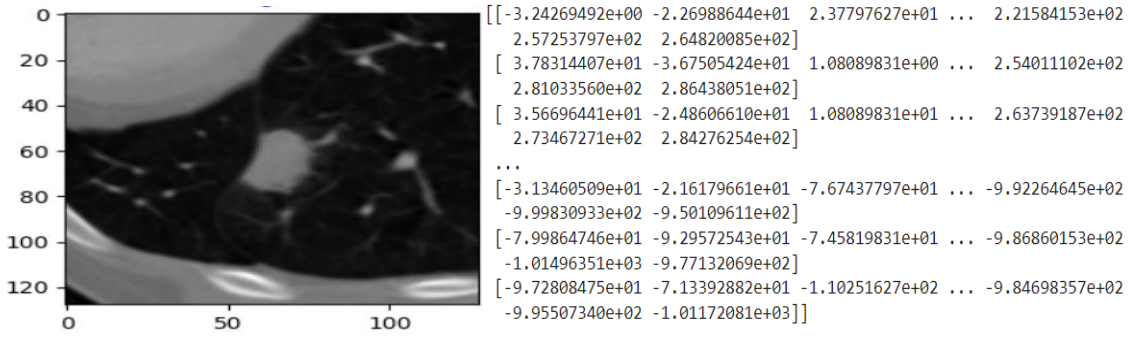
```
[[[ 4.300e+01 -2.700e+01  3.000e+00 ...  1.530e+02  1.280e+02  2.160e+02]
  [ 2.700e+01  9.000e+00  3.600e+01 ...  1.840e+02  1.870e+02  2.460e+02]
  [ 5.800e+01  5.100e+01  2.200e+01 ...  2.000e+02  2.420e+02  2.660e+02]
  ...
  [-5.800e+01 -7.200e+01 -1.220e+02 ... -8.840e+02 -9.000e+02 -8.740e+02]
  [-8.300e+01 -1.000e+00 -4.800e+01 ... -9.050e+02 -9.090e+02 -8.540e+02]
  [-2.400e+01  6.300e+01 -5.900e+01 ... -9.170e+02 -9.210e+02 -8.740e+02]]
```

(a) Original nodule CT scan slice and its pixel intensity values

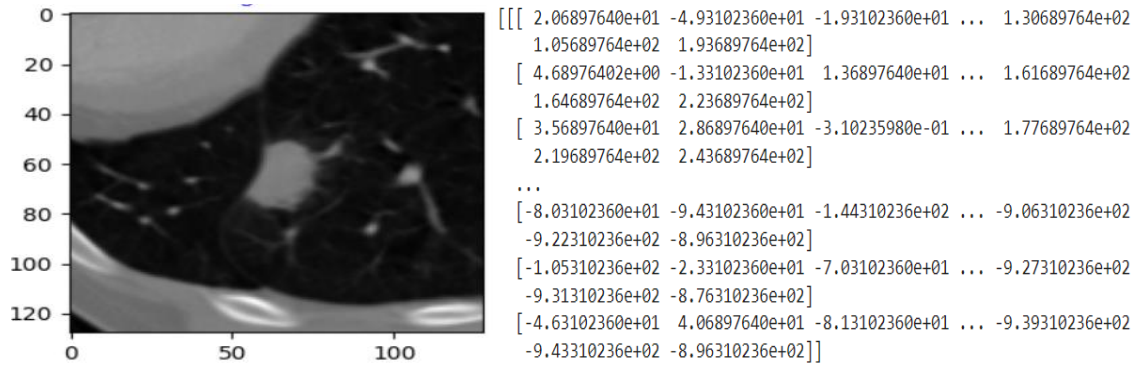


```
[[[ 4.29941813e+01 -2.69971504e+01  2.93395567e+00 ...  1.52987623e+02
    1.27847917e+02  2.15758995e+02]
  [ 2.71642656e+01  8.98332893e+00  3.59930802e+01 ...  1.83878053e+02
    1.86964549e+02  2.46127793e+02]
  [ 5.79288367e+01  5.08976397e+01  2.20644403e+01 ...  2.00137844e+02
    2.42068730e+02  2.66043488e+02]
  ...
  [-5.81856579e+01 -7.21031782e+01 -1.21897311e+02 ... -8.83966156e+02
   -8.99973692e+02 -8.74031995e+02]
  [-8.29618448e+01 -9.21269176e-01 -4.80088576e+01 ... -9.04899202e+02
   -9.09079907e+02 -8.54046207e+02]
  [-2.41342325e+01  6.30429400e+01 -5.90332393e+01 ... -9.17003015e+02
   -9.20954246e+02 -8.74112107e+02]]
```

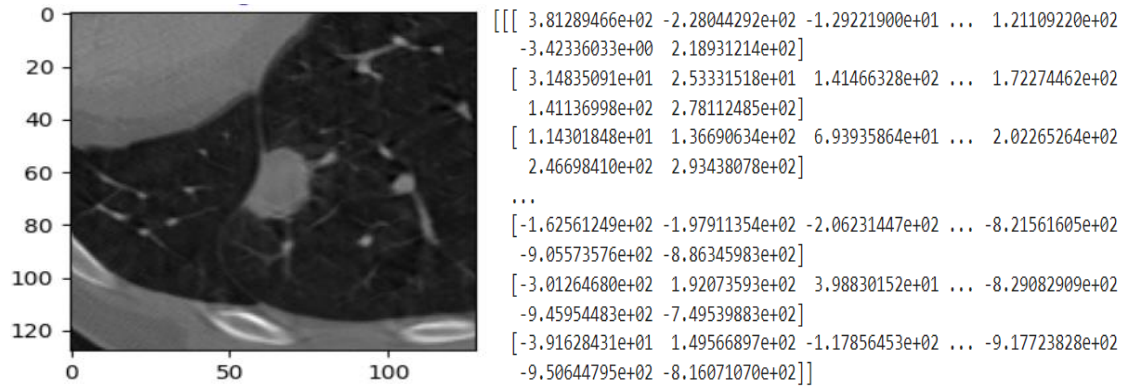
(b) Random Gaussian noise added CT scan slice and its pixel intensity values



(c) Randomly Contrast adjusted CT scan slice and its pixel intensity values



(d) Randomly Brightness adjusted CT scan slice and its pixel intensity values



(e) Spline filtered or elastic distorted CT scan slice and its pixel intensity values

Figure 3.4 The effect of noise addition, contrast and brightness adjustment, and elastic distortion for augmentation (The pixel intensity values are shown to depict how the 3D image is changed.)

Following data augmentation, we applied two essential preprocessing steps commonly used for chest CT scans before training a deep learning model:

Windowing: Building upon the established understanding of Hounsfield Units (HU) for representing CT scan voxel intensities (refer to Chapter 2), we explore the data preprocessing steps applied to the lung nodule patches within this dataset. The raw HU values in our dataset range from -3024.0 to +6054.0. Considering the typical HU ranges for various tissue types

in chest CT scans, we strategically clip the voxel intensities to focus on the lung parenchyma, the region of interest for nodule classification. Specifically, we employ a clipping threshold of -1000 HU for air and 400 HU for bone. This process ensures that voxels with intensities exceeding the bone threshold (potentially representing unrelated anatomical structures or image artifacts) are clipped to 400 HU. Similarly, voxels with intensities lower than the air threshold (potentially representing background noise) are set to -1000 HU.

Normalization: (Min-Max Scaling): Following data augmentation, the other crucial technique used was normalization. This step is applied to the lung nodule patches before feeding them into the deep learning model for training. Normalization aims to scale the pixel intensity values within a specific range, promoting numerical stability and improving the learning process. In this work, we employ a min-max scaling technique combined with zero-centering. The min-max scaling process is achieved using the following formula:

$$Volume_{normalize} = (Volume - HU_{Min})/(HU_{Max} - HU_{Min}) \qquad eq.\ (3.1)$$

Where: volume represents the original voxel intensity array of the lung nodule patch. HU_{min} and HU_{max} denote the minimum and maximum HU values encountered within the 3D patch, respectively. This formula effectively scales each voxel intensity within the original range (HU_{min} to HU_{max}) to a new range between -1 and 1. This compressed range facilitates the learning process for the deep learning model by ensuring all features reside on a similar scale. By incorporating these data augmentation and preprocessing techniques, we aimed to create a more robust and generalizable model for lung nodule classification.

Following the aforementioned preprocessing steps, the 3D lung nodule patches undergo a final stage of resizing to conform to the input requirements of the chosen deep learning model architecture. This resizing process ensures compatibility between the data and the model's expected input dimensions. we specifically employ a technique known as spline interpolated zoom (SIZ) for resizing the 3D patches. SIZ leverages spline interpolation, a mathematical method for creating smooth curves or surfaces that pass through a set of data points (Zunair et al., 2020). This approach offers advantages over simpler resizing techniques like nearest neighbor interpolation, which can introduce artifacts during the resizing process.

Following SIZ resizing, the 3D image voxel arrays are converted into a format suitable for deep learning models. In this work, we leverage NumPy arrays to represent the preprocessed and resized 3D patches. Each voxel intensity value within the array is represented by a single

numerical value. Additionally, labels are assigned to each patch, with "0" indicating benign nodules and "1" signifying malignant nodules. This binary labeling scheme facilitates the training process for a classification task.

Visualization: Throughout the preprocessing stage, data visualization techniques were potentially utilized to gain further insights into the data distribution, identify potential anomalies, and assess the effectiveness of the applied preprocessing steps.

3.2.3 Deep Learning Model Development and Evaluation

This section focuses on the development, training, and evaluation of a deep learning model for the task of lung nodule classification in chest CT scans. As noted earlier, Deep Neural Networks (DNNs) have emerged as powerful tools for automated image-based analysis, achieving remarkable performance in various classification problems. Due to the inherent three-dimensional (3D) nature of chest CT data, we opted to leverage 3D Convolutional Neural Networks (3D CNNs) to exploit the spatial information within the volumetric data. Compared to their 2D counterparts, 3D CNNs utilize 3D kernels or filters to capture spatial relationships across all three dimensions (depth, width, and height). While offering superior performance in processing 3D data, 3D CNNs come at the cost of increased computational complexity. The figure below shows 3D convolution.

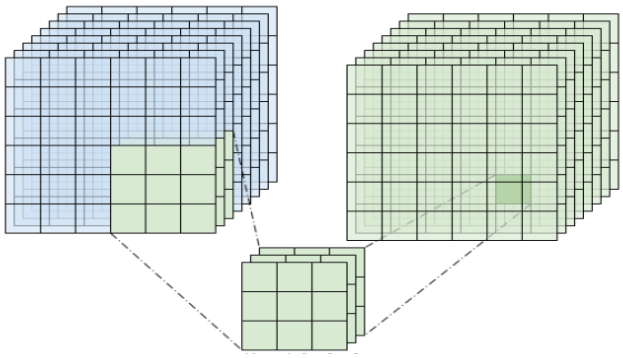


Figure 3.5 3D Convolution

Baseline Model Architecture and Rationale

To establish a baseline for comparison and explore the feasibility of leveraging simpler architectures, we employed a straightforward 3D CNN architecture inspired by the work of (Zunair et al., 2020). Their work explored methods for processing CT scans using 3D CNNs for tuberculosis prediction. Additionally, we investigated the effectiveness of adapting established 2D CNN architectures, namely AlexNet and VGG, into 3D CNN variants.

The detailed configurations of these models and their training will be presented in the subsequent chapter.

The rationale behind this model selection strategy is twofold. Firstly, considering the inherent computational demands of 3D CNNs, we aimed to evaluate the performance of a relatively simple 3D CNN architecture as a baseline. Secondly, we explored the potential of adapting well-established 2D CNN architectures, leveraging their success in image classification tasks, into the 3D domain for lung nodule classification.

Model Evaluation

Following the development and training of the deep learning models, a rigorous evaluation process is essential to assess their generalizability and performance on unseen data. This evaluation is typically conducted using a separate testing dataset that the models have not encountered during training. This ensures an unbiased assessment of the model's ability to classify nodules in real-world scenarios.

To comprehensively evaluate the models' performance, we employed a range of established performance metrics commonly used in binary classification tasks. The following were our performance metrices.

Accuracy: is the most fundamental metric, representing the overall proportion of correctly classified nodules. It is calculated as the sum of true positives (TP) and true negatives (TN) divided by the total number of samples. While offering a basic overview of model performance, accuracy can be misleading in imbalanced datasets.

$$Accuracy = (TP + TN)/(TP + TN + FP + FN) \qquad eq\ (3.2)$$

Precision: Precision focuses on the model's ability to identify true positive cases. In the context of lung nodule classification, it indicates the proportion of nodules identified as malignant by the model that are indeed confirmed as malignant by the radiologists. A high precision value signifies a low false positive rate, meaning the model is not prone to misclassifying benign nodules as malignant.

$$Precision = TP/(TP + FP) \qquad eq.\ (3.3)$$

Recall (Sensitivity): Recall, also known as sensitivity, emphasizes the model's ability to detect all actual positive cases (malignant nodules).

It is calculated as the proportion of true positives identified by the model out of the total number of actual positive cases present in the testing data. A high recall value suggests the model has a low false negative rate, meaning it effectively identifies most malignant nodules.

$$Recall = TP/(TP + FN) \qquad \textit{eq. (3.4)}$$

F1-Score: The F1-score offers a balanced view of the model's performance by incorporating both precision and recall into a single metric. It is the harmonic mean of precision and recall, providing a more comprehensive assessment than either metric alone. A high F1-score signifies a model that excels in both identifying true positives and minimizing false positives.

$$F1 - Score = 2 * (Precision * Recall)/(Precision + Recall) \qquad \textit{eq. (3.5)}$$

Area Under the ROC Curve (AUC): The Receiver Operating Characteristic (ROC) curve is a graphical representation that depicts the model's performance at various classification thresholds. The AUC metric quantifies the overall performance of the model by measuring the total area under the ROC curve. A higher AUC value indicates better discriminative ability, signifying the model's effectiveness in distinguishing between benign and malignant nodules.

Confusion Matrix: In addition to the performance metrics mentioned above, we also employed a confusion matrix to visualize the model's performance. A confusion matrix is a tabular representation that summarizes the distribution of prediction outcomes for the testing data. It provides insights into the number of correctly and incorrectly classified cases, further aiding in the evaluation of the model's performance.

By employing this comprehensive evaluation strategy, we gain valuable insights into the strengths and weaknesses of the deep learning models for lung nodule classification. This information can then be used to refine the model architecture, training parameters, or data preprocessing techniques to achieve optimal performance in this critical medical application.

3.2.4 Deep Learning Model Interpretation Using Grad-CAM:

In this stage, we explored the use of Gradient-weighted Class Activation Mapping (Grad-CAM) for interpreting the predictions of our deep learning model for lung nodule classification. Grad-CAM is a technique that aims to provide visual insights into the regions of a lung nodule image that contribute most significantly to the model's classification decision (malignant vs. benign).

The details of the implementations used will be discussed on the next chapter. The general block diagram of our methodology for implementing our proposed model looks is depicted on figure 3.7.

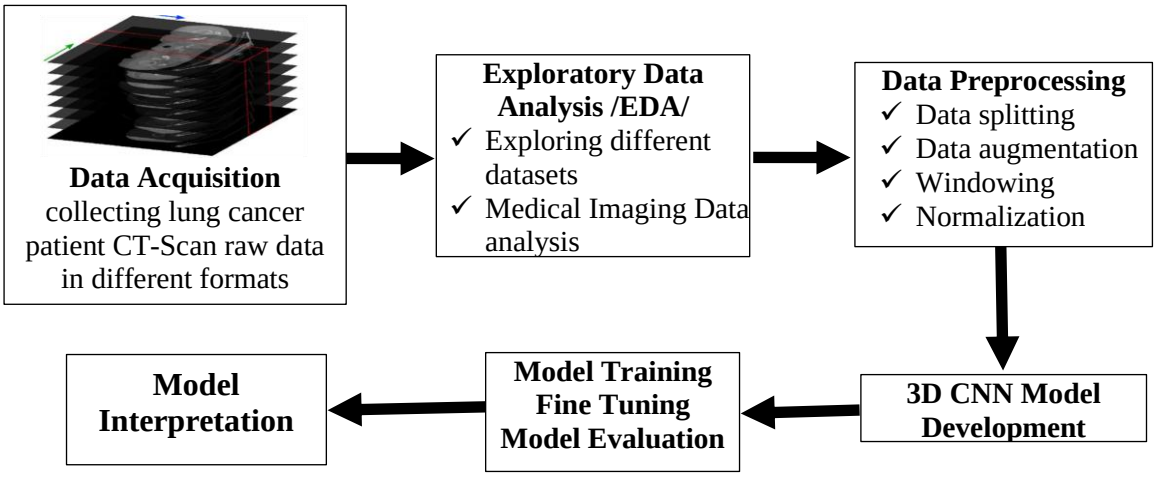


Figure 3.6 Block diagram summary of project work methodology

CHAPTER FOUR

IMPLEMENTATION OF PROPOSED MODEL

4.1 Introduction

This chapter focuses on the development and evaluation of the four different 3D Convolutional Neural Network (CNN) models for lung nodule classification in chest CT scans. The organization of the chapter is as follows: Model Architectures: This section provides a detailed description for all the four 3D CNN architectures. It will explain the number and type of convolutional layers, pooling layers, activation functions, and any other relevant components employed in the models. Diagrams or visualizations can be included to enhance clarity and understanding. For simplicity we labeled the baseline model as Model-1, the 3D AlexNet Model as Model-2, the initially proposed 3D CNN model as Model-3 and the proposed 3D CNN model with CBAM as Model-4.

Training and Evaluation Procedures: This section delves into the training process utilized for the 3D CNN models. It explains the experiment design or the hyperparameters used during model compilation and training. These includes the optimizers we have used, learning rate scheduling strategy, and the loss function used for evaluation.

Finally, we will address the interpretability aspect of the models. It will delve into the details of using Gradient-weighted Class Activation Mapping (Grad-CAM) for interpreting the predictions of the 3D CNNs. This section explains how Grad-CAM provides visual insights into the regions within lung nodule images that contribute most significantly to the model's classification decision (malignant vs. benign) and how it was implemented.

During our data exploration and model development, computational constraints influenced our choice of platform between Kaggle and Colab. Colab, despite its limited RAM and GPU on the free tier, offered larger disk space for storing the large-sized lung CT scan datasets. This made it suitable for data exploration purposes. Conversely, Kaggle provided more robust computational resources, including greater RAM and GPU capabilities, but lacked the necessary disk space for handling large-scale CT scan data. Given the computationally intensive nature of training 3D CNN models, which typically demand more memory and processing power than 2D CNNs, we opted for Kaggle as our primary platform for model building and training.

4.2 Preprocessing Techniques

As outlined in Chapter 3, we divided the dataset into three distinct subsets: training, validation, and testing. This partitioning was executed prior to any data augmentation measures to safeguard against data leakage, thereby ensuring the integrity of our evaluation process. To address the inherent data imbalance and mitigate the risk of overfitting within our training data, we strategically implemented data augmentation techniques. This augmentation step effectively expanded the training set, enhancing its diversity and robustness.

Subsequently, we carried out windowing and normalization procedures on all three datasets. Windowing served to constrain intensity values within a predetermined range. Normalization, on the other hand, standardized the volume intensities to a specific range (-1 to 1), facilitating efficient model convergence and improving generalization performance.
Following these transformations, we resized the 3D volumes to a uniform dimension of 64x64x64. This standardization is crucial for compatibility with deep learning models, which often require fixed input sizes. Additionally, we introduced a channel dimension to the volumes, aligning them with the standard format expected by convolutional neural networks (CNNs).

Finally, we assigned binary class labels to each volume: 0 for benign (cancer-negative) cases and 1 for malignant (cancer-positive) cases. This labeling was consistently applied across all training, validation, and testing sets, ensuring consistency and facilitating evaluation of the model's predictive capabilities.

4.3 Model Architectures Used

This section details the development of the proposed 3D Convolutional Neural Network (CNN) models for lung nodule classification in chest CT scans. The Keras functional API was employed to construct these models; due to its flexibility in handling complex architectures. The functional API allows for the creation of models with non-linear topologies, shared layers, and multiple inputs and outputs, making it suitable for designing deep learning models for various tasks. It is more flexible than Keras Sequential API (Fchollet & Team Keras, 2019).

The development process involved three key steps: (1) Define input to the model, involved defining the model's input layer, specifying the dimensions of the input tensor that represents a 3D lung nodule patch extracted from a chest CT scan. (2) Defne a set of interconnected layers on the input. A series of interconnected layers were defined, building the core architecture of the model. These layers typically include convolutional layers for feature extraction, pooling layers for dimensionality reduction, and activation functions for introducing non-linearity. The specific types and configurations of these layers varied depending on the chosen architecture. (3) Define the model using the input and output layers.

Four distinct 3D CNN architectures were employed in this study. The first architecture served as a baseline model and was inspired by the work done for processing CT scans using 3D CNNs for tuberculosis prediction in (Zunair et al., 2020). Their research explored the application of 3D CNNs for tuberculosis prediction. This baseline model provided a foundation for comparison and allowed us to assess the effectiveness of other 3D convolutional architectures for lung nodule classification.

The second architecture was 3D adaptations of well-established 2D CNN model, namely AlexNet. The adaptations aimed to leverage the success of the architecture in image classification tasks while extending their capabilities to process volumetric data like 3D lung nodule patches. The third architecture was built by modifying and combining the two architectures to have greater results during training and evaluation. The fourth architecture was developed by adding CBAM Module (Convolutional Block Attention Module). The next subsections are model architecture depiction diagrams and their implementation code snippets of the four models highlighting the specific configurations and any modifications implemented to enhance their performance for lung nodule classification. All of them are for an input shape of (64, 64, 64). Model summaries are provided in Appendix D.

4.3.1 Model-1 / 3D-CNN Baseline Model

Our baseline model is a simple 3D-CNN model with four convolutional layers, the model architecture is depicted on Figure 4.1. We added ‘kernel_regularizer=l2(l2_reg)’ to all Conv3D and Dense layers to apply L2 regularization which is a valuable technique for improving the generalization performance and for preventing overfitting in deep learning models.

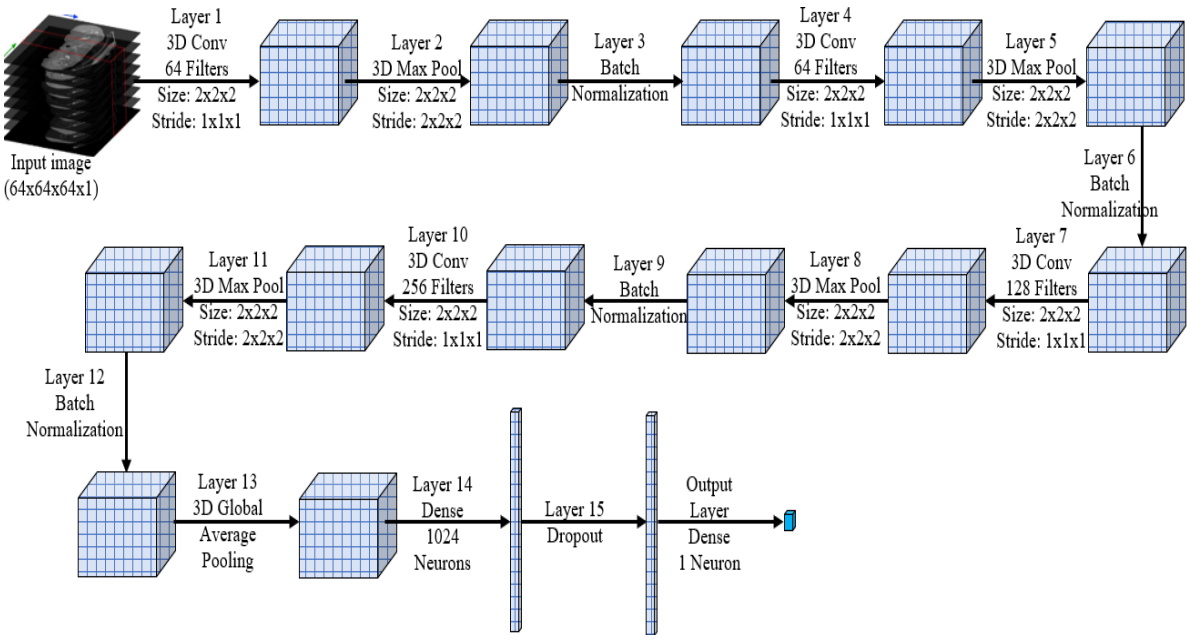


Figure 4.1 Baseline Model Architecture

4.3.2 Model-2 / 3D-AlexNet Model

AlexNet, developed by Alex Krizhevsky in 2012, is a pioneering CNN architecture that achieved remarkable results on the ImageNet Large Scale Visual Recognition Challenge (ILSVRC). Its key features include: Convolutional layers with filters that can detect specific features within images. For its time, AlexNet was considered a deep network with five convolutional layers followed by three fully-connected layers. This depth allowed the model to learn complex relationships between features at different levels of abstraction.

AlexNet introduced the ReLU (Rectified Linear Unit) activation function as an alternative to the traditional sigmoid function. ReLU offers faster training and improved performance in deeper networks. To prevent overfitting, AlexNet incorporated a dropout layer after some fully-connected layers. Dropout randomly drops a certain percentage of neurons during training, encouraging the network to learn robust features that are not dependent on specific neurons.

While AlexNet achieved groundbreaking results, it is a computationally expensive model compared to more recent architectures. Modern CNNs often have deeper structures and leverage techniques like batch normalization to improve training efficiency. Overall, AlexNet remains a significant milestone in deep learning history. Its innovative architecture and successful application in image recognition continue to influence the field of computer vision.

We used the architecture used in (Rani et al., 2022), it was an efficient 3D AlexNet architecture for object recognition in MRI brain tumor classification. We added batch normalization and drop out layers, to improve training efficiency. The model architecture is depicted on Figure 4.2.

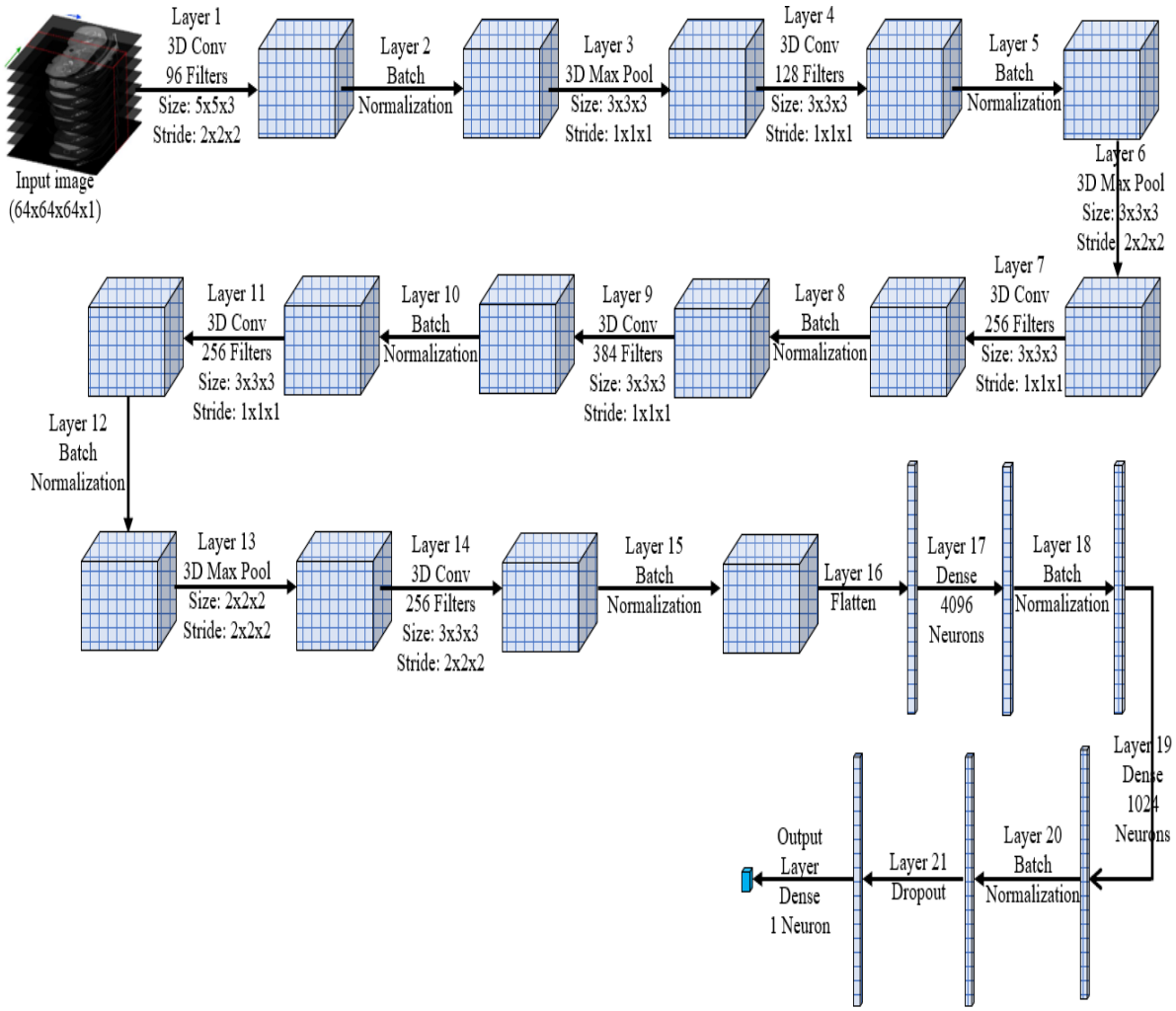


Figure 4.2 3D AlexNet Model Architecture

4.3.3 Model-3 / Proposed 3D CNN Model

While trying to achieve strong performance and experimenting with different architecture, we have come up with the following architecture. The model architecture is depicted on Figure 4.3.

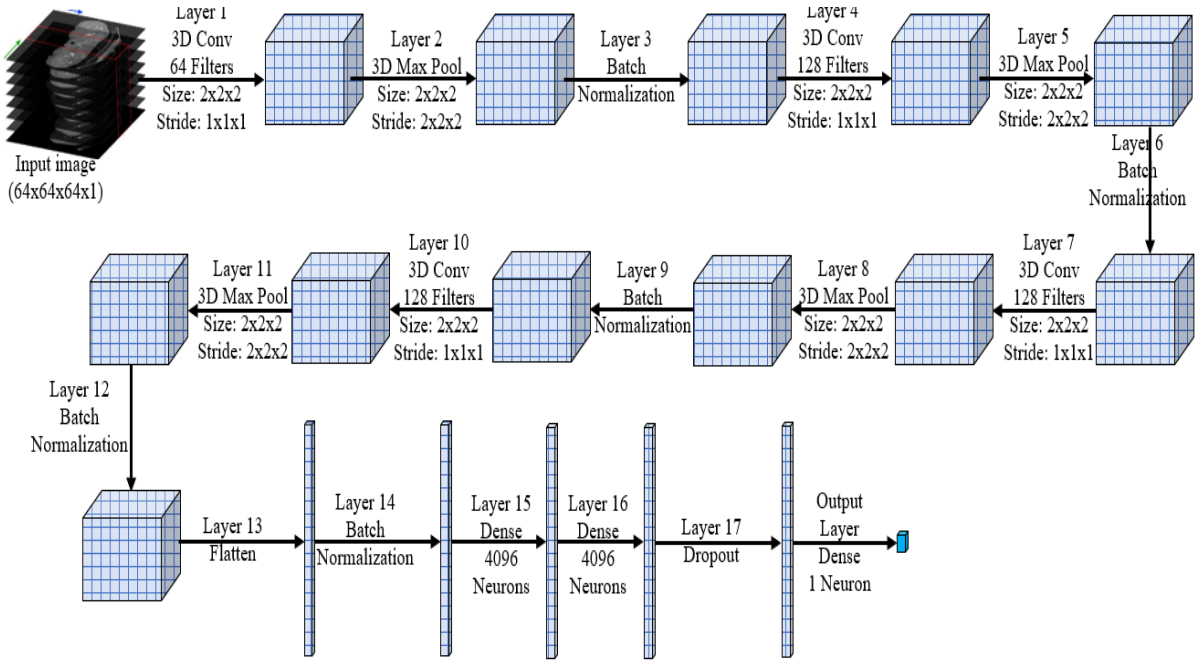
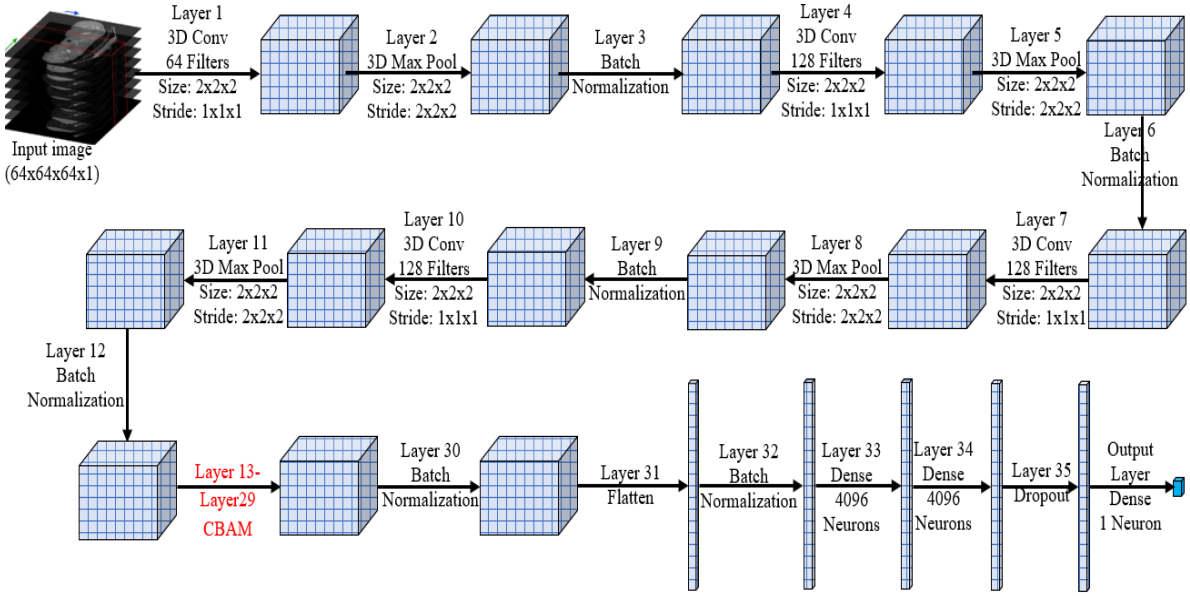


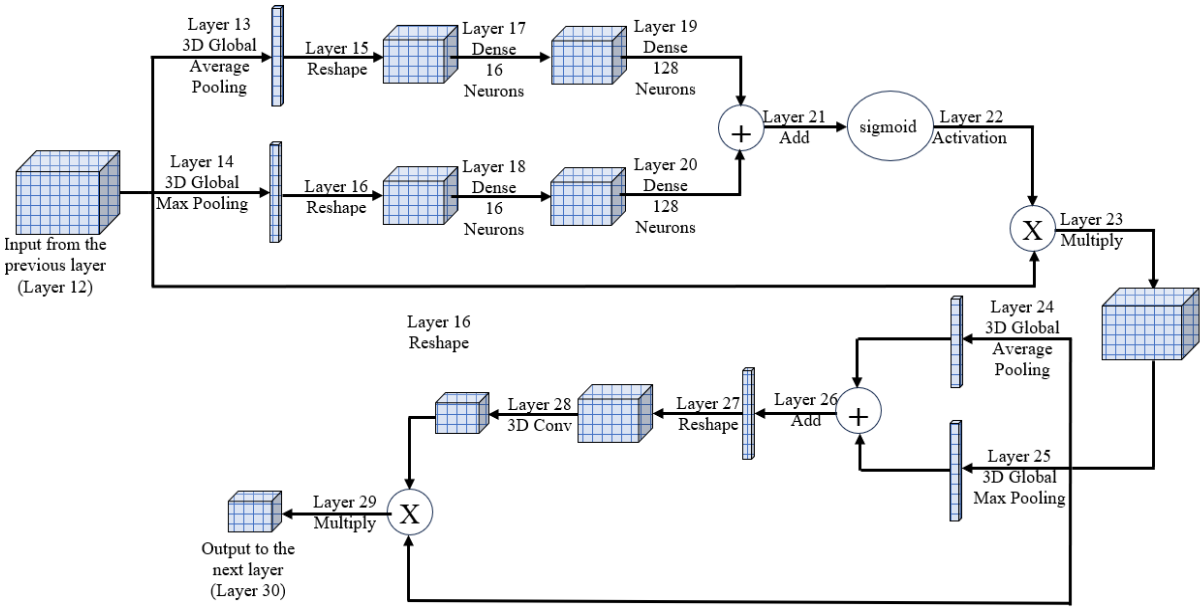
Figure 4.3 Proposed Model Architecture

4.3.4 Model-4 / Proposed 3D CNN Model with CBAM

We did additional experiments to our proposed model to achieve strong performance and come up with the idea of integrating a CBAM module (Convolutional Block Attention Module) after the last convolutional layer and before the flattening layer. This allowed the model to focus on the most informative features before proceeding to the dense layers. The overall structure remains similar to our original proposed model, but now it benefits from the attention mechanism provided by CBAM. The model architecture is depicted on Figure 4.4 followed by the corresponding code implementation.



(a) The overall model architecture



(b) CBAM Module

Figure 4.4 Proposed Model with CBAM Architecture

4.3.5 Model Training and Evaluation

For all the three models we followed the steps depicted on the block diagram in figure 4.5.

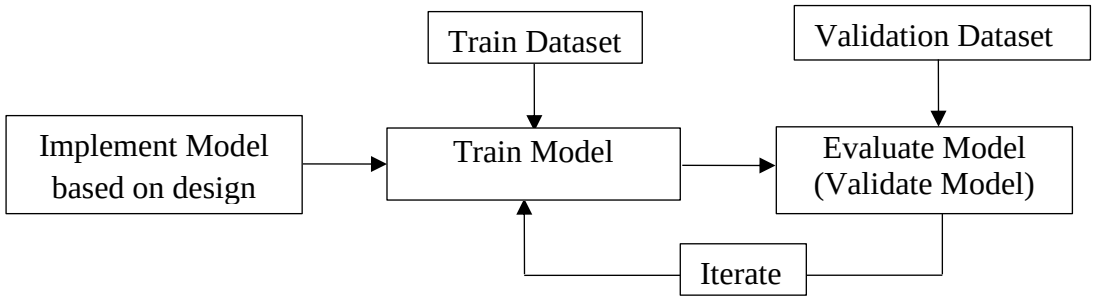


Figure 4.5 Block Diagram of Model Training Process

Following the training process, the models were saved for future use. The were saved in the Hierarchical Data Format 5 (HDF5) format, commonly denoted by the `.h5` extension. This format is a popular choice for deep learning models due to its ability to store not only the model architecture but also the trained weights and optimizer state

To ensure optimal performance, a checkpointing mechanism was employed during training using the 'ModelCheckpoint callback'. This technique periodically evaluates the model's performance on the validation set and saves a copy of the model with the best validation accuracy achieved so far. This allows for resuming training from the best performing model if the training process is interrupted or if hyperparameter tuning necessitates restarting training.

Given the binary classification nature of the problem, the Binary Cross-Entropy (BCE) loss function was employed. But after different experimentation since the validation data is not augmented, we chose to use Binary Focal Crossentropy. It is a modified version of binary crossentropy that introduces a focusing parameter to down-weight easy examples and put more focus on hard-to-classify examples. It is particularly useful in scenarios with class imbalance, as it helps the model focus more on the minority class or hard-to-classify examples. The formula for binary focal crossentropy is given on eq. (4.3).

$$BFCL = -\frac{1}{N}\sum_{i=1}^{N}\alpha(1-p_i)^{\gamma}\left[y_i\log(p_i)+(1-y_i)\log(1-p_i)\right] \qquad \textit{eq. (4.3)}$$

where: *BFCL* is Binary Focal Crossentropy Loss, α is a balancing factor to adjust the importance of positive/negative examples. γ is the focusing parameter that adjusts the rate at which easy examples are down-weighted. A higher value of γ increases the focus on hard examples.

4.4 Model Interpretation with GRAD-CAM

After saving our models and testing them, we used Grad-CAM to visualize important regions in our 3D CNN models for predicting specific classes. This technique analyzes gradients from the final convolutional layer to create heatmaps highlighting influential areas. Since deeper layers capture higher-level features while preserving spatial information, they're suitable for Grad-CAM. We identified the last convolutional layer using the model summary and leveraged Grad-CAM's class-discriminative nature to generate separate visualizations for each class. The heatmap is created by calculating the weighted sum of the feature map channels based on their importance for the predicted class. Finally, normalization (0-1) is applied for visualization purposes. The code snippet is available on Appendix E.

CHAPTER FIVE

RESULTS AND DISCUSSION

5.1 Introduction

This section details into the results obtained during the training and evaluation of our four 3D CNN models. We leverage the performance metrics discussed in Chapter 3 to assess their effectiveness. Following the performance evaluation, we present Grad-CAM visualizations, a technique employed to interpret the models' decision-making processes for each model.

5.2 Results

5.2.1 Training History

Training History of Model-1 / Baseline Model:

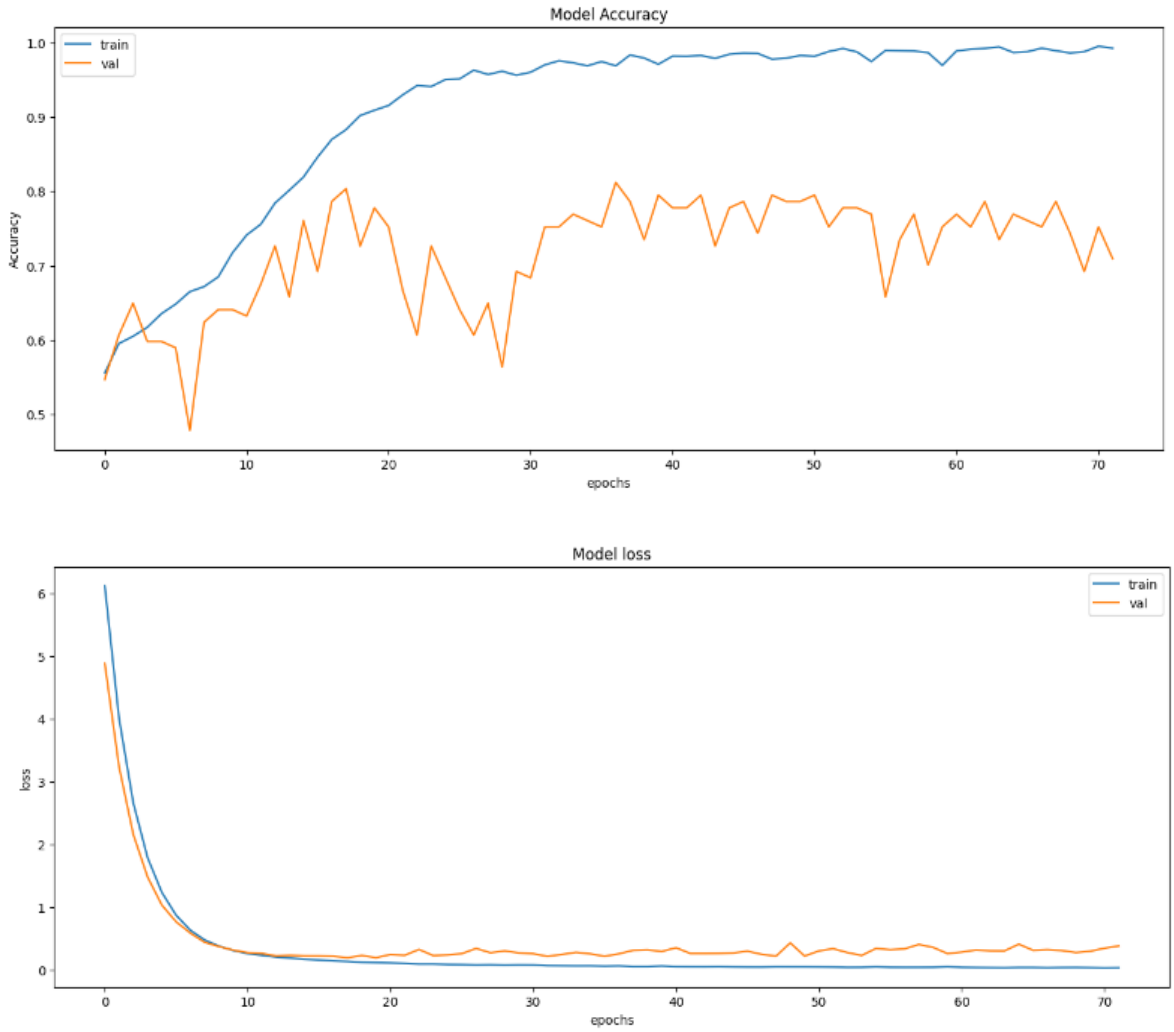


Figure 5.1 Baseline Model Accuracy and Loss During Training

During training, we got a maximum value validation accuracy of 81.19% and at that epoch we got training (accuracy = 99.23%, precision = 99.25%, recall = 99.33%, AUC = 99.38% and F1-score = 99.29%) and validation (accuracy = 81.19%, precision = 80.64%, recall = 94.93%, AUC = 74.76% and F1-score = 87.20%).

Training History of Model-2 / 3D AlexNet Model:

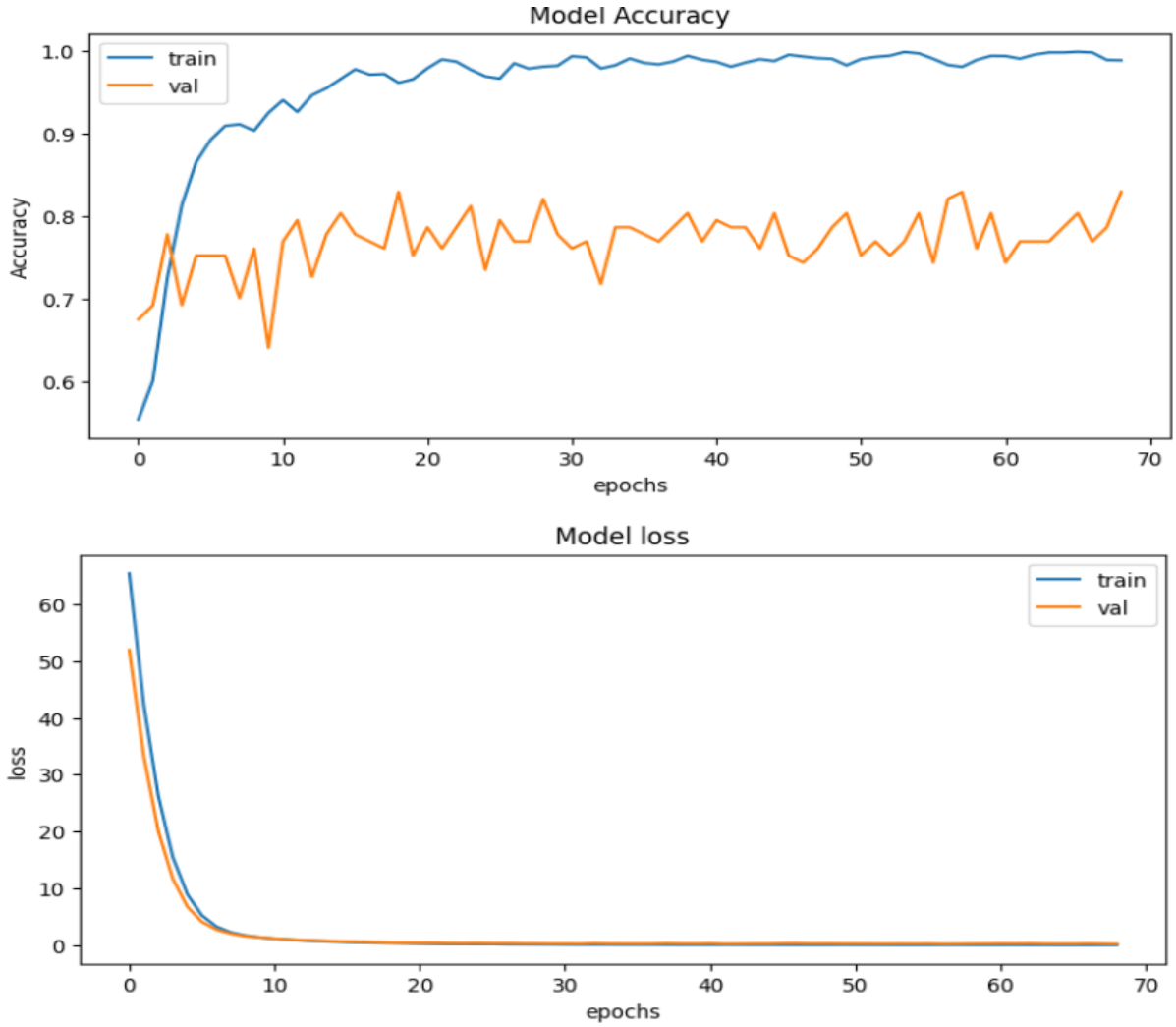


Figure 5.2 3D AlexNet Model Accuracy and Loss During Training

During training the 3D AlexNet model we got a maximum value validation accuracy of 82.90% and at that epoch we got training (accuracy = 98.78%, precision = 98.70%, recall = 99.05%, AUC = 99.91% and F1-score = 98.88%) and validation (accuracy = 82.90%, precision = 84.70%, recall = 91.13%, AUC = 84.42% and F1-score = 87.80%).

Training History of Model-3 / Proposed Model:

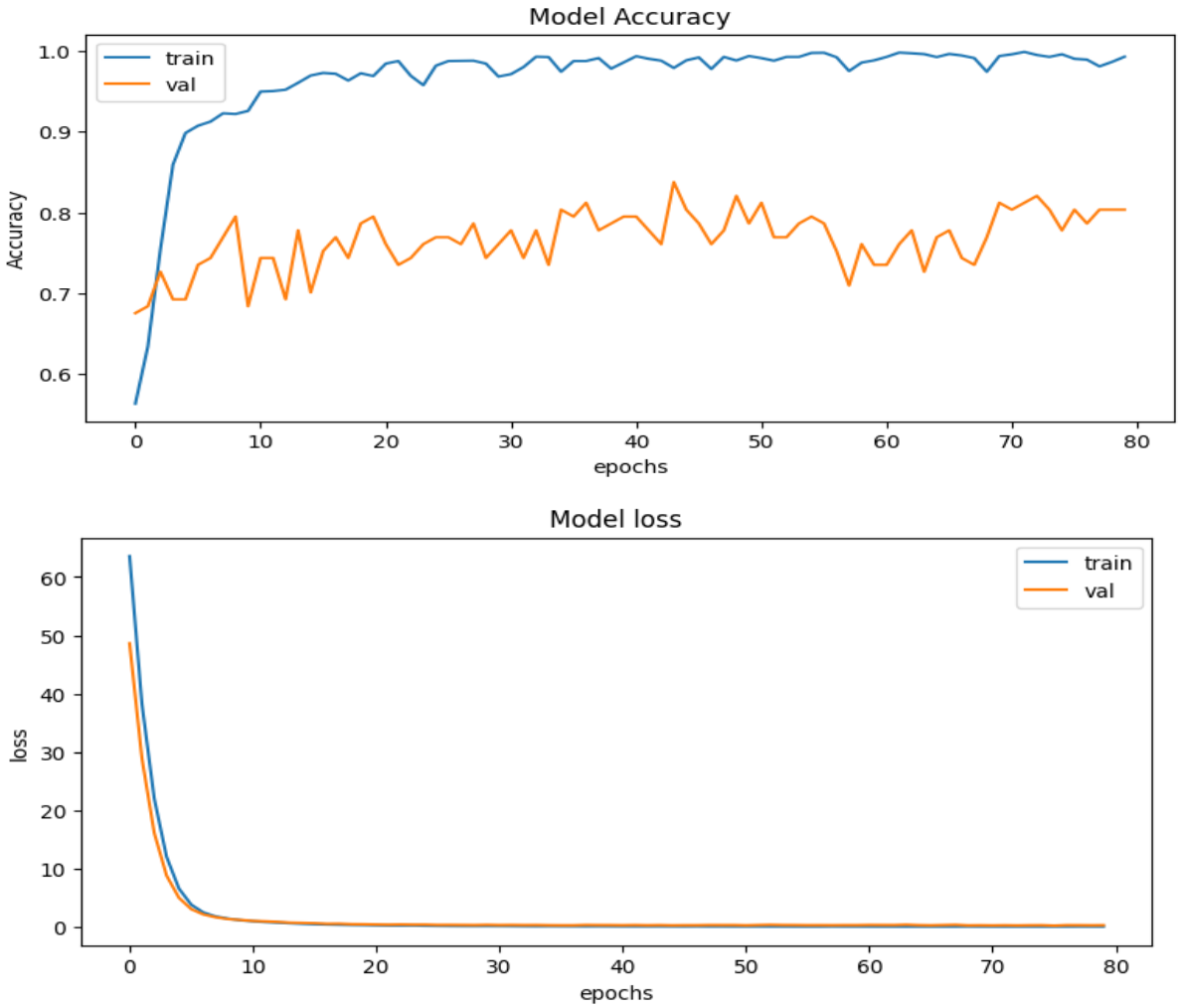


Figure 5.3 Proposed Model Accuracy and Loss During Training

During training Model-2 / Proposed model, we got a maximum value validation accuracy of 83.76% and at that epoch we got training (accuracy = 97.07%, precision = 97.58%, recall = 97.01%, AUC = 99.64% and F1-score = 97.92%) and validation (accuracy = 83.76%, precision = 82.61%, recall = 96.20%, AUC = 82.15% and F1-score = 88.79%).

Training History of Model-4 / Proposed Model with CBAM Model:

During training the Proposed 3D-CNN with CBAM Model, we got a maximum value validation accuracy of 88.03% and at that epoch we got training (accuracy = 97.95%, precision = 97.49%, recall = 98.84%, AUC = 99.75% and F1-score = 98.16%) and validation (accuracy = 88.03%, precision = 88.24%, recall = 94.94%, AUC = 90.04% and F1-score = 91.46%).

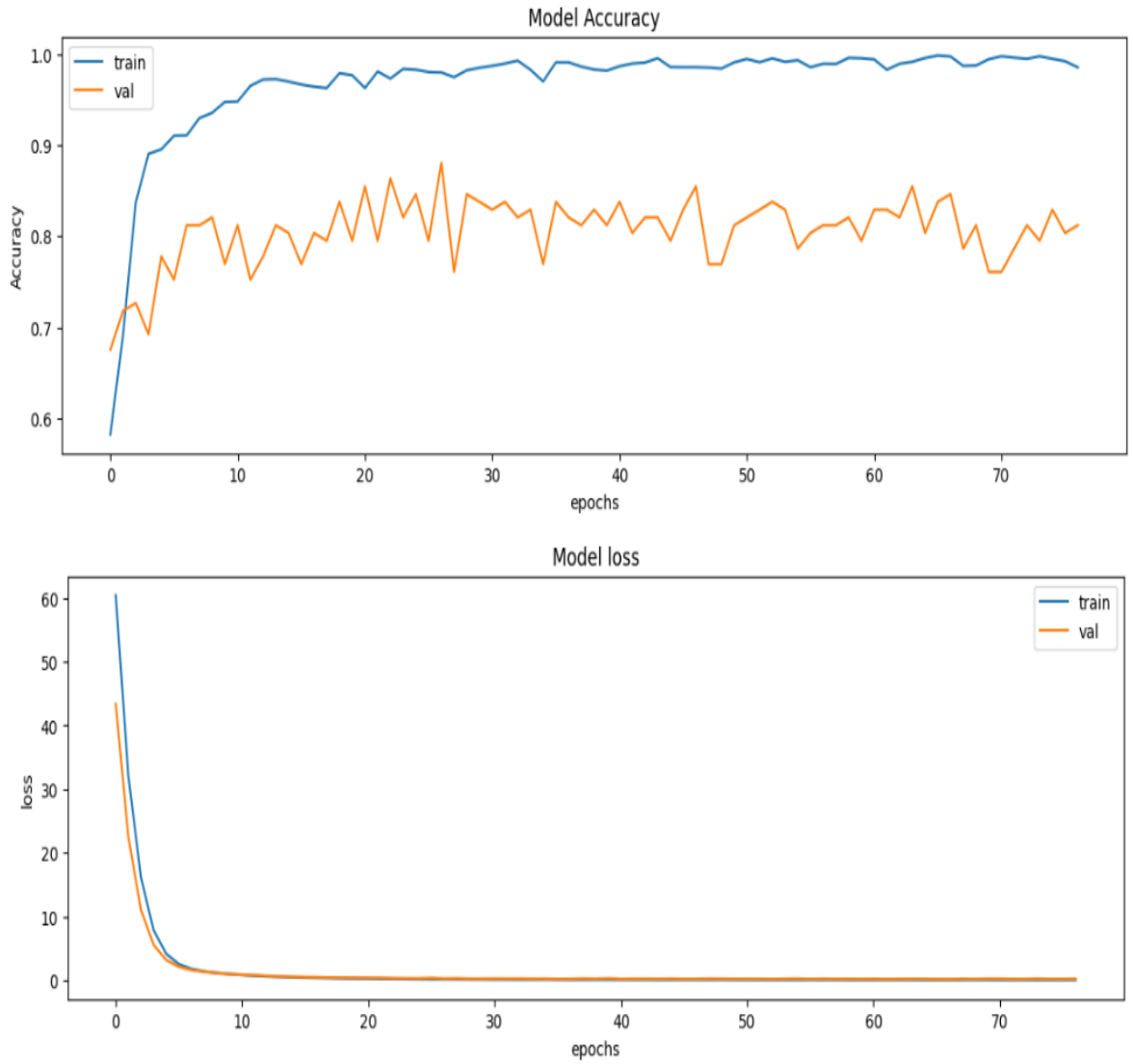


Figure 5.4 Proposed 3D-CNN with CBAM Model Accuracy and Loss during training

5.2.2 Testing Results

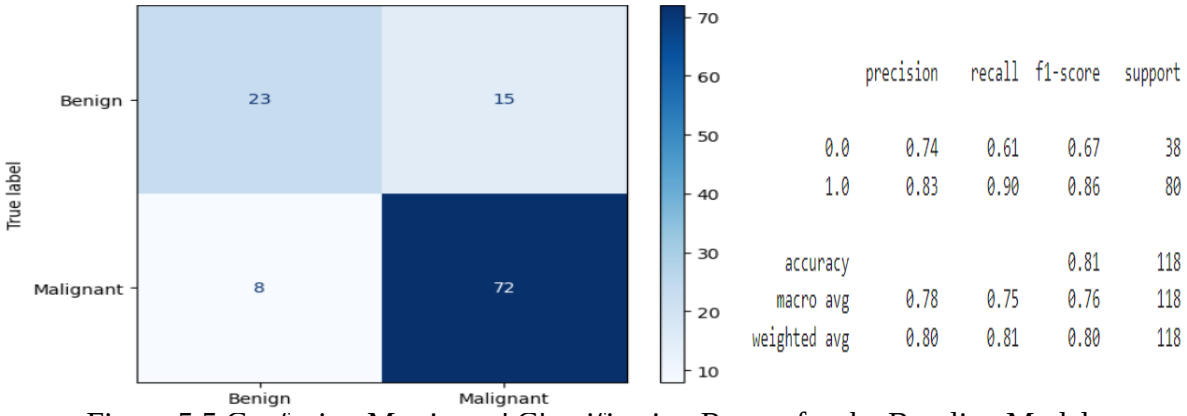
These are results obtained during model testing using the test dataset which is 10% of the whole dataset. The testing set has the remaining 38 test cases of benign class and 80 test cases of malignant class.

Table 5.1 Results of Performance Metrices for Testing Set

Models	**Performance Metrix**		
	Accuracy (%)	AUC (%)	F1-Score (%)
Baseline 3D CNN Model	80.50	85.26	86.75
3D ALEXNET	83.05	79.76	88.78
Proposed 3D CNN Model	88.98	97.91	91.50
Proposed 3D CNN Model with CBAM	**94.06**	**98.84**	**95.56**

Confusion Matrices and Classification Report for all the Four Models:

For Model-1 / Baseline

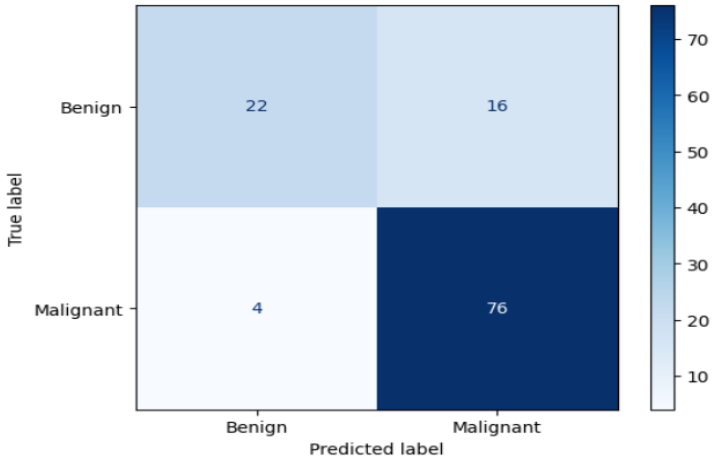


	precision	recall	f1-score	support
0.0	0.74	0.61	0.67	38
1.0	0.83	0.90	0.86	80
accuracy			0.81	118
macro avg	0.78	0.75	0.76	118
weighted avg	0.80	0.81	0.80	118

Figure 5.5 Confusion Matrix and Classification Report for the Baseline Model

The confusion matrix provided in figure 5.5 shows the performance of Model-1 in classifying lung nodules as benign or malignant. True Positives: Malignant nodules correctly classified as malignant. True Negatives: Benign nodules correctly classified as benign.
False Positives: Benign nodules incorrectly classified as malignant. And False Negatives: Malignant nodules incorrectly classified as benign.
The model correctly predicted 72 malignant cases and 23 benign cases. The model incorrectly predicted 15 benign cases as malignant and 8 malignant cases as benign.

For Model-2 / 3D AlexNet model

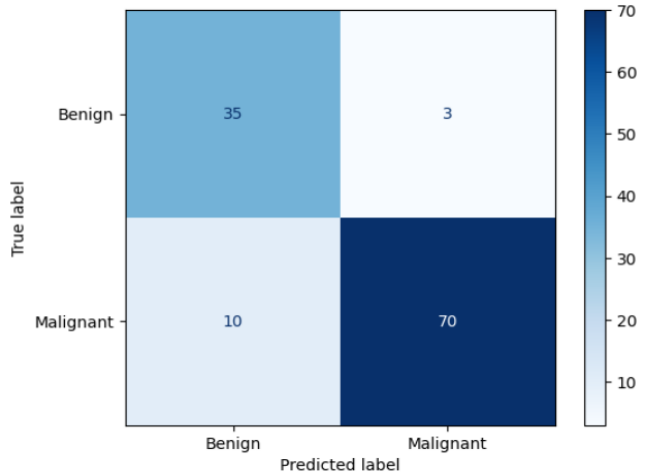


	precision	recall	f1-score	support
0.0	0.85	0.58	0.69	38
1.0	0.83	0.95	0.88	80
accuracy			0.83	118
macro avg	0.84	0.76	0.79	118
weighted avg	0.83	0.83	0.82	118

Figure 5.6 Confusion Matrix and Classification Report for the 3D AlexNet Model

The confusion matrix provided in figure 5.6 shows the performance of Model-2 in classifying lung nodules as benign or malignant. The model correctly predicted 76 malignant cases and 22 benign cases. The model incorrectly predicted 16 benign cases as malignant and 4 malignant cases as benign.

For Model-3 / Proposed Model

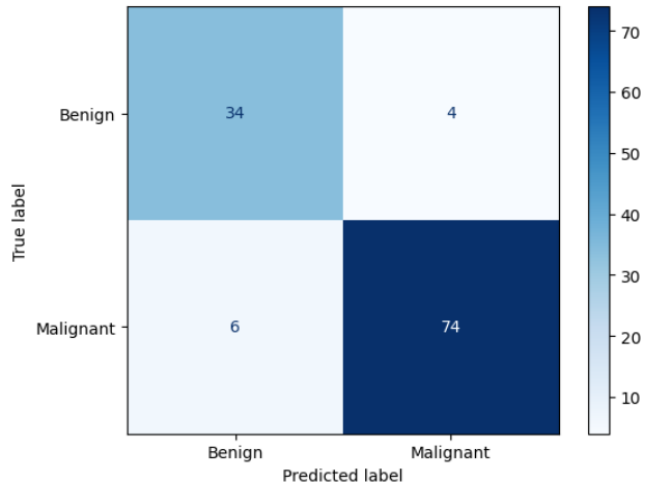


	precision	recall	f1-score	support
0.0	0.78	0.92	0.84	38
1.0	0.96	0.88	0.92	80
accuracy			0.89	118
macro avg	0.87	0.90	0.88	118
weighted avg	0.90	0.89	0.89	118

Figure 5.7 Confusion Matrix and Classification Report for Proposed Model

As we can see from the confusion matrix provided in figure 5.7, the model correctly predicted 70 malignant cases and 35 benign cases. The model incorrectly predicted 3 benign cases as malignant and 10 malignant cases as benign.

For Model-4 / Proposed Model with CBAM



	precision	recall	f1-score	support
0.0	0.85	0.89	0.87	38
1.0	0.95	0.93	0.94	80
accuracy			0.92	118
macro avg	0.90	0.91	0.90	118
weighted avg	0.92	0.92	0.92	118

Figure 5.8 Confusion Matrix and Classification Report for Proposed Model with CBAM

As we can see from the confusion matrix provided in figure 5.8, the model correctly predicted 74 malignant cases and 34 benign cases. The model incorrectly predicted 4 benign cases as malignant and 6 malignant cases as benign.

5.2.3 Interpretability Results

The interpretability results for all the three models are depicted on the next table. We showed the visual explanations using the heatmap generated by Grad-CAM for a single slice from the whole 64 slices.

Table 5.2 Visual Insights into the Baseline 3D CCN model's decision making

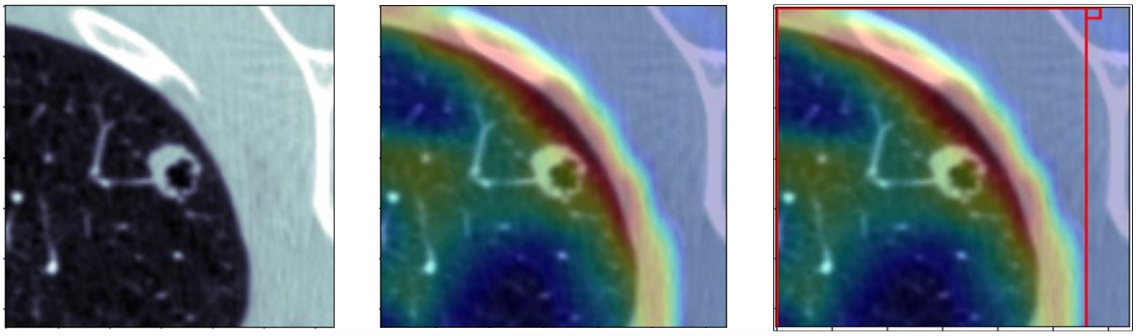
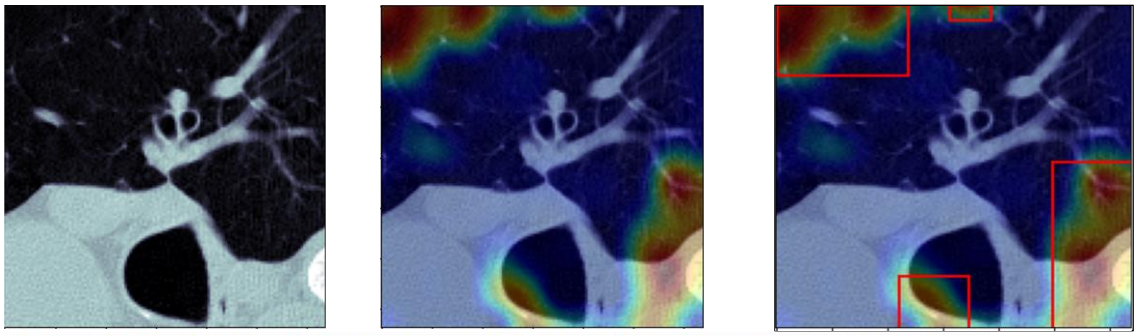
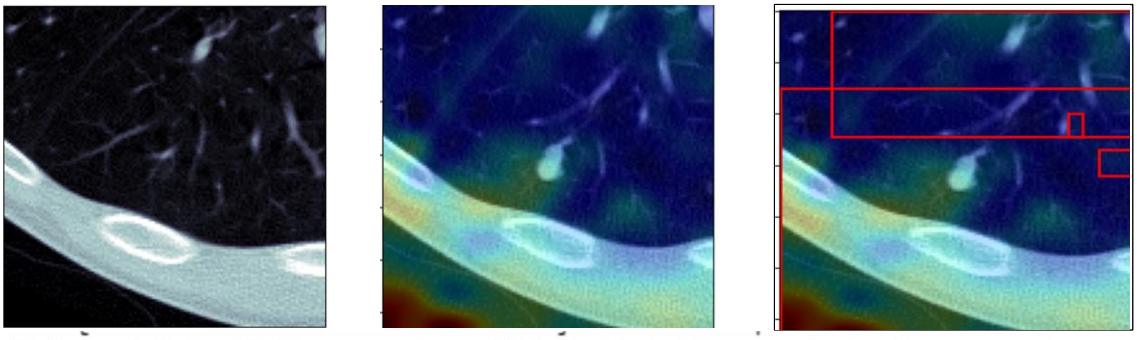
Original Slice	Generated Heatmap	Generated Heatmap with Bounding Box
 This model is 40.27 percent confident that CT scan is benign. This model is 59.73 percent confident that CT scan is malignant. True value = Malignant		
 This model is 95.72 percent confident that CT scan is benign. This model is 4.28 percent confident that CT scan is malignant. True value = Malignant		
 This model is 17.95 percent confident that CT scan is benign. This model is 82.05 percent confident that CT scan is malignant. True value = Malignant		

Table 5.3 Visual Insights into the 3D AlexNet model's decision making

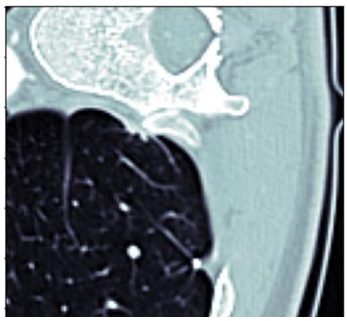
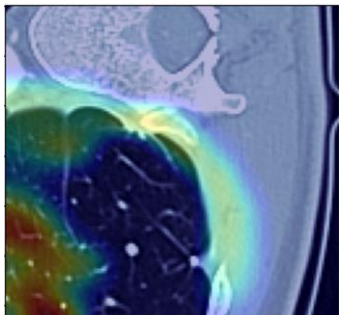
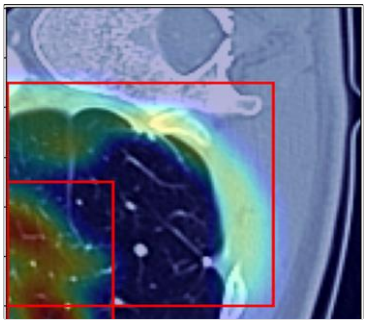
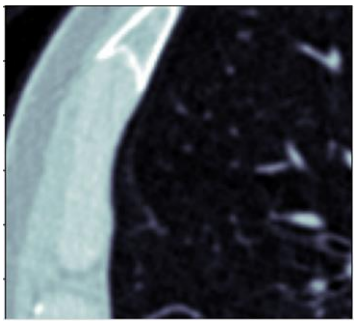
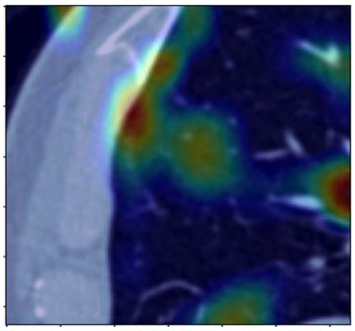
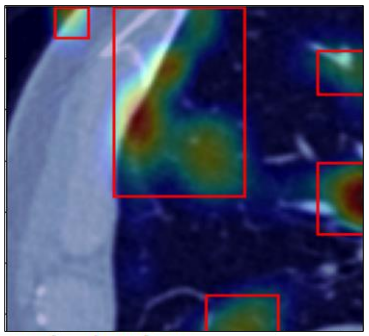
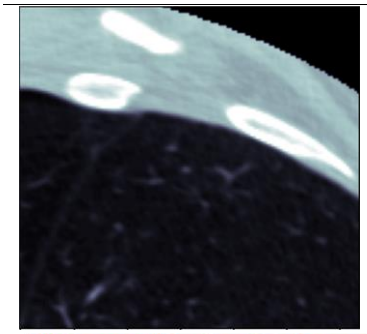
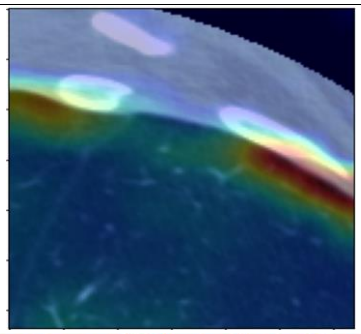
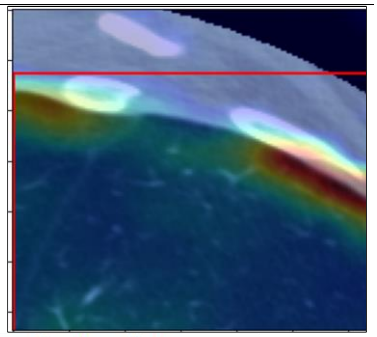
Original Slice	Generated Heatmap	Generated Heatmap with Bounding Box
		
This model is 38.51 percent confident that CT scan is benign. This model is 61.49 percent confident that CT scan is malignant. True value = Malignant		
		
This model is 49.34 percent confident that CT scan is benign. This model is 50.66 percent confident that CT scan is malignant. True value = Benign		
		
This model is 64.13 percent confident that CT scan is benign. This model is 35.87 percent confident that CT scan is malignant. True value = Malignant		

Table 5.4 Visual Insights into the Proposed model's decision making

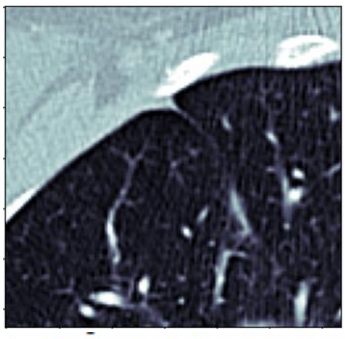
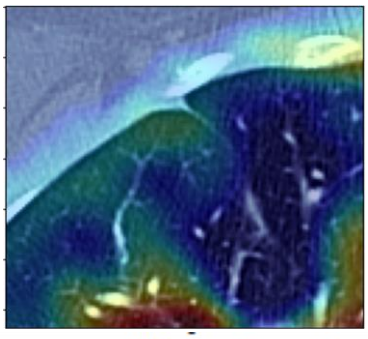
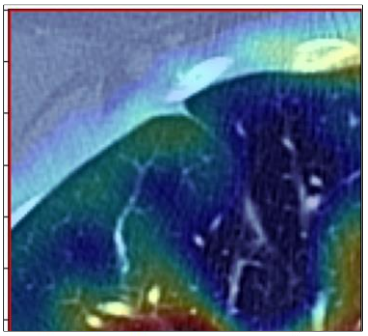
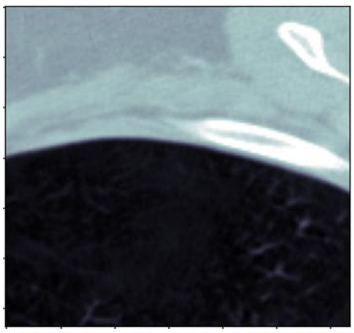
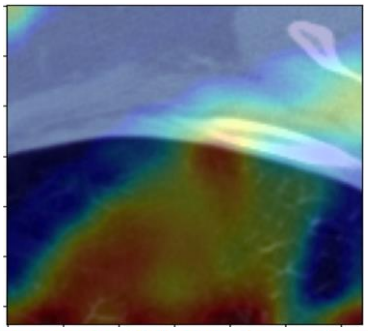
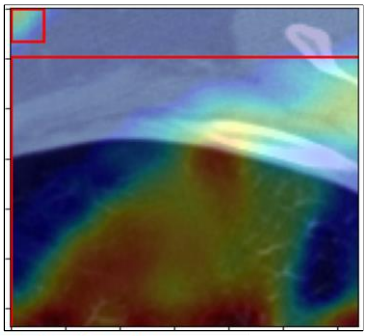
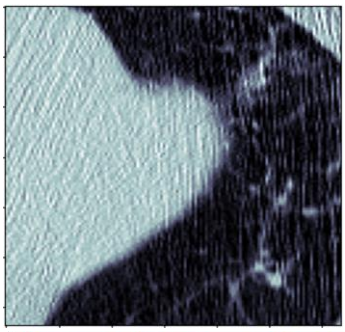
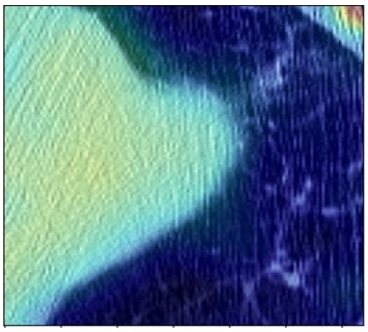
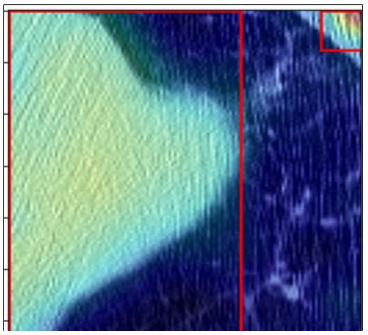
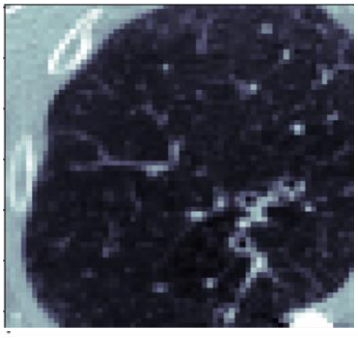
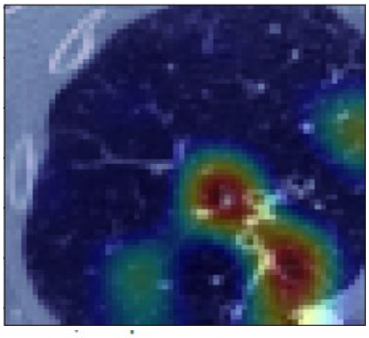
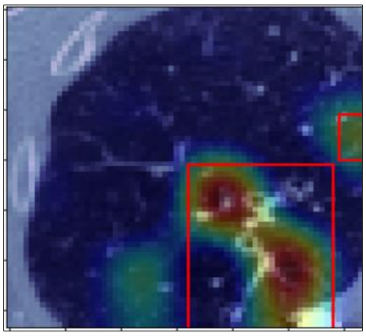
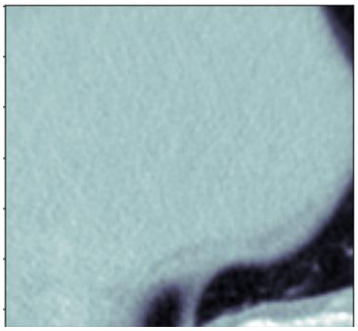
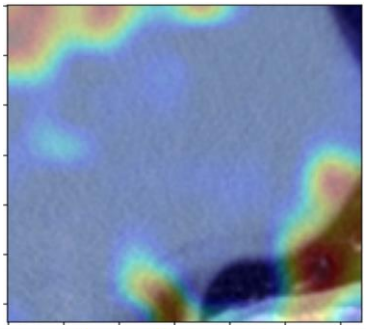
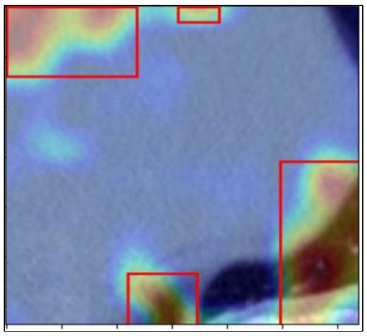
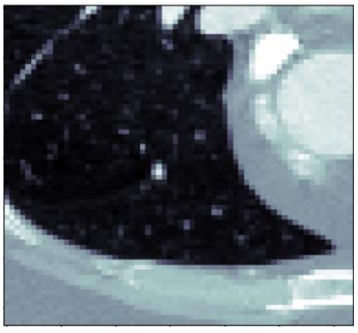
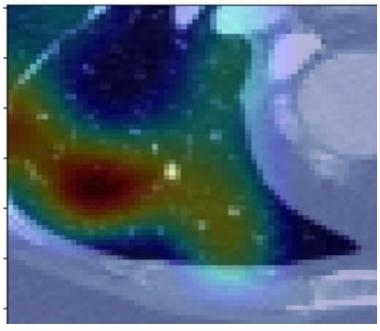
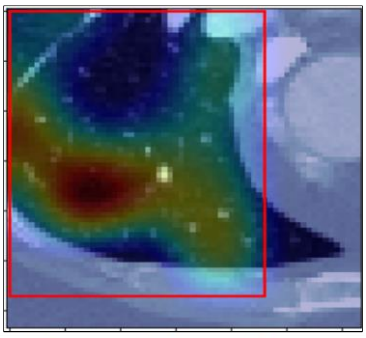
Original Slice	Generated Heatmap	Generated Heatmap with Bounding Box
		
This model is 50.21 percent confident that CT scan is benign. This model is 49.79 percent confident that CT scan is malignant. True value = Benign		
		
This model is 50.22 percent confident that CT scan is benign. This model is 49.78 percent confident that CT scan is malignant. True value = Benign		
		
This model is 7.39 percent confident that CT scan is benign. This model is 92.61 percent confident that CT scan is malignant. True value = Benign		

Table 5.5 Visual Insights into the Proposed model with CBAM model's decision making

Original Slice	Generated Heatmap	Generated Heatmap with Bounding Box
		
This model is 6.98 percent confident that the CT scan nodule patch is Benign. This model is 93.02 percent confident that the CT scan nodule patch is Malignant. The Radiologists determined the nodule as (true class) is: Malignant.		
		
This model is 40.11 percent confident that the CT scan nodule patch is Benign. This model is 59.89 percent confident that the CT scan nodule patch is Malignant. The Radiologists determined the nodule as (true class) is: Malignant.		
		
This model is 38.77 percent confident that the CT scan nodule patch is Benign. This model is 61.23 percent confident that the CT scan nodule patch is Malignant. The Radiologists determined the nodule as (true class) is: Malignant.		

Lung tissue often appears darker on CT scans, with lung nodules presenting as white spots. Consequently, radiologists typically focus on these black regions containing white spots. Our models should ideally mimic this approach, concentrating solely on lung areas. However, our visualizations revealed instances where certain models, like Model 4, were drawn to non-lung regions, even when the classification was accurate. Figure 5.9 demonstrates a sample test lung nodule CT-scan image, illustrating how the model might inadvertently attend to areas outside the lung regions.

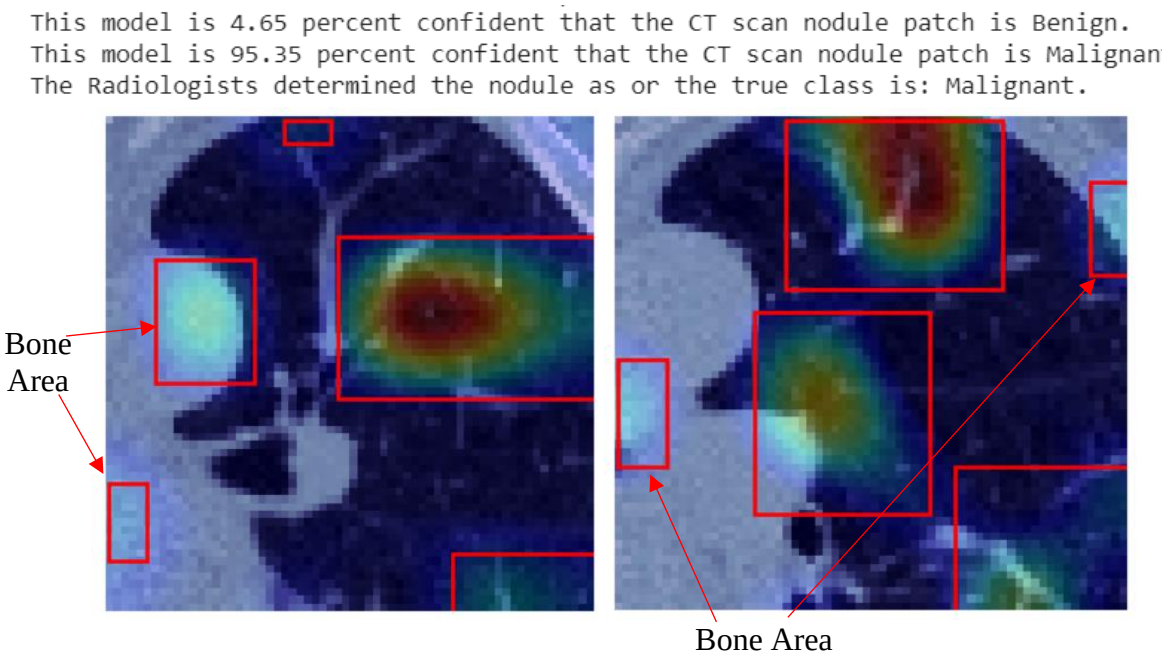


Figure 5.9 Sample Model-4 Result Interpretability Evaluation by the Radiologists

Based on expert radiologist feedback, interpretability visualizations should ideally present all three planes of the 3D CT-scan image: axial, sagittal, and coronal. This aligns with how radiologists typically view and analyze CT scans using medical imaging software. To provide a comprehensive understanding of the model's focus, we created 3D visualizations for all three planes, as depicted in Figure 5.10.

The 3D visualization encompassing all three planes (axial, sagittal, and coronal) was overwhelmingly preferred by experts over single-plane or solely axial views. This comprehensive approach provided a more holistic understanding of the model's focus, aiding in the interpretation of its predictions. By presenting the model's attention across different perspectives, clinicians were better equipped to assess the relevance and accuracy of the generated heatmaps, ultimately enhancing the interpretability and trustworthiness of the model's outputs.

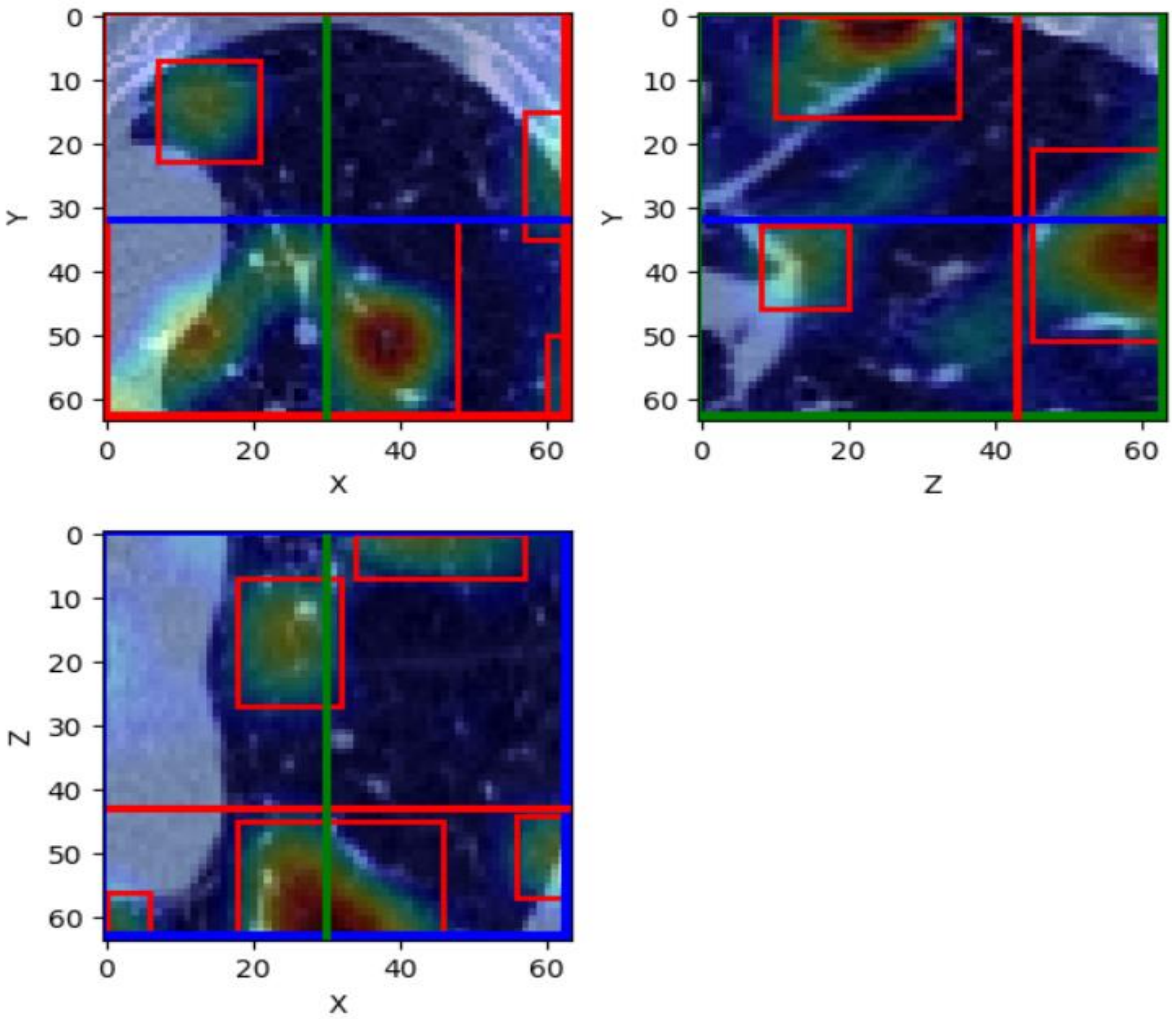


Figure 5.10 Sample 3D Interpretability Visualization Based on the Experts Suggestions

5.3 Discussions

5.3.1 Results Discussion

The training history suggests that the initial 3D CNN models achieved effective classification of lung nodules into benign and malignant categories. However, a performance decrease on the test and validation sets indicated potential overfitting. To mitigate this, dropout layers and L2 regularization techniques were incorporated in each model with dropout rates ranging from 60% to 80%. The validation accuracy peaked at a dropout rate of 70%, demonstrating its effectiveness in reducing overfitting.

Furthermore, an investigation into various learning algorithms and optimizers was conducted. The Adam optimizer consistently yielded the best results across all 3D CNN models. This suggests that Adam's adaptive learning rate capabilities were well-suited for the optimization process within the context of this task.

Due to the inherent memory demands of 3D data and 3D CNN architectures, experimenting with larger batch sizes was not feasible. Consequently, a batch size of 16 was employed for all the four models. These configurations ensured successful training without encountering resource exhaustion or out-of-memory errors.

These modifications, including dropout layer implementation and Adam optimizer selection, demonstrably addressed the overfitting issue and yielded improved performance on the validation set. Additionally, the chosen batch sizes ensured efficient training while staying within GPU memory limitations. Now let's discuss the results of each model.

The LUNA22 ISMI dataset is an imbalanced dataset it has almost two times more nodules that are malignant than the benign ones. Even though we tried augmentation and make the dataset balanced; the models have learnt to distinguish malignant class nodules more accurately than the benign ones. We investigated this more by calculating specificity of the models for the training, testing and validation sets.

Specificity: Specificity complements recall by evaluating the model's ability to correctly classify negative cases (benign nodules). It is calculated as the proportion of true negatives divided by the total number of negative samples present in the testing data.

$$Specificity = TN/(FP + TN) \qquad eq.\ (5.1)$$

When we calculate the specificity using equation 5.1 for the test set, we found for the baseline model is 74.19%, for the AlexNet model is 84.61%, for the proposed model is 77.78%, and for the proposed model with CBAM is 85%. The Proposed model with CBAM achieved a higher training and testing accuracy, its specificity (ability to correctly identify benign nodules) is high also.

The current study's findings, while demonstrating the model's capacity for lung nodule classification, do not necessarily represent the state-of-the-art performance achievable in this domain. As acknowledged in the literature review, other studies employing different datasets have reported superior classification results.

A crucial limitation of this research is the lack of a publicly available leaderboard for benchmarking performance on the specific dataset utilized. The absence of established benchmarks hinders a direct comparison of our model's accuracy and generalizability to existing solutions.

The convolutional layers on the fourth model extract relevant features from the CT scan images. These features might include patterns associated with lung cancer nodules, such as changes in tissue density, texture, or shape.

Attention Mechanisms**;** the CBAM module plays a crucial role in enhancing feature learning for lung cancer nodule detection: Channel Attention; By focusing on the most informative channels, the channel attention mechanism helps the model identify the specific features within the CT scan images that are most indicative of lung cancer nodules. This can improve the model's sensitivity to subtle changes in tissue characteristics.

Spatial Attention: By highlighting the critical spatial locations within the CT scan images, the spatial attention mechanism enables the model to pinpoint the exact regions where lung cancer nodules are likely to be present. This can improve the model's localization accuracy and reduce the number of false positives.

The Improved Feature Learning; CBAM helps the model to focus on the most discriminative features within the CT scan images, leading to more accurate and robust lung cancer nodule Classification. Enhanced Localization: By identifying the precise spatial locations of lung cancer nodules, CBAM can improve the model's ability to localize these nodules within the CT scan images.

By suppressing irrelevant features and focusing on the most informative regions, CBAM can help to reduce the number of false positive detections, improving the model's specificity (Reduced False Positives).

CBAM can help the model to learn more generalizable features, making it less prone to overfitting and improving its performance on unseen CT scan images (Improved Generalization).

Overall, the 3D CNN model with CBAM is well-suited for lung cancer nodule classification from CT scan images. By effectively leveraging the attention mechanisms and focusing on the most relevant features and spatial locations, the model can achieve high accuracy and sensitivity in detecting lung cancer nodules.

In our implementation, Grad-CAM was applied to individual slices within CT scan patches. These patches, containing 64 axial slices of nodule images, represent the model's input data. By generating heatmaps for each slice, Grad-CAM visualizes the areas that the model attends

to most when classifying a specific nodule patch as malignant or benign. These interpretable heatmaps provide valuable information for clinicians and radiologists. By visualizing the model's focus areas within each slice, these heatmaps can potentially:

Enhance Decision Support; Offer insights that complement the radiologist's expertise, potentially aiding in the final classification of a nodule as malignant or benign.

Improve Explainability; Increase the transparency of the model's predictions, allowing clinicians to better understand the rationale behind the model's classification decisions.

Identify Potential Biases: The heatmaps may reveal any potential biases within the model, such as an over-reliance on specific image artifacts or features that might not be clinically relevant. While Grad-CAM offers valuable interpretability, it's crucial to acknowledge its limitations. Grad-CAM primarily focuses on localized regions contributing to the classification and might not capture the model's higher-level reasoning processes (Chattopadhyay et al., 2017; Selvaraju et al., 2017). We were capable of exploring more advanced interpretability techniques that delve deeper into the model's internal workings and feature interactions.

We chose to integrate CBAM into the best of the three models; which was Model-3. Grad-CAM (Gradient-weighted Class Activation Maps) and CBAM (Convolutional Block Attention Module) are powerful techniques used to improve the interpretability of deep learning models. Grad-CAM generates class-specific heatmaps that highlight the most important regions in an input image that contribute to a given prediction. CBAM, on the other hand, dynamically adjusts feature responses by learning adaptive attention weights (Woo et al., n.d.). Combining Grad-CAM and CBAM can provide a deeper understanding of how the model makes decisions. The input image is passed through the model's layers, including the CBAM module. CBAM dynamically adjusts the feature responses based on their importance. The target class's gradient is computed and propagated backward through the model. The gradients are weighted by the activations of the final convolutional layer.

The weighted gradients are averaged to generate a class-specific heatmap. This heatmap highlights the regions in the input image that contributed most to the prediction.

The advantages of combining Grad-CAM and CBAM are Enhanced Interpretability; CBAM's attention mechanism can help Grad-CAM focus on the most relevant features, making the generated heatmaps more informative and easier to understand. Improved

Localization; The attention weights learned by CBAM can guide Grad-CAM to more accurately localize the important regions in the input image. Better Understanding of Model Behavior: By combining these two techniques, we can gain a deeper understanding of how the model is making decisions and what features it is focusing on.

Additionally, integrating these interpretable heatmaps into a user-friendly clinical interface could further enhance their utility in real-world medical practice. Since the interpretation is done for the 3D models, the users/the radiologists can go through the full series of the CT-scan nodules for deeper investigation. This is depicted on figure 5.9.

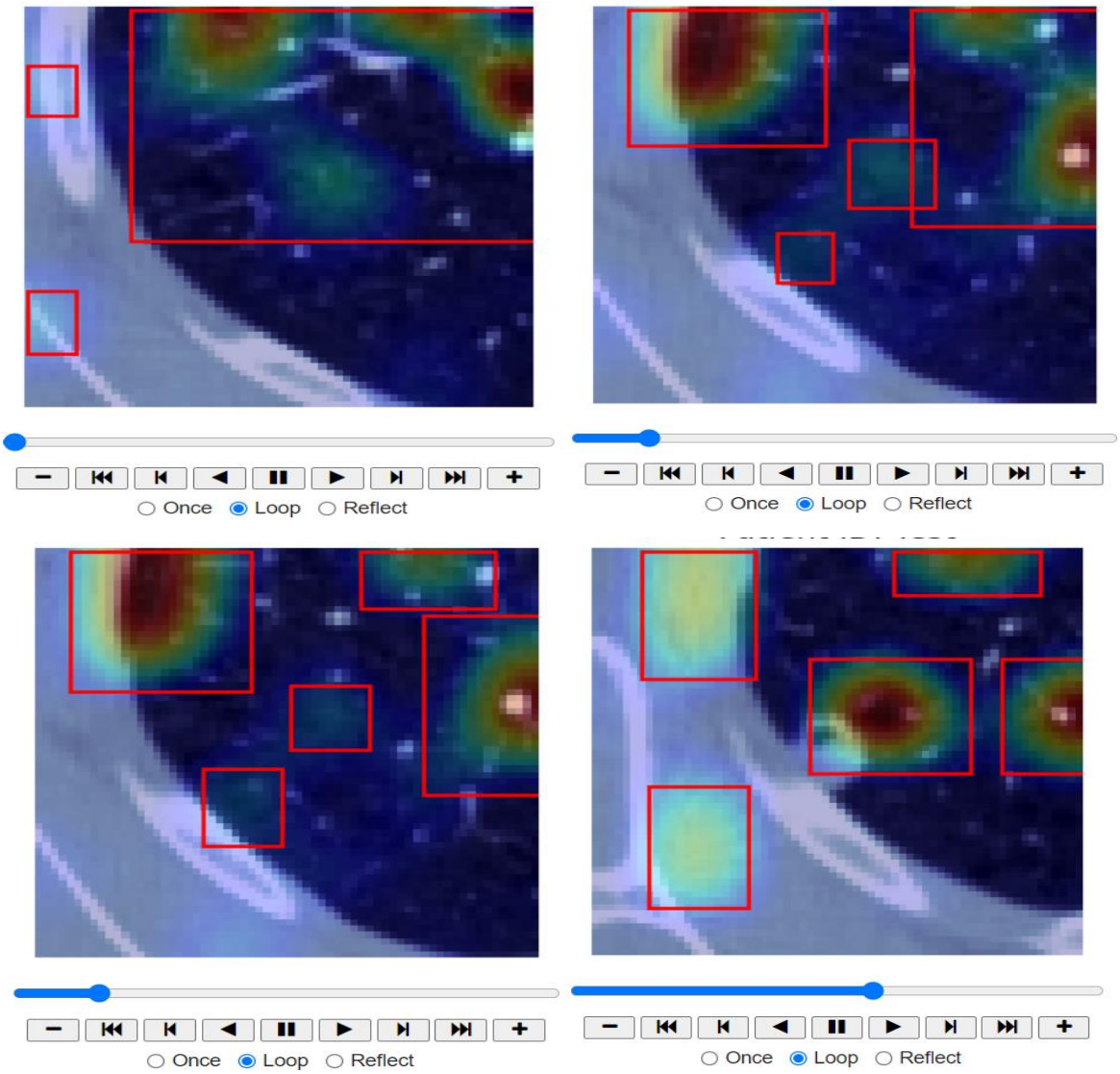


Figure 5.11 3D Grad-CAM Visualizations for the axial slices

5.3.2 Research Question Discussion

Answer for RQ1: This study digs into the exploration of various techniques and tools for utilizing the rich information captured by chest CT scans. A key focus was placed on harnessing the inherent 3D nature of this data modality. Unlike simpler image formats, CT scans provide volumetric information crucial for tasks like lung nodule classification. By employing 3D data as input, we were able to leverage the spatial relationships and contextual information existing between voxels (3D volume elements) within the CT scans. This 3D information potentially holds valuable features for differentiating malignant from benign lung nodules.

To exploit this 3D information effectively, we opted for the development of 3D convolutional neural networks (CNNs). 3D CNNs are specifically designed to process volumetric data, allowing them to learn features directly from the 3D CT scans. These learned features can then be utilized for various applications, such as the classification of lung nodules in our case. The 3D CNN approach offers several advantages compared to traditional methods that might rely on pre-processing steps to convert 3D CT scans into simpler 2D representations. These pre-processing steps can potentially lead to information loss and hinder the model's ability to capture the full spectrum of features present in the 3D data. By directly using 3D CNNs, we can:

Preserve Spatial Relationships and maintain the spatial information between voxels within the CT scan, potentially leading to more robust feature extraction for classification. Learn 3D Features, the 3D CNN architecture allows the model to learn features directly from the 3D data, potentially capturing intricate or detailed relationships between anatomical structures within the lung. Based on our experiments, our proposed model with CBAM performed better than the other models in classifying the lung nodules. CBAM helped the model focus on the most relevant features, leading it to better classification performance.

Answer for RQ2: The field of XAI encompasses a diverse set of techniques categorized based on various criteria. One prominent categorization focuses on the level of explanation provided:
Model-Agnostic Methods: These techniques are applicable to any AI model, including "black-box" models where the internal workings are not readily interpretable. They leverage the input data and the model's output to generate explanations.

Examples include SHAP (SHapley Additive exPlanations) and LIME (Local Interpretable Model-agnostic Explanations).

Model-Specific Methods: These techniques are tailored to explain the inner workings of specific model architectures. They exploit the structure and characteristics of the model itself to provide explanations. Examples include Grad-CAM (Gradient-weighted Class Activation Mapping) for convolutional neural networks (CNNs) and feature importance analysis for various models.

This study explores the application of XAI techniques to interpret the results of our developed lung cancer detection model. By understanding the model's reasoning, we aim to improve its transparency and facilitate its integration into clinical workflows.

In our research, we utilize Grad-CAM, which falls under the category of model-specific methods. Grad-CAM is particularly well-suited for interpreting CNNs, which are extensively used in medical image analysis tasks like lung nodule classification. By applying Grad-CAM to our 3D CNN model, we were able to generate heatmaps that visualize the image regions the model focuses on when classifying a specific nodule patch as malignant or benign.

Integrating CBAM into our 3D CNN model can enhance the interpretability of the generated heatmaps. By focusing the model's attention on the most informative regions of the CT scans, CBAM can help Grad-CAM identify more relevant areas for classification. This can lead to more accurate and meaningful explanations.

These interpretable heatmaps offer valuable insights for healthcare professionals, specifically radiologists. They provide visual cues into the model's rationale behind its classification decisions, potentially leading to several benefits:

Complementing Clinical Expertise: The heatmaps can highlight the image regions the model deems most significant for classification. This visual information can complement a radiologist's expertise, potentially aiding in the final diagnosis of a nodule.

Improved Decision Support: By understanding the model's focus areas, clinicians might gain additional insights to support their own decision-making processes.

Enhanced Explainability: The heatmaps foster transparency by enabling clinicians to comprehend the model's reasoning behind its predictions. This can build trust in the model and encourage its integration into clinical workflows.

Identifying Potential Biases: In some cases, the heatmaps might reveal any potential biases within the model. For instance, an over-reliance on specific image artifacts that might not be clinically relevant could be identified and addressed. By including XAI techniques like Grad-CAM, we can bridge the gap between AI models and human healthcare professionals. These fosters trust in AI applications within the medical domain and empowers clinicians to leverage these models effectively in their work.

By incorporating interpretability and explainability techniques, researchers are developing deep learning models that are more trustworthy and can be effectively integrated into real-world applications across various domains. As we have discussed on the previous parts; There's a trade-off between interpretability and accuracy. Highly interpretable models might be less accurate than complex black-box models. The choice of XAI technique depends on the specific model architecture, task, and desired level of explanation. Combining multiple XAI techniques can sometimes be beneficial for a more comprehensive understanding of the model's behavior.

CHAPTER SIX

CONCLUSSIONS AND FUTURE WORK

6.1 Conclusion

This study investigated the potential of 3D convolutional neural networks (CNNs) for lung nodule classification. We employed a 3D CNN architecture to extract features directly from the volumetric CT scan data, aiming to achieve robust classification performance for differentiating malignant from benign nodules.

To gain insights into the decision-making processes of the 3D CNN model, we incorporated Grad-CAM (Gradient-weighted Class Activation Mapping) for interpretability analysis. Grad-CAM generates heatmaps that visualize the image regions within a CT scan slice that contribute most significantly to the model's classification decision for a specific nodule patch. This visual information can potentially be used by healthcare professionals to understand the model's rationale behind its predictions.

To mitigate the impact of class imbalance and enhance model performance, we employed data augmentation and fine-tuning techniques. Although further improvement is needed our proposed 3D CNN model achieved accuracies of 94.06% and 88.06% for the testing and validation sets respectively.

It's crucial to acknowledge the inherent challenges associated with medical image analysis. Medical images, like CT scans, are often fine-grained and exhibit subtle variations that can be difficult for models to distinguish, particularly between benign and malignant nodules. Additionally, the 3D nature of CT scans adds complexity compared to simpler 2D image formats, potentially requiring more sophisticated models and larger datasets for optimal performance.

The proposed 3D CNN model with CBAM achieved an AUC of 98.84, and F1-score of 95.56 on the test set. These metrics suggest that the model exhibits promising potential for accurate lung nodule classification. Notably, the high recall value (94.99) indicates a low rate of missed malignant nodules, which is crucial in clinical settings. The test set performance, demonstrates a good degree of generalizability. The consistency between validation and test set performance suggests that the model is not overfitting to the training data and can potentially perform well on unseen data.

While the evaluation metrics indicate encouraging performance, it's important to acknowledge potential limitations. A larger dataset could potentially lead to further improvement in generalizability. Additionally, comparing these metrics with results from alternative models or established benchmarks would provide a more comprehensive perspective on the 3D CNN model's relative strengths and weaknesses.

6.2 Future Work

Building upon the promising results achieved by the 3D CNN models, future research efforts can explore several avenues to further enhance its performance and generalizability. A deeper investigation into the generated class activation maps (CAMs) using Grad-CAM is very helpful for explainability. By carefully analyzing these maps with the assistance of radiologists, we can gain valuable insights into the model's focus areas when classifying lung nodules. Identifying the image regions the model prioritizes for decision-making can reveal potential strengths and weaknesses. This knowledge can then be leveraged to refine the network architecture or training process to improve classification accuracy.

Leveraging Local Datasets and Expert Knowledge: The availability of a locally collected dataset obtained with the collaboration of healthcare professionals presents a valuable opportunity. By strategically integrating this local dataset with publicly available datasets, we can create a more diversified and robust data pool for model training. This diversification can enhance the model's ability to generalize to unseen data and potentially improve its performance across various nodule types and appearances.

Addressing Class Imbalance: As previously acknowledged, class imbalance in the training data can hinder the model's ability to learn features effectively from the under-represented class (benign nodules). By balancing the class distribution within the training data, we can potentially improve the model's ability to accurately classify both malignant and benign nodules.

The field of deep learning is constantly evolving, with new and improved architectures emerging. Investigating alternative 3D CNN architectures specifically designed for medical image analysis could potentially lead to superior feature extraction capabilities for lung nodule classification. Additionally, incorporating attention mechanisms within the network architecture might allow the model to focus on more discriminative regions within the CT scans, further enhancing classification accuracy.

Close collaboration with radiologists throughout the research process is crucial. Their expertise can be invaluable in, interpreting Class Activation Maps: Radiologists' insights can aid in interpreting the generated CAMs, helping to identify the image features the model prioritizes for classification and assess their clinical relevance.

Validating Model Performance: Involving radiologists in the evaluation process can provide valuable feedback on the model's performance in a real-world clinical setting.

Guiding Future Research Directions: Collaboration with healthcare professionals can help identify areas where the model can be further optimized to improve its clinical utility.

By implementing these strategies and fostering collaboration with healthcare professionals, we can strive towards developing a highly accurate and generalizable 3D CNN model for lung nodule classification, ultimately contributing to more effective diagnosis and patient care for the future.

REFERENCES

Alghamdi, H. S. (2022). Towards Explainable Deep Neural Networks for the Automatic Detection of Diabetic Retinopathy. *Applied Sciences (Switzerland)*, *12*(19). https://doi.org/10.3390/app12199435

Ali, S., Abuhmed, T., El-Sappagh, S., Muhammad, K., Alonso-Moral, J. M., Confalonieri, R., Guidotti, R., Del Ser, J., Díaz-Rodríguez, N., & Herrera, F. (2023). Explainable Artificial Intelligence (XAI): What we know and what is left to attain Trustworthy Artificial Intelligence. *Information Fusion*, *99*, 101805. https://doi.org/https://doi.org/10.1016/j.inffus.2023.101805

American College of Radiology. (2023). *Lung cancer screening (LCS) - LungRADS.* https://www.acr.org/Clinical-Resources/Reporting-and-Data-Systems/Lung-Rads

B. C., K., & K. B., N. (2023). An Approach of AlexNet CNN Algorithm Model for Lung Cancer Detection and Classification. *International Journal on Recent and Innovation Trends in Computing and Communication*, *11*(11s), 49–54. https://doi.org/10.17762/ijritcc.v11i11s.8069

Bach, S., Binder, A., Montavon, G., Klauschen, F., Müller, K. R., & Samek, W. (2015). On pixel-wise explanations for non-linear classifier decisions by layer-wise relevance propagation. *PLoS ONE*, *10*(7). https://doi.org/10.1371/journal.pone.0130140

Bao, L., Bao, T., Zheng, Y., & Xia, J. (2020, July 21). A Simple Residual Network for Lung Nodule Classification. *ACM International Conference Proceeding Series*. https://doi.org/10.1145/3403782.3403808

Barredo Arrieta, A., Díaz-Rodríguez, N., Del Ser, J., Bennetot, A., Tabik, S., Barbado, A., Garcia, S., Gil-Lopez, S., Molina, D., Benjamins, R., Chatila, R., & Herrera, F. (2020). Explainable Artificial Intelligence (XAI): Concepts, taxonomies, opportunities and challenges toward responsible AI. *Information Fusion*, *58*, 82–115. https://doi.org/https://doi.org/10.1016/j.inffus.2019.12.012

Brett, M., Markiewicz, C. J., Hanke, M., Côté, M.-A., Cipollini, B., McCarthy, P., Jarecka, D., Cheng, C. P., Larson, E., Halchenko, Y. O., Cottaar, M., Ghosh, S., Wassermann, D., Gerhard, S., Lee, G. R., Baratz, Z., Wang, H.-T., Papadopoulos Orfanos, D.,

Kastman, E., … freec84. (2024). *nipy/nibabel: 5.2.1*. Zenodo. https://doi.org/10.5281/zenodo.10714563

Bushara, A. R., & Kumar, R. S. V. (2022). Deep Learning-based Lung Cancer Classification of CT Images using Augmented Convolutional Neural Networks. *Electronic Letters on Computer Vision and Image Analysis*, *21*(1), 130–142. https://doi.org/10.5565/REV/ELCVIA.1490

Causey, J. L., Li, K., Chen, X., Dong, W., Walker, K., Qualls, J. A., Stubblefield, J., Moore, J. H., Guan, Y., & Huang, X. (2022). Spatial Pyramid Pooling With 3D Convolution Improves Lung Cancer Detection. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, *19*(2), 1165–1172. https://doi.org/10.1109/TCBB.2020.3027744

Chattopadhyay, A., Sarkar, A., Balasubramanian, V., Howlader, P., & Balasubramanian, V. N. (2017). *Grad-CAM++: Generalized Gradient-based Visual Explanations for Deep Convolutional Networks*. https://doi.org/10.48550/arXiv.1710.11063

Conor O'Sullivan. (n.d.). *XAI with Python*. A Dat Odyssey. Retrieved January 27, 2024, from adataodyssey.com

Dr. Rosebrock, A. (2017). *Deep Learning for Computer Vision with Python Starter Bundle 1st Edition (1.1.0)*.

Essaf, F., Li, Y., Sakho, S., & Gadosey, P. K. (2020). Improved Convolutional Neural Network for Lung Cancer Detection. *ACM International Conference Proceeding Series*, 48–54. https://doi.org/10.1145/3398329.3398337

Fchollet, & Team Keras. (2019). Keras documentation: The Functional API. In *Keras, Complete guide to the functional API.* https://keras.io/guides/functional_api/

Gebremariam, T. H., Haisch, D. A., Fernandes, H., Huluka, D. K., Binegdie, A. B., Woldegeorgis, M. A., Ergetie, W., Worku, A., Zerihun, L. M., Cohen, M., Massion, P. P., Sherman, C. B., Saqi, A., & Schluger, N. W. (2021). Clinical Characteristics and Molecular Profiles of Lung Cancer in Ethiopia. *JTO Clinical and Research Reports*, *2*(7). https://doi.org/10.1016/j.jtocrr.2021.100196

Giard, C. (2023). *Radiology Workflow | Efficiencies*. https://enlitic.com/blogs/radiology-workflow-efficiencies/

Gierada, D. S., Black, W. C., Chiles, C., Pinsky, P. F., & Yankelevitz, D. F. (2020). Low-dose ct screening for lung cancer: Evidence from 2 decades of study. In *Radiology: Imaging Cancer* (Vol. 2, Issue 2). Radiological Society of North America Inc. https://doi.org/10.1148/rycan.2020190058

Guerra-Manzanares, A., Lopez, L. J. L., Maniatakos, M., & Shamout, F. E. (2023). *Privacy-preserving machine learning for healthcare: open challenges and future perspectives*. https://doi.org/10.1007/978-3-031-39539-0_3

Heindl, Dr. A. (2022). *What's the Difference Between DICOM and NIfTI?* https://encord.com/blog/dicom-and-nifti-comparison/

IARC, W. H. O. (2022). *Cancer*. https://www.who.int/news-room/fact-sheets/detail/cancer

IARC, W. H. O. (2024). *Cancer Tomorrow*. https://gco.iarc.who.int/tomorrow/en

Islam, S. K. M. S., Nasim, M. A. Al, Hossain, I., Ullah, Dr. M. A., Gupta, Dr. K. D., & Bhuiyan, M. M. H. (2023). *Introduction of Medical Imaging Modalities*. http://arxiv.org/abs/2306.01022

Jiang, W., Cai, G., Hu, P., & Wang, Y. (2021). Personalized medicine of non-gene-specific chemotherapies for non-small cell lung cancer. In *Acta Pharmaceutica Sinica B* (Vol. 11, Issue 11, pp. 3406–3416). Chinese Academy of Medical Sciences. https://doi.org/10.1016/j.apsb.2021.02.003

Jiang, W., Zeng, G., Wang, S., Wu, X., & Xu, C. (2022). Application of Deep Learning in Lung Cancer Imaging Diagnosis. In *Journal of Healthcare Engineering* (Vol. 2022). Hindawi Limited. https://doi.org/10.1155/2022/6107940

Kenton, W. (2023). *What Is a Black Box Model? Definition, Uses, and Examples*. https://www.investopedia.com/terms/b/blackbox.asp#citation-5

Keras. (2019). *Home - Keras Documentation*. https://keras.io/

Lecun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. In *Nature* (Vol. 521, Issue 7553, pp. 436–444). Nature Publishing Group. https://doi.org/10.1038/nature14539

Li, X., Morgan, P. S., Ashburner, J., Smith, J., & Rorden, C. (2016). The first step for neuroimaging data analysis: DICOM to NIfTI conversion. *Journal of Neuroscience Methods*, *264*, 47–56. https://doi.org/10.1016/j.jneumeth.2016.03.001

Lin, Z., Zheng, J., & Hu, W. (2020). Using 3D convolutional networks with shortcut connections for improved lung nodules classification. *ACM International Conference Proceeding Series*, 42–49. https://doi.org/10.1145/3404512.3404525

Lundberg, S., & Lee, S.-I. (2017). *A Unified Approach to Interpreting Model Predictions*. http://arxiv.org/abs/1705.07874

Luo, G., Zhang, Y., Etxeberria, J., Arnold, M., Cai, X., Hao, Y., & Zou, H. (2023). Projections of Lung Cancer Incidence by 2035 in 40 Countries Worldwide: Population-Based Study. *JMIR Public Health and Surveillance*, *9*, e43651. https://doi.org/10.2196/43651

Markotić, V., Pojužina, T., Radančević, D., Miljko, M., & Pokrajčić, V. (2021). The Radiologist Workload Increase; Where Is the Limit?: Mini Review and Case Study. *Psychiatria Danubina*, *33*, 768–770. https://pubmed.ncbi.nlm.nih.gov/34718316/#:~:text=Daily%20average%20of%20analyzed%20imaging

Mason, D., scaramallion, mrbean-bremen, rhaxton, Suever, J., Vanessasaurus, Orfanos, D. P., Lemaitre, G., Panchal, A., Rothberg, A., Herrmann, M. D., Massich, J., Kerns, J., van Golen, K., Robitaille, T., Biggs, S., moloney, Bridge, C., Shun-Shin, M., … Wortmann, J. (2023). *pydicom/pydicom: Pydicom v2.4.0*. Zenodo. https://doi.org/10.5281/zenodo.8034250

Mayekar, N., Pattewar, S., Patil, S., & Dhruv, A. (2022). Preliminary Lung Cancer Detection using Deep Neural Networks. *International Research Journal of Engineering and Technology*. www.irjet.net

Mayo, J. R. (2009). *CT Evaluation of Diffuse Infiltrative Lung Disease Dose Considerations and Optimal Technique*. www.thoracicimaging.com

McKelvey, T., Ahmad, M., Teredesai, A., & Eckert, C. (2018). *Interpretable Machine Learning in Healthcare*.

Mi, X., Zou, B., Zou, F., & Hu, J. (2021). Permutation-based identification of important biomarkers for complex diseases via machine learning models. *Nature Communications*, *12*(1). https://doi.org/10.1038/s41467-021-22756-2

Molnar, C. (2022). *Interpretable machine learning : a guide for making black box models explainable* (2nd ed.). christophm.github.io/interpretable-ml-book/

National Cancer Institute, National Institutes of Health, & U.S. Department of Health and Human Services. (2021, October 11). *What Is Cancer?* National Cancer Institute: Patient Education Publications. https://www.cancer.gov/about-cancer/understanding/what-is-cancer

NEMA PS3 / ISO 12052, Digital Imaging and Communications in Medicine (DICOM) Standard, N. E. M. A., & National Electrical Manufacturers Association. (2024). *About DICOM: Overview*. ISO 12052 Standard. http://www.dicomstandard.org/

Panesar, A. (2019). Machine Learning and AI for Healthcare: Big Data for Improved Health Outcomes. In *Machine Learning and AI for Healthcare: Big Data for Improved Health Outcomes*. Apress Media LLC. https://doi.org/10.1007/978-1-4842-3799-1

Ramana, K., Kumar, M. R., Sreenivasulu, K., Gadekallu, T. R., Bhatia, S., Agarwal, P., & Idrees, S. M. (2022). Early Prediction of Lung Cancers Using Deep Saliency Capsule and Pre-Trained Deep Learning Frameworks. *Frontiers in Oncology*, *12*. https://doi.org/10.3389/fonc.2022.886739

Rani, S., Ghai, D., Kumar, S., Kantipudi, M. P., Alharbi, A. H., & Ullah, M. A. (2022). Efficient 3D AlexNet Architecture for Object Recognition Using Syntactic Patterns from Medical Images. *Computational Intelligence and Neuroscience*, *2022*. https://doi.org/10.1155/2022/7882924

Ribeiro, M. T., Singh, S., & Guestrin, C. (2016). "Why should i trust you?" Explaining the predictions of any classifier. *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, *13-17-August-2016*, 1135–1144. https://doi.org/10.1145/2939672.2939778

Sarker, I. H. (2021). Deep Learning: A Comprehensive Overview on Techniques, Taxonomy, Applications and Research Directions. In *SN Computer Science* (Vol. 2, Issue 6). Springer. https://doi.org/10.1007/s42979-021-00815-1

Selvaraju, R. R., Cogswell, M., Das, A., Vedantam, R., Parikh, D., & Batra, D. (2017). Grad-CAM: Visual Explanations from Deep Networks via Gradient-Based Localization. *Proceedings of the IEEE International Conference on Computer Vision*, *2017-October*, 618–626. https://doi.org/10.1109/ICCV.2017.74

Shady Hermena, & Michael Young. (2023, August 8). *CT-scan Image Production Procedures*. National Library of Medicine. https://www.ncbi.nlm.nih.gov/books/NBK574548/

Shafi, I., Din, S., Khan, A., Díez, I. D. L. T., Casanova, R. del 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). https://doi.org/10.3390/cancers14215457

Sori, W. J., Feng, J., & Liu, S. (2019). Multi-path convolutional neural network for lung cancer detection. *Multidimensional Systems and Signal Processing*, *30*(4), 1749–1768. https://doi.org/10.1007/s11045-018-0626-9

Speith, T. (2022). A Review of Taxonomies of Explainable Artificial Intelligence (XAI) Methods. *ACM International Conference Proceeding Series*, 2239–2250. https://doi.org/10.1145/3531146.3534639

Stiglic, G., Kocbek, P., Fijacko, N., Zitnik, M., Verbert, K., & Cilar, L. (2020). Interpretability of machine learning-based prediction models in healthcare. In *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery* (Vol. 10, Issue 5). Wiley-Blackwell. https://doi.org/10.1002/widm.1379

Sujatha, M., & Prabhakar, S. (2019). Issue 3 www.jetir.org (ISSN-2349-5162). In *JETIRAU06101 Journal of Emerging Technologies and Innovative Research* (Vol. 6). JETIR. www.jetir.org

Sun, B., Liu, F., Zhou, Y., Jin, S., Li, Q., & Jin, X. (2020, October 20). Classification of Lung Nodules Based on GAN and 3D CNN. *ACM International Conference Proceeding Series*. https://doi.org/10.1145/3424978.3425094

Tandon, R., Agrawal, S., Chang, A., & Band, S. S. (2022). VCNet: Hybrid Deep Learning Model for Detection and Classification of Lung Carcinoma Using Chest Radiographs. *Frontiers in Public Health*, *10*. https://doi.org/10.3389/fpubh.2022.894920

TensorFlow. (2019). *TensorFlow*. Google. https://www.tensorflow.org/

Venkadesh, K. V., & Jacobs, C. (2022). *LUNA22-ISMI*. Zenodo. https://doi.org/10.5281/zenodo.6559584

Whittaker, M. (2019). *Disability, Bias, and AI*.

WHO, I. (2022). *Cancer*. https://www.who.int/news-room/fact-sheets/detail/cancer

Woo, S., Park, J., Lee, J.-Y., & Kweon, I. S. (n.d.). *CBAM: Convolutional Block Attention Module*.

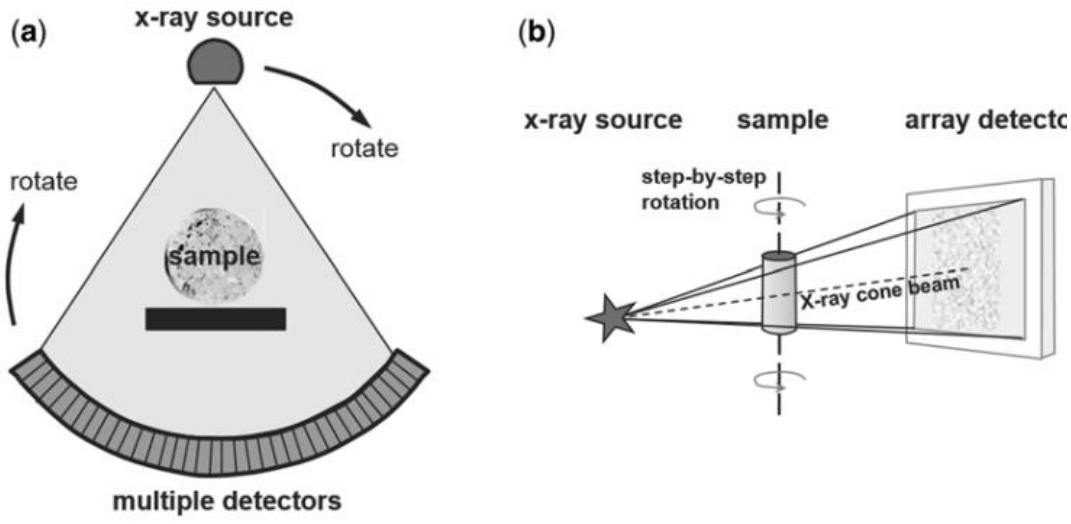
Zunair, H., Rahman, A., Mohammed, N., & Cohen, J. P. (2020). *Uniformizing Techniques to Process CT scans with 3D CNNs for Tuberculosis Prediction*. http://arxiv.org/abs/2007.13224

APPENDICES

A. CT Scan image - Advanced X-ray Imaging

CT scans utilize specialized detectors to capture multiple X-ray images from various angles simultaneously. This allows for detailed 3D reconstructions of the body. Additionally, techniques like multi-energy imaging and spatial filtering enhance image quality, while dose modulation minimizes radiation exposure (Islam et al., 2023). CT scan images can be viewed in various formats, including cross-sections, 3D models, and even VR

The CT scanner is a donut-shaped machine that the patient lies down on a table that slides into it. The scanner's internal X-ray tube revolves around the patient, shooting a variety of X-ray beams at various angles. A pattern of attenuated X-rays is produced as the X-ray beams travel through the body and are absorbed to differing degrees by various tissues. An array of detectors placed opposite from the X-ray tube, which transforms the attenuated X-ray pattern into electrical impulses, picks up on it. After receiving these electrical impulses, a computer employs complex algorithms to rebuild the data into a three-dimensional picture (Islam et al., 2023).



Principle of Computed Tomography Imaging

The images from a CT-scan are not like ordinary jpeg images, they have different formats. The most common formats are DICOM and NIfTI. Let’s see how the CT scanner detectors detect and pass the images to the image processor.

CT Scan Hardware:

- X-ray Generator: Produces the X-ray beam.
- Gantry: Houses the X-ray tube and detectors, rotating around the patient for 360-degree views.
- X-ray Tube: Converts electricity into X-rays.
- Photon Detectors: Capture and measure X-rays passing through the body.
- Shielding Elements: Absorb scattered X-rays, minimizing noise and radiation exposure.
- Patient Table: Moves the patient through the gantry during the scan.
- Image Processor: Processes the raw scan data.
- Console: Controls the entire scanning process.

B. Important Factors to Consider with CT-scan Images

Image Windowing: When viewing CT images, the human eye can differentiate much less grey colors, even though the HU values are expressed on a 5000-unit scale. The displayed greyscale brightness and contrast can be adjusted by adjusting the number of included Hounsfield Units (referred to as the "window") and/or the Hounsfield Unit value that is set as the central/middle value (referred to as the "level"). The image viewer can manually change these parameters by interacting with the image software with a mouse. "Windowing" or "changing the window" refers to altering what is shown in the image by shifting the visible HU in this way (Shady Hermena & Michael Young, 2023).

The window width determines the contrast of the image. As breadth increases, the contrast diminishes. The range of HU present in the image is displayed by the window width. It is appropriate to examine structures with similar attenuation values through a small window (50-350 HU). It is appropriate to examine structures with significantly varied attenuation values using a large window (400-2000 HU).

Level: The brightness of the image is dependent on the window level. The brightness increases as the level does. By adjusting the window level, the central or midpoint gray value for the HU range displayed in the image can be found. The brightness of the image increases with increasing window level and vice versa. The window level also affects the best imaging of various tissues (Shady Hermena & Michael Young, 2023).. More dense tissues, higher level and lower dense tissues lower level.

The wider the window, more densities seen but less contrast. And narrow window, less densities seen but more contrast. Window (W) is how many HU within 256 shades of grey while Level (L) is where the window is centered. Anything, lower than the window is going to appear black and anything higher than the window is going to look white. {i.e. < L – 1/2W = black and > L + 1/2W = white}. For lung the window is commonly +1500 and level is commonly -700.

Planes and Orientations: The capacity to fully rebuild the pictures in the **Axial or Transversal, sagittal,** and **Coronal or Frontal** planes is one benefit of obtaining a volume acquisition CT scan. When assessing the degree of disease in a patient, it is especially useful to view the anatomy and pathology in all three planes (Mayo, 2009).

- Anterior: Refers to the front of the body, closer to the head.
- Posterior: Refers to the back of the body, closer to the rear.

CT scan images can be viewed in various formats:

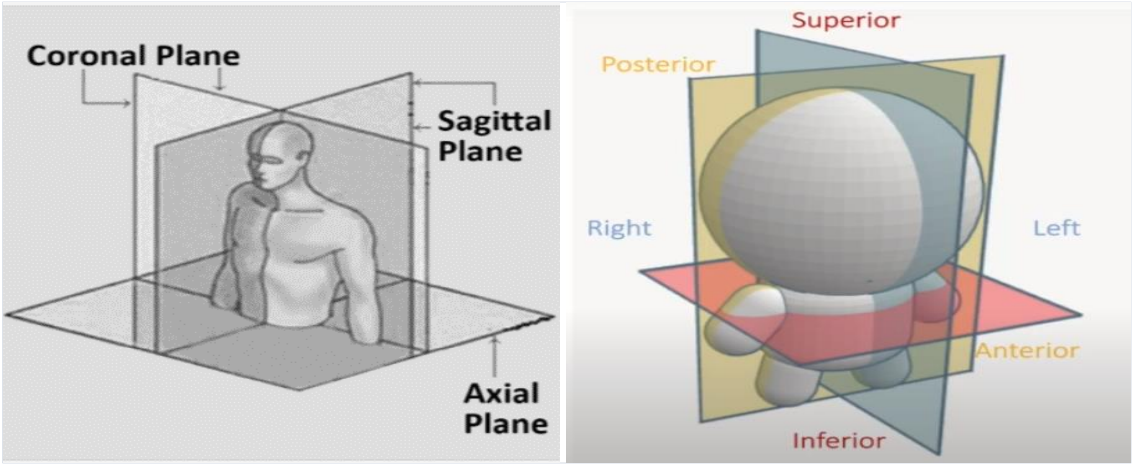
Axial Slices: These are cross-sectional images, like slices of bread, providing a horizontal view of the body at different levels.

Coronal Slices: These slices are oriented vertically, providing a view from the front to the back of the body.

Sagittal Slices: These slices are also vertical, but oriented from side to side, offering a view from the left to the right of the body.

3D Reconstructions: These are computer-generated 3D models of the scanned body, allowing for visualization from any angle.

Anterior: nearer the front, especially in the front of the body, or nearer to the head or forepart. **Posterior** further back in position; of or nearer the rear or hind end. Next, let's see the different formats of CT-scan images.



Planes and Orientations and Anatomical Orientations

C. How Does Low-Dose CT-scans Show Whether We Have Lung Cancer?

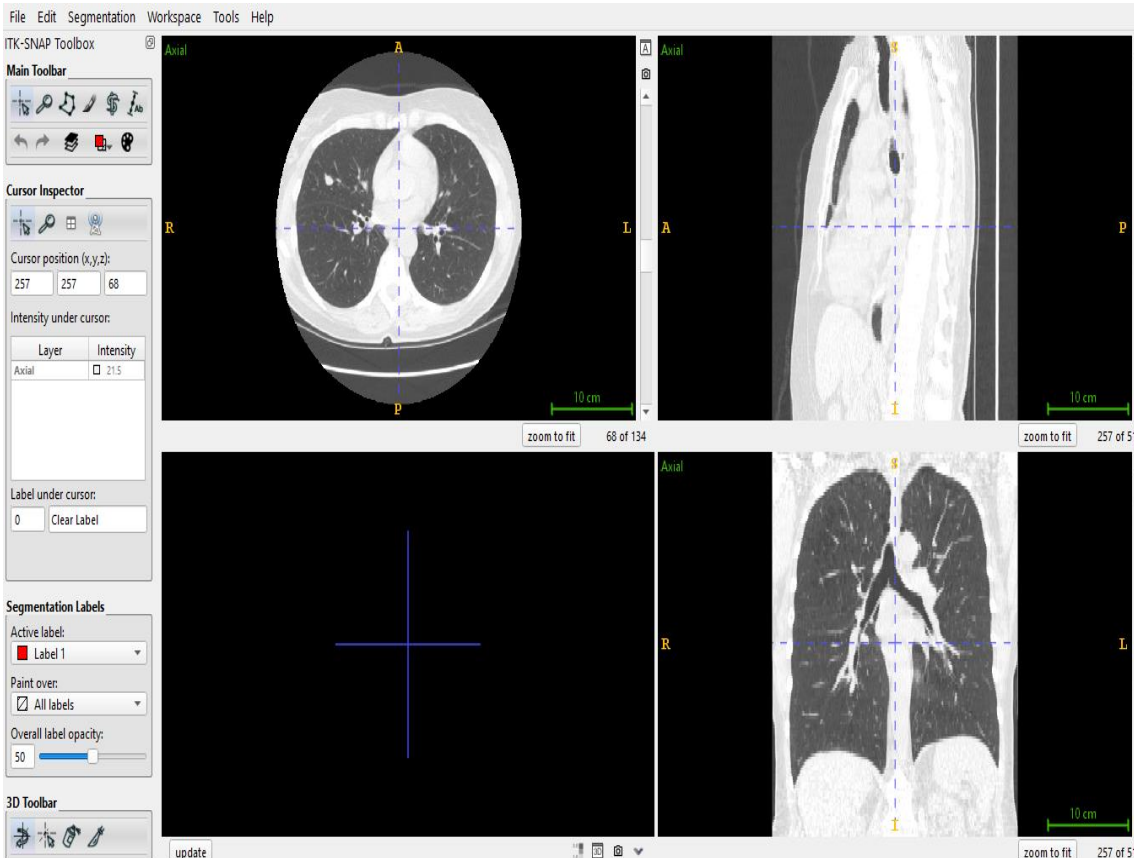
Unlike traditional CT scans, LDCT employs a significantly lower radiation dose, making it a safer option for screening purposes (National Cancer Institute et al., 2021). These scans excel at visualizing small nodules within the lungs, which may be indicative of early-stage lung cancer (Gierada et al., 2020). Standard chest X-rays often lack the sensitivity to detect such subtle nodules. The ability of LDCT to identify these early lesions allows for intervention at a more treatable stage, potentially improving patient prognosis

While LDCT aids in visualization, interpreting the findings necessitates a structured approach. Radiologists rely on established guidelines, such as those outlined in the Lung-RADS classification system, to ensure consistent interpretation across healthcare settings. These guidelines provide a framework for evaluating nodule characteristics, including size, number, and morphology (solid, part-solid, or non-solid). Based on this evaluation, a Lung-RADS score (1-4) is assigned to the scan (American College of Radiology, 2023).

Lung-RADS 1: No nodules identified, indicating a low suspicion for cancer. Follow-up screening is typically recommended in one year (National Cancer Institute et al., 2021)
Lung-RADS 2: A small, non-concerning nodule is present. Similar to Lung-RADS 1, follow-up screening at one year is often advised (National Cancer Institute et al., 2021).
Lung-RADS 3: A small nodule with features that warrant closer monitoring. In this case, a follow-up scan at six months might be recommended (National Cancer Institute et al., 2021).
Lung-RADS 4: This category encompasses nodules with a higher suspicion for malignancy based on their appearance on the CT scan. While LDCT can provide valuable information regarding nodule characteristics, it's important to acknowledge limitations. Distinguishing benign from malignant nodules solely based on imaging can be challenging, and additional investigations such as biopsies may be necessary for definitive diagnosis (National Cancer Institute et al., 2021).

The textural characteristics of nodules, categorized as solid, part-solid, or non-solid on LDCT scans, can offer clues regarding their potential malignancy (Sujatha & Prabhakar, 2019).Solid nodules, appearing opaque on the scan, are generally considered more suspicious for cancer compared to non-solid nodules, where underlying lung tissue can be visualized(American College of Radiology, 2023).

Part-solid nodules exhibit a mixed appearance, with both solid and non-solid components. These require careful evaluation and may warrant further investigation depending on their specific features (Gierada et al., 2020).



D. Model Architecture Summaries of the Four Models Used

Number of parameters for each layer for all the three models can be calculated using the formulas in *eq. (F.1)* and *eq. (F.2)*.

- **For convolutional layer**:

$$\#\ param = (w * h * d * Pf * + 1) * Cf \qquad \textit{eq. (F.1)}$$

- **For FC-layer:**

$$\#\ param = (Pn + 1) * Cn \qquad \textit{eq. (F.2)}$$

Where "# param" is total number of parameters in the layer, "w" is filter width, "h" is filter height, "d" is filter depth, "Pf" is number of filters of the previous layer, "Cf" is number of filters of the current layer, "Pn" is previous layer number of neurons, "Cn" is current layer number of neurons. ('None' in the output shape denotes the batch size)

Baseline Model / Model-1 Model Summary:

Layer	Output Shape	Param #
Input Layer	[(None, 64, 64, 64, 1)]	0
Conv3D	(None, 63, 63, 63, 64)	576
Max Pooling 3D	(None, 31, 31, 31, 64)	0
Batch Normalization	(None, 31, 31, 31, 64)	256
Conv3D	(None, 30, 30, 30, 64)	32,832
Max Pooling 3D	(None, 15, 15, 15, 64)	0
Batch Normalization	(None, 15, 15, 15, 64)	256
Conv3D	(None, 14, 14, 14, 128)	65,664
Max Pooling 3D	(None, 7, 7, 7, 128)	0
Batch Normalization	(None, 7, 7, 7, 128)	512
Conv3D	(None, 6, 6, 6, 256)	262,400
Max Pooling 3D	(None, 3, 3, 3, 256)	0
Batch Normalization	(None, 3, 3, 3, 256)	1,024
Global Average Pooling 3D	(None, 256)	0
Dense	(None, 1024)	263,168
Dropout	(None, 1024)	0
Dense	(None, 1)	1,025
Total params: 627,713 (2.39 MB) Trainable params: 626,689 (2.39 MB) Non-trainable params: 1,024 (4.00 KB)		

3D AlexNet Model / Model -2 Model Summary:

Layer	Output Shape	Param #
Input Layer	[(None, 64, 64, 64, 1)]	0
Conv3D	(None, 30, 30, 31, 96)	7,296
Batch Normalization	(None, 30, 30, 31, 64)	384
Max Pooling 3D	(None, 28, 28, 29, 96)	0
Conv3D	(None, 26, 26, 27, 128)	331,904
Batch Normalization	(None, 26, 26, 27, 128)	512
Max Pooling 3D	(None, 12, 12, 13, 128)	0
Conv3D	(None, 12, 12, 13, 256)	884,992
Batch Normalization	(None, 12, 12, 13, 256)	1,024
Conv3D	(None, 12, 12, 13, 384)	2,654,592
Batch Normalization	(None, 12, 12, 13, 384)	1,536
Conv3D	(None, 12, 12, 13, 256)	2,654,464
Batch Normalization	(None, 12, 12, 13, 256)	1,024
Max Pooling 3D	(None, 6, 6, 6, 256)	0
Conv3D	(None, 3, 3, 3, 256)	1,769,728
Batch Normalization	(None, 3, 3, 3, 256)	1,024
Flatten	(None, 6912)	0
Dense	(None, 4096)	28,315,648
Batch Normalization	(None, 4096)	16,384
Dense	(None, 1024)	4,195,328
Batch Normalization	(None, 1024)	4,096
Dropout	(None, 1024)	0
Dense	(None, 1)	1,025
Total params: 40,840,961 (155.80 MB) Trainable params: 40,827,969 (155.75 MB) Non-trainable params: 12,992 (50.75 KB)		

Proposed 3D CNN Model / Model-3 Model Summary:

Layer	Output Shape	Param #
Input Layer	[(None, 64, 64, 64, 1)]	0
Conv3D	(None, 63, 63, 63, 64)	576
Max Pooling 3D	(None, 31, 31, 31, 64)	0
Batch Normalization	(None, 31, 31, 31, 64)	256
Conv3D	(None, 30, 30, 30, 128)	65,664
Max Pooling 3D	(None, 15, 15, 15, 128)	0
Batch Normalization	(None, 15, 15, 15, 128)	512
Conv3D	(None, 14, 14, 14, 128)	131,200
Max Pooling 3D	(None, 7, 7, 7, 128)	0
Batch Normalization	(None, 7, 7, 7, 128)	512
Conv3D	(None, 6, 6, 6, 128)	131,200
Max Pooling 3D	(None, 3, 3, 3, 128)	0
Batch Normalization	(None, 3, 3, 3, 128)	512
Flatten	(None, 3456)	0
Batch Normalization	(None, 3456)	13, 824
Dense	(None, 4096)	14,159,872
Dense	(None, 4096)	16,781,312
Dropout	(None, 4096)	0
Dense	(None, 1)	4,097
Total params: 31,289,537 (119.36 MB) Trainable params: 31,281,729 (119.33 MB) Non-trainable params: 7,808 (30.50 KB)		

Proposed 3D CNN Model with CBAM / Model-4 Model Summary:

Layer	Output Shape	Param #	Connected to
Input Layer	[(None, 64, 64, 64, 1)]	0	-
Conv3D-1	(None, 63, 63, 63, 64)	576	Input Layer
Max Pooling 3D-1	(None, 31, 31, 31, 64)	0	Conv3D-1
Batch Normalization-1	(None, 31, 31, 31, 64)	256	Max Pooling 3D-1
Conv3D-2	(None, 30, 30, 30, 128)	65,664	Batch Normalization-1
Max Pooling 3D-2	(None, 15, 15, 15, 128)	0	Conv3D-2
Batch Normalization-2	(None, 15, 15, 15, 128)	512	Max Pooling 3D-2
Conv3D-3	(None, 14, 14, 14, 128)	131,200	Batch Normalization-2
Max Pooling 3D-3	(None, 7, 7, 7, 128)	0	Conv3D-3
Batch Normalization-3	(None, 7, 7, 7, 128)	512	Max Pooling 3D-3
Conv3D-4	(None, 6, 6, 6, 128)	131,200	Batch Normalization-3
Max Pooling 3D-4	(None, 3, 3, 3, 128)	0	Conv3D-4
Batch Normalization-4	(None, 3, 3, 3, 128)	512	Max Pooling 3D-4
Global Average Pooling 3D-1	(None, 128)	0	Batch Normalization-4
Global Max Pooling 3D-1	(None, 128)	0	Batch Normalization-4
Reshape-1	(None, 1, 1, 1, 128)	0	Global Average Pooling 3D-1
Reshape-2	(None, 1, 1, 1, 128)	0	Global Max Pooling 3D-1
Dense-1	(None, 1, 1, 1, 16)	2,064	Reshape-1
Dense-2	(None, 1, 1, 1, 16)	2,064	Reshape-2
Dense-3	(None, 1, 1, 1, 128)	2,176	Dense-1
Dense-4	(None, 1, 1, 1, 128)	2,176	Dense-2
Add-1	(None, 1, 1, 1, 128)	0	Dense-1 and Dense-2
Activation	(None, 1, 1, 1, 128)	0	Add-1
Multiply-1	(None, 1, 1, 1, 128)	0	Batch Normalization-4 and Activation
Global Average Pooling 3D-2	(None, 128)	0	Multiply-1
Global Max Pooling 3D-2	(None, 128)	0	Multiply-1
Add-2	(None, 128)	0	Global Average Pooling 3D-1 and Global Max Pooling 3D-2
Reshape-3	(None, 1, 1, 1, 128)	0	Add-2
Conv3D-5	(None, 1, 1, 1, 1)	43,905	Reshape-3
Multiply-2	(None, 3, 3, 3, 3)	0	Multiply-1 and Conv3D-5
Batch Normalization-5	(None, 3, 3, 3, 128)	512	Multiply-2
Flatten	(None, 3456)	0	Batch Normalization-5
Batch Normalization-6	(None, 3456)	13, 824	Flatten
Dense-5	(None, 4096)	14,159,872	Batch Normalization-6
Dense-6	(None, 4096)	16,781,312	Dense-5
Dropout	(None, 4096)	0	Dense-6
Dense-7	(None, 1)	4,097	Dropout
Total params: 31,342,434 (119.56 MB) Trainable params: 31,334,370 (119.53 MB) Non-trainable params: 8,064 (31.50 KB)			

E. Code Snippets

Proposed Model with CBAM

```
from keras.layers import Input, Conv3D, MaxPooling3D, BatchNormalization, Flatten, Dense, Dropout
from keras.layers import GlobalAveragePooling3D, GlobalMaxPooling3D, Reshape, Multiply, Add, Activation
from keras.models import Model
from keras.regularizers import l2
def channel_attention(x, ratio=8):
    """Channel Attention Module."""
    channel_axis = -1
    channels = x.shape[channel_axis]
    # Global Average Pooling
    avg_pool = GlobalAveragePooling3D()(x)
    avg_pool = Reshape((1, 1, 1, channels))(avg_pool)
    avg_pool = Dense(channels // ratio, activation='relu')(avg_pool)
    avg_pool = Dense(channels, activation='sigmoid')(avg_pool)
    # Global Max Pooling
    max_pool = GlobalMaxPooling3D()(x)
    max_pool = Reshape((1, 1, 1, channels))(max_pool)
    max_pool = Dense(channels // ratio, activation='relu')(max_pool)
    max_pool = Dense(channels, activation='sigmoid')(max_pool)
    # Combine both attention vectors
    attention = Add()([avg_pool, max_pool])
    attention = Activation('sigmoid')(attention)
    return Multiply()([x, attention])
def spatial_attention(x):
    """Spatial Attention Module."""
    avg_pool = GlobalAveragePooling3D()(x)
    max_pool = GlobalMaxPooling3D()(x)
    concat = Add()([avg_pool, max_pool])
    concat = Reshape((1, 1, 1, x.shape[-1]))(concat)
    # Convolution to generate spatial attention map
    conv = Conv3D(filters=1, kernel_size=7, padding='same', activation='sigmoid')(concat)
    return Multiply()([x, conv])
def cbam(x):
    """CBAM Module."""
    x = channel_attention(x)
    x = spatial_attention(x)
    return x
def get_model_3(width=64, height=64, depth=64, l2_reg=0.01):
    """Build a 3D convolutional neural network model with L2 regularization and CBAM."""
    inputs = Input((width, height, depth, 1))
    x = Conv3D(filters=64, kernel_size=2, activation='relu', kernel_regularizer=l2(l2_reg))(inputs)
    x = MaxPooling3D(pool_size=2)(x)
    x = BatchNormalization()(x)
    x = Conv3D(filters=128, kernel_size=2, activation='relu', kernel_regularizer=l2(l2_reg))(x)
    x = MaxPooling3D(pool_size=2)(x)
    x = BatchNormalization()(x)
    x = Conv3D(filters=128, kernel_size=2, activation='relu', kernel_regularizer=l2(l2_reg))(x)
    x = MaxPooling3D(pool_size=2)(x)
    x = BatchNormalization()(x)
    x = Conv3D(filters=128, kernel_size=2, activation='relu', kernel_regularizer=l2(l2_reg))(x)
    x = MaxPooling3D(pool_size=2)(x)
    x = BatchNormalization()(x)
    # Apply CBAM after the last convolutional block
    x = cbam(x)
    x = BatchNormalization()(x)
    x = Flatten()(x)
    x = BatchNormalization()(x)
    x = Dense(4096, activation='relu', kernel_regularizer=l2(l2_reg))(x)
    x = Dense(4096, activation='relu', kernel_regularizer=l2(l2_reg))(x)
    x = Dropout(0.7)(x)
    outputs = Dense(units=1, activation="sigmoid")(x)
    return Model(inputs, outputs, name='Proposed-3D-CNN-with-CBAM-and-L2')
```

Model Interpretation

```
def make_gradcam_heatmap(img_array, model, last_conv_layer_name, pred_index=None):
  """Generate class activation heatmap"""
  # First, we create a model that maps the input image to the activations
  # of the last conv layer as well as the output predictions
  grad_model = Model(
      [model.inputs], [model.get_layer(
          last_conv_layer_name).output, model.output]
  )

  # Then, we compute the gradient of the top predicted class for our input image
  # with respect to the activations of the last conv layer
  with tf.GradientTape() as tape:
    last_conv_layer_output, preds = grad_model(img_array)
    if pred_index is None:
        pred_index = tf.argmax(preds[0])
    class_channel = preds[:, pred_index]

  # This is the gradient of the output neuron (top predicted or chosen)
  # with regard to the output feature map of the last conv layer
  grads = tape.gradient(class_channel, last_conv_layer_output)

  # This is a vector where each entry is the mean intensity of the gradient
  # over a specific feature map channel (equivalent to global average pooling)
  pooled_grads = tf.reduce_mean(grads, axis=(0, 1, 2, 3))
```

```
  # We multiply each channel in the feature map array
  # by 'how important this channel is' with regard to the top predicted class
  # then sum all the channels to obtain the heatmap class activation
  last_conv_layer_output = last_conv_layer_output[0]
  heatmap = last_conv_layer_output @ pooled_grads[..., tf.newaxis]
  heatmap = tf.squeeze(heatmap)

  # For visualization purpose, we will also normalize the heatmap between 0 & 1
  # Notice that we clip the heatmap values, which is equivalent to applying ReLU
  heatmap = tf.maximum(heatmap, 0) / tf.math.reduce_max(heatmap)
  return heatmap.numpy()
```

Expanding Heatmap Dimensions

```
def get_resized_heatmap(heatmap, shape):
    """Resize heatmap to shape"""
    # Rescale heatmap to a range 0-255
    upscaled_heatmap = np.uint8(255 * heatmap)

    upscaled_heatmap = zoom(
        upscaled_heatmap,
        (
            shape[0] / upscaled_heatmap.shape[0],
            shape[1] / upscaled_heatmap.shape[1],
            shape[2] / upscaled_heatmap.shape[2],
        ),
    )

    return upscaled_heatmap


resized_heatmap = get_resized_heatmap(heatmap, input_volume.shape)
```