LEVERAGING DEEP LEARNING FOR ACCURATE LUNG CANCER DETECTION

BY

LEE LI JIE

A REPORT

SUBMITTED TO

Universiti Tunku Abdul Rahman

in partial fulfillment of the requirements

for the degree of

BACHELOR OF INFORMATION SYSTEMS (HONOURS) INFORMATION SYSTEMS ENGINEERING

Faculty of Information and Communication Technology

(Kampar Campus)

JUNE 2025

COPYRIGHT STATEMENT

ACKNOWLEDGEMENTS

First and foremost, I would like to express my deepest gratitude to my supervisor, Abdulkarim Kanaan Jebna and my moderator, Dr Ramesh Kumar Ayyasamy, for their invaluable guidance, continuous support, and encouragement throughout the course of this project. Their expertise, constructive feedback, and patience have been instrumental in shaping the direction of my work.

I am also grateful to Universiti Tunku Abdul Rahman (UTAR) and the Department of Faculty of Information and Communication Technology for providing the resources and opportunities to undertake this Final Year Project. This journey has not only strengthened my academic foundation but also laid the groundwork for my future aspirations in the field of IT design.

Lastly, I would like to dedicate my deepest appreciation to my beloved parents and family members. Their unconditional love, moral support, and encouragement have been the pillar of strength that helped me persevere and complete this project successfully.

ABSTRACT

This research falls in the Artificial Intelligence in Medical Imaging applications to Computer-Aided Diagnosis (CAD) category. Lung cancer is one of the leading global causes of death due to cancer, where survival directly depends on early detection and accurate tumour localisation [2]. Conventional pipelines rely heavily on manual interpretation of CT scans, which are time-consuming, subjective, and prone to errors [2]. Modern CAD frameworks also have limitations, with classification-only pipelines lacking localisation, segmentation-first pipelines squandering computation, and parallel pipelines introducing redundancy [15]. To address these problems, this project proposes a classification-first pipeline with EfficientNetB1 for lung cancer classification and conditional Fuzzy C-Means (FCM) segmentation module activated only after cancer detection. The system was developed and deployed via an interactive Streamlit dashboard, which integrates multiple functions like classification, segmentation, lesion measurement, structured reporting (PDF and AIM XML), review by radiologists, and storing cases with SQLite. The workflow included preparation of the dataset from the Kaggle 2D Chest CT Scan dataset, strict preprocessing and augmentation, and comparison-based evaluation of EfficientNetB0 and EfficientNetB1 models with stratified k-fold cross-validation. EfficientNetB1 was employed as the ultimate classifier with accuracy of 94.60% on the test set. The FCM segmentation produced clinically interpretable tumour overlays, and the dashboard successfully integrated error handling, simulation of workflow, and peer review by radiologists. The project concludes with a light, modular, and clinically useful diagnostic system for the assistance of radiologists in the early detection of lung cancer. The future holds quantitative assessment of segmentation and scaling to PACS integration for clinical deployment.

Area of Study (Minimum 1 and Maximum 2): Machine Learning, Deep Learning

Keywords (Minimum 5 and Maximum 10): Lung Cancer Detection, Deep Learning, EfficientNetB1, Fuzzy C-Means (FCM), Segmentation, Classification-First Pipeline, Streamlit Dashboard, Computer-Aided Diagnosis (CAD), SQLite Case Management.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF SYMBOLS

α	Depth
β	Width
γ	Resolution

LIST OF ABBREVIATIONS

CT	Computed Tomography
AI	Artificial Intelligence
ML	Machine Learning
DL	Deep Learning
CNNs	Convolutional Neural Networks
ResNet	Deep Residual Network
DenseNet	Dense Convolutional Network
AIM	Annotation and Image Markup
NAS	Neural Architecture Search
MBConv	Mobile Inverted Bottleneck Convolution
SE	Squeeze-and-Excitation
RMSProp	Root Mean Square Propagation
SiLU	Sigmoid Linear Unit
CPU	Central Processing Unit
TL	Transfer Learning
GPU	Graphics Processing Unit
FCM	Fuzzy C-Means
GUI	Graphical User Interface
BIR	Binary Image Representation
CRISP-DM	Cross-Industry Standard Process for Data Mining
DSC	Dice Similarity Coefficient
CAD	Computer Aided Design
IoU	Intersection over Union
VDT	Volume Doubling Time
GLCM	Gray-Level Co-occurrence Matrix
GK	Gustafson-Kessel
CLAHE	Contrast Limited Adaptive Histogram Equalization
SFS	Sequential Forward Selection
XAI	Explainable AI
RPN	Region Proposal Network

Chapter 1
Introduction

This chapter provides a comprehensive overview of the project. It begins by establishing the urgency and importance of early lung cancer detection in the global health context, followed by a discussion of the limitations in traditional diagnostic approaches and existing computer-aided detection systems. The chapter then outlines the potential of deep learning—particularly transfer learning with EfficientNet and lightweight unsupervised segmentation—to overcome these limitations. The project's core problem is identified, along with a motivation rooted in real-world clinical needs. Subsequently, the chapter presents the project's objectives, scope, and expected deliverables, highlighting the unique contributions of the proposed classification-first, modular, and clinically-oriented pipeline. The chapter concludes with an overview of the report structure to guide the reader through the subsequent chapters.

1.1 Project Background

1.1.1 Lung Cancer: A Persistent Threat

Lung cancer ranks among the foremost causes of cancer death worldwide, with an estimated 1.8 million deaths every year as proposed by the World Health Organization. Prompt and true diagnosis of the disease is essential for enhancing survival in patients. Some of the most utilized medical imaging techniques in screening lung cancer are Computed Tomography (CT) scans, mainly due to the fact that they are highly effective in identifying pulmonary nodules and atypical growths [2].

Figure 1.1 illustrates the worldwide distribution of cancer-related fatalities in 2020 [2]. It emphasized the necessity of early detection of lung cancer, which has the potential to raise survival rates. Early detection can significantly reduce mortality rates as a result of negligence. A high percentage of individuals diagnosed with lung cancer do not survive because of delayed treatment [2].

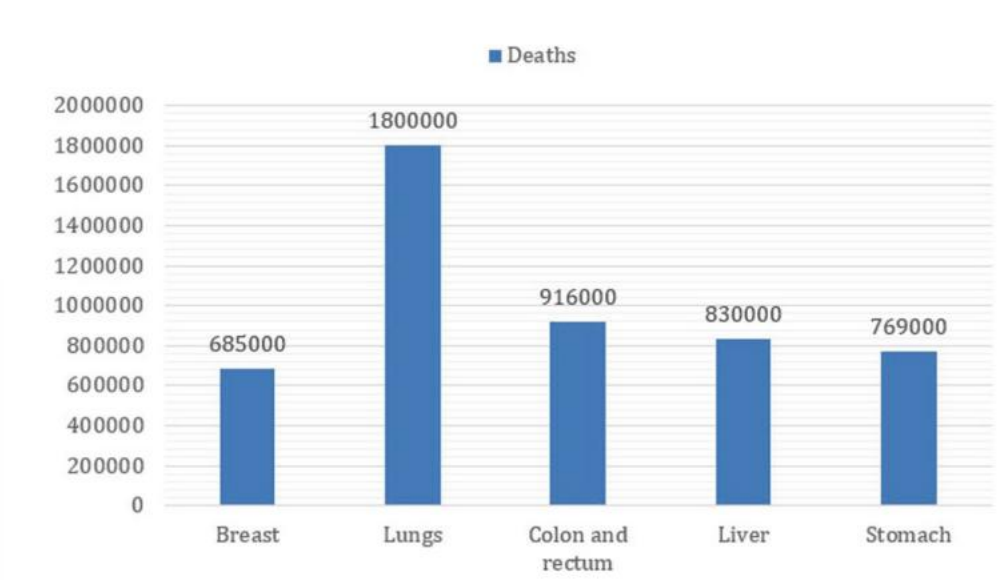


Figure 1.1 Worldwide Distribution of Cancer-Associated Mortality in 2020

1.1.2 The Challenge of Traditional Methods

Conventional diagnostic examinations, such as chest X-rays, possess some advantages but are not able to detect initial or subtle symptoms of lung cancer [3]. Radiographic imaging modalities, Computed Tomography (CT) scans in particular, have emerged as the principal non-invasive method for direct detection of lung cancer. CT scans generate high-resolution cross-sectional images of the lungs, allowing radiologists to detect abnormal nodules or suspicious tumors within lung tissue.

CT imaging is optimal for lung nodule detection due to its ability to produce high-resolution, three-dimensional (3D) images of the chest. This increased spatial resolution allows for better visualization of nodules and tumor pathology. Computer-aided interpretation of CT imaging has also been optimally utilized in clinical practice to assist in lung nodule diagnosis [16].

While CT scans significantly improve early detection prospects and effective treatment, they may still miss minute, asymptomatic nodules or faint changes in lung structure. Moreover, the sheer volume of imaging data, especially when acquired from multiple modalities, presents a challenge for manual interpretation. Such a task is time-consuming, prone to human error, and can lead to delayed diagnosis. These limitations detract from the effectiveness of early detection and can possibly have negative implications on patient outcome.

However, most existing computer-aided lung cancer detection systems fall into one of two design categories: segmentation-first or parallel pipelines. In segmentation-first approaches, the lung region or potential tumor is segmented from the CT scan before classification is performed. While this may aid spatial localization, it results in unnecessary computation for nodule-free scans. Parallel pipelines, on the other hand, perform classification and segmentation independently and simultaneously, which increases resource usage regardless of clinical necessity. Both approaches lack conditional control and lead to redundant processing on normal scans, limiting their efficiency and deployability in real-time or resource-constrained clinical environments.

1.1.3 The Promise of Deep Learning

The recent acceleration of artificial intelligence (AI) and especially machine learning (ML) has led to increased interest in creating automated diagnostic aids for the support of radiologists. As a subset of ML, deep learning has proved its unprecedented efficacy in classifying images and is being progressively integrated into medical image analysis. It alleviates the practitioners' workload by delivering high precision and dependability in diagnostic and predictive outcomes [2].

The two primary methods that have fueled recent developments in medical image analysis for lung cancer detection are deep learning (DL), which automatically extracts pertinent features from data, and traditional machine learning (ML), which relies on manually constructed feature extraction [6]. Because Convolutional Neural Networks (CNNs) can learn hierarchical representations directly from raw image inputs, they have become the most popular DL architecture for image classification. As a result, manual feature engineering is no longer necessary, and the model can effectively generalize even across different images that share common patterns. Nevertheless, CNNs usually need big datasets to function well, which can be problematic in medical fields where there is a shortage of high-quality labelled data [7].

One of the most promising methods for addressing the challenges posed by the lack of annotated data for medical image analysis is transfer learning (TL). In traditional deep learning, high-performance model training usually involves a significant quantity of high-quality labeled

data, which is difficult, expensive, and time-consuming to obtain in medicine. Transfer learning bypasses this constraint by utilizing pre-trained models on large data sets—such as ImageNet—and adapting them to domain-specific use such as lung cancer identification.

Transfer learning is employed here to fine-tune EfficientNetB1, a light and accurate convolutional neural network from Google AI. With its best balance between performance and model size, EfficientNetB1 is particularly well-suited for medical image classification tasks. Based on the pre-trained weights, the network can extract important features from CT images more effectively even with a relatively smaller training dataset. It not only accelerates training but also improves generalization performance, making the system deployable and robust in real-world clinical environments. This signifies a substantial advancement in the fight against lung cancer, with the capacity to transform diagnosis and therapy [4].

The application of transfer learning significantly enhances the accuracy and effectiveness of the diagnostic system constructed in this research. A deeper technical description of the approach, as well as supporting research demonstrating its efficacy in medical imaging, is presented in Chapter 2.

For enhanced interpretability and clinical relevance, the project also incorporates a Fuzzy C-Means (FCM) clustering segmentation module. Unlike deep segmentation networks like U-Net that require extensive training data and GPU support, FCM is computationally lightweight and unsupervised. The segmentation function is only triggered when the classifier predicts a cancerous scan, keeping unnecessary computation low. The FCM segmentation is also enhanced with application-specific heuristics including lung masking, brightness thresholding, morphological filtering, and lobe-based spatial constraints—resulting in improved tumor localization from CT scans.

As a whole, lung cancer is the deadliest disease worldwide, largely due to delays hindering early and accurate diagnosis. Conventional imaging techniques such as chest X-rays and CT scans are effective but limited by human judgment and escalating complexity in medical images. The recent advances in artificial intelligence, particularly deep learning, have presented beaming possibilities for the automation of lung cancer detection and diagnosis. This project proposes a two-stage deep learning framework employing EfficientNetB1 for accurate

lung cancer classification and a FCM-based method for precise tumor segmentation. With the application of transfer learning and robust convolutional models, the proposed system can improve diagnostic accuracy, reduce interpretation time, and provide clinical decision support for early lung cancer detection. Further technical justification and rationale behind model choice are discussed in the following chapter.

1.2 Problem Statement and Motivation

Despite remarkable progress in computer-aided lung cancer detection, the majority of existing systems are suffered by severe inefficiencies that make them impractical. The majority of modern approaches fall into two groups: (1) classification-only pipelines, which are able to detect the presence of cancer but do not have tumor location information, and (2) segmentation-first or parallel pipelines, where segmentation is performed on all the CT scans regardless of clinical necessity [15][29]. These schemes cause redundant and unnecessary computation, especially on regular (nodule-free) scans, hence leading to wasted processing time and additional computational cost [18].

Some studies have attempted to use classification-first pipelines, where segmentation is only triggered after cancer is detected [17][25][26]. However, these often rely on computationally expensive deep segmentation models (e.g., U-Net), and lack design features such as modular architecture, lightweight unsupervised clustering methods, or conditional activation logic to minimize resource use. Furthermore, Afshar et al. [18] note that traditional two-stage CAD systems are often overwhelmed by segmentation processes even when no tumor is present. Li et al. [19] emphasize that skipping segmentation entirely results in a loss of spatial context that hampers diagnostic precision. Khosravan et al. [20] point out that most modern solutions are multi-stage and computationally intensive, making them unsuitable for real-time use or deployment in resource-constrained clinical environments.These limitations highlight a critical gap in the current landscape: there is no lightweight, modular, and intelligent lung cancer detection system that performs classification first and only activates segmentation when necessary—thus optimizing for both accuracy and computational efficiency. This project aims to fill that gap.

This project is motivated by the need to create a lung cancer detection system that better aligns with real-world clinical workflows and resource limitations. As lung cancer continues to be a

leading cause of cancer-related deaths worldwide, the ability to accurately and efficiently detect tumors from CT scans is crucial for improving patient outcomes—especially in early-stage diagnosis where intervention is most effective. To address the problems outlined above, this project introduces a classification-first pipeline using EfficientNetB1 to detect the presence of cancer in CT scans. Only when a scan is classified as cancerous does the system activate a segmentation stage based on an enhanced Fuzzy C-Means (FCM) algorithm—a lightweight and unsupervised method that avoids the computational burden of deep models like U-Net.

The architecture is also modular, allowing the classification and segmentation components to function independently and be reused or extended. This separation not only improves maintainability but also enables conditional execution, meaning healthy scans can bypass segmentation entirely—saving time and resources. Moreover, the system produces clinically actionable outputs, including lesion overlays, tumor measurements (size, location, lobe), and Annotation Image Markup (AIM)-compliant XML reports for integration into hospital systems. This makes the system more than just a research prototype; it is designed to serve as a practical diagnostic assistant for radiologists, especially in low-resource settings or high-throughput screening environments.
By combining efficiency, interpretability, and practical utility, this project aims to bridge the gap between academic research and real-world clinical needs.

1.3 Objectives

This project's main goal is to create a lightweight, effective deep learning pipeline specifically designed for the early identification and localisation of lung cancer from CT scan pictures. The objectives of the project are shown below:

1. **To develop a lung cancer classification model using EfficientNetB1** that accurately distinguishes between normal and various types of cancerous chest CT scans.
2. **To evaluate and compare the classification performance** of EfficientNetB1 against a lighter baseline model (EfficientNetB0) using K-fold stratified cross-validation.
3. **To implement a modified Fuzzy C-Means (FCM) segmentation module** to generate tumor masks only when cancer is detected.

4. **To develop an interactive Streamlit-based medical dashboard** that supports lung cancer prediction, tumor segmentation, lesion measurement, radiologist review, and automated reporting for streamlined clinical decision support.
5. **To visualize tumor segmentation by overlaying masks on CT scans** and generate structured Annotation Image Markup (AIM) XML annotations for potential clinical integration.
6. **To implement an SQLite-based case storage system** that records patient details, predictions, radiologist reviews, and generated reports for easy retrieval.

This project does not involve hardware-level deployment, real-time clinical integration, or the development of user interfaces for hospital systems. Instead, it focuses on model development, validation, and performance analysis to facilitate early-stage lung cancer diagnosis and support clinical decision-making.

1.3 Project Scope and Direction

This project is intended to develop a lightweight, interpretable, and modular lung cancer diagnosis system from chest CT scan images. A publicly 2D Chest CT Scan Image Dataset on Kaggle is utilized, which has four classes of images: Normal, Adenocarcinoma, Large Cell Carcinoma, and Squamous Cell Carcinoma. Images are all in .jpg or .png format and pre-split into training, validation, and test datasets with a ratio of 70/10/20.

The system consists of three large functional blocks. First is classification module based on an EfficientNetB1-based deep learning architecture performing multi-classification (adenocarcinoma, squamous cell carcinoma, or large cell carcinoma, normal) of CT images. For comparison, an EfficientNetB0 model is also compared with K-fold cross-validation. Second block is segmentation module, which will be triggered if the image is classified to be cancerous. The module employs a custom Fuzzy C-Means (FCM) segmentation algorithm with lung masking, morphological operations (i.e., filling holes, removing small objects), brightness filtering, shape heuristics, and lobe-based filtering for the refinement of the tumor masks' accuracy. The reporting module overlays the detected tumor area and centroids on the original image and saves a PDF report, along with an AIM XML annotation file. These include lesion measurements such as area, volume, diameter, and lobe position. All case data is stored in a local SQLite database for convenient future retrieval and analysis.

Project deliverables are trained classification model, complete Streamlit-based interactive web dashboard, segmentation output with tumor mask and overlay mask, local case database, and structured annotations and formatted reports export. The system is intended for clinical practitioners (e.g., radiologists), medical imaging researchers, healthcare AI engineers, and clinical IT staff interested in deployable CAD systems.

The system supports the upload of a chest CT scan, automatic cancer presence classification, prediction confidence visualization, and—in the case of cancer—tumor mask generation and visualization with accurate measurements. It supports PDF and XML report downloading, simulation of radiologist feedback (accept, reject, edit), and patient case viewing or batch processing. But this project is not covering 3D segmentation or volumetric CNN models, DICOM image processing, PACS system real-time integration, real-time online deployment, clinical trials, or retraining with radiologist feedback. The system is developed as a proof-of-concept diagnostic tool for lung cancer diagnosis from 2D images.

1.4 Contributions

Lung cancer remains the leading cause of cancer-related deaths globally. However, early detection drastically improves patient outcomes: five-year survival rates exceed 70% for early-stage tumors, compared to less than 10% when diagnosed at a late stage [21][22]. These sobering statistics underscore the urgent need for accurate, efficient, and deployable diagnostic systems for early lung cancer detection.

This project proposes a lightweight, modular deep learning pipeline for tumor detection and localization from CT scans. Unlike many previous systems that are either resource-intensive or focus only on classification or segmentation, this work combines both in a classification-first architecture [15][29]. The pipeline uses EfficientNetB1 for multi-class classification (normal, adenocarcinoma, squamous cell carcinoma, large cell carcinoma) and conditionally activates a modified Fuzzy C-Means (FCM) segmentation step only when cancer is predicted. This mirrors real-world triage workflows, reduces redundant computation on healthy scans,

and makes the system suitable for low-resource settings like small clinics or academic hospitals.

The segmentation module is enhanced with domain-specific heuristics—such as lung masking, brightness and shape filtering, and lobe-based spatial constraints—which improve tumor mask precision and reduce false positives without relying on deep models like U-Net.

Beyond detection, the system emphasizes clinical usability. It automatically generates structured PDF reports with tumor overlays, measurements (area, diameter), laterality, and lobe information. It also exports AIM-compliant XML annotations, facilitating integration with hospital PACS and structured reporting systems. These features, often absent in academic prototypes, are vital for real-world clinical adoption.

In summary, this project delivers a clinically interpretable, resource-efficient, and deployable lung cancer detection system that balances accuracy, modularity, and real-world readiness—bridging the gap between AI research and practical healthcare deployment.

1.5 Report Organization

This report is organised into seven chapters. **Chapter 1** introduces the project, including its background, motivation, scope, objectives, contributions, and organisation. **Chapter 2** reviews related works on lung cancer detection and segmentation methods, highlighting existing CAD systems and research gaps. **Chapter 3** presents the overall system design, including the architecture, workflow, and dashboard features. **Chapter 4** details the system implementation, covering dataset preparation, model development, segmentation, reporting, and database integration. **Chapter 5** discusses the evaluation of the developed system and its deployment through a Streamlit dashboard. **Chapter 6** presents system testing, challenges, and evaluation against objectives. Finally, **Chapter 7** concludes the report and provides recommendations for future improvements and clinical deployment.

Chapter 2

Literature Review

This chapter presents a comprehensive review of the key technologies, existing systems, and related studies relevant to computer-aided lung cancer detection. The chapter begins by exploring foundational technologies that form the backbone of this project—specifically transfer learning, the EfficientNetB1 architecture, and the Fuzzy C-Means (FCM) segmentation algorithm—highlighting their principles, advantages, and suitability for medical image analysis. Following this, an in-depth review of existing lung cancer detection and segmentation systems is provided, including public tools, dashboards, and research prototypes. The strengths and limitations of these systems are analyzed to identify technological gaps. Finally, related works from scholarly research are examined, and a critical comparison is made to underscore how the proposed system advances beyond prior efforts. This review serves to justify the design choices of the project and to position its contributions within the broader research landscape.

2.1 Review of the Technologies

One of the deadliest types of cancer in the world is lung cancer, and increasing patient survival depends on early detection. Conventional techniques that depend on human interpretation of CT scans are frequently laborious and prone to mistakes. Deep learning in medical imaging is leading to the development of computer-aided diagnostic systems that are more accurate and efficient. This section reviews the key technologies used in this project—namely, transfer learning with EfficientNetB1 for classification, and Fuzzy C-Means (FCM) and its variants for lung tumor segmentation.

2.1.1 Transfer Learning in Medical Image Analysis

Transfer learning (TL) has emerged as a very common strategy for medical image analysis in alleviating some of the drawbacks of training deep learning models from scratch, especially on small datasets. Here, pre-trained Convolutional Neural Networks (CNNs) that have already

learned feature representations on large-scale datasets are adapted to novel, domain-specific tasks like lung cancer classification. This adaptation can be either achieved by unfreezing and fine-tuning the selected layers of the pre-trained model or by adding additional layers and training the model from end to end. These processes enable the model to automatically learn the more effective and task-specific features, leading to better classification performance [8].

The utility of transfer learning in lung cancer identification has been shown to be effective in a number of studies. For example, utilized deep residual networks (ResNets) and fine-tuned them on a dataset of lung cancer to classify different pathological types of lung cancer with great precision [9]. Likewise, [10] employed deep transfer learning methods to fine-tune pre-trained CNN models for the detection of lung cancer from CT images with encouraging outcomes. These works demonstrate the strength of transfer learning in exploiting pre-learned knowledge from general image recognition tasks and transferring it to more specialized medical fields, with the effect of enhancing model robustness, accuracy, and training efficiency.

2.1.2 EfficientNetB1: Architecture, Scaling, and Performance

Google AI's research team in 2019 presented a family of models EfficientNetB0 to EfficientNetB7 that soon became renowned for their enhanced performance and efficiency on most computer vision tasks [11]. EfficientNet, as an optimized and scalable architecture, has continually outperformed numerous state-of-the-art CNNs including Inception-V3, ResNet-50, Inception-ResNetV2, and Dense Convolutional Network (DenseNet) on benchmarks including ImageNet classification, segmentation tasks, and transfer learning tasks.

EfficientNet is based on a mobile-sized baseline network called EfficientNet-B0, proposed by [11]. The baseline network was found by multi-objective neural architecture search (NAS) that optimizes both accuracy and computational efficiency (FLOPS). The search aims to balance both metrics with a compound function, leading to a very efficient model. EfficientNet-B0 uses mobile inverted bottleneck convolution (MBConv) blocks and squeeze-and-excitation (SE) mechanisms and is somewhat larger than MnasNet as it uses a larger target floating point operations per second (FLOPS).

To scale EfficientNet-B0 to larger, more powerful models, [11] suggested a two-step compound scaling approach. They initially performed a small grid search to obtain the best scaling coefficients— $\alpha = 1.2$ (depth), $\beta = 1.1$ (width), and $\gamma = 1.15$ (resolution)—under the constraint $\alpha \cdot \beta^2 \cdot \gamma^2 \approx 2$, which gives a balanced scaling of the model dimensions. Then, for a given user-specified φ value, they scaled the base model to larger models named EfficientNetB1 to B7, each with increased performance and no loss of efficiency.

EfficientNet training involves advanced optimization techniques to further enhance performance. These include the Root Mean Square Propagation (RMSProp) optimizer, Swish (SiLU) activation function, AutoAugment for data augmentation, stochastic depth, and scaled dropout regularization from 0.2 for B0 to 0.5 for B7. These strategies are significant to help the model generalize well while avoiding overfitting.
EfficientNets are highly suitable to process medical images since they have the capacity for representative and informative features without trading off the model size versus model performance. These EfficientNets with pre-trained weights are improved by transfer learning such that feature learning can be mapped from large-scale datasets (e.g., ImageNet) to target domain-specific tasks such as lung cancer classification and segmentation [12]. EfficientNet models, when trained on medical image data, have been found to generalize to subtle anatomy differences, thereby improving classification accuracy as well as segmentation performance.

Additionally, EfficientNet models' computational efficiency and memory optimization render them extremely appropriate for clinical environments where processing time and resource constraints are issues of paramount importance [13]. All of these advantages render EfficientNet a better option as a backbone for building intelligent diagnostic systems in medicine.

EfficientNetB1, a scaled-up version of EfficientNetB0, specifically, offers a remarkable accuracy-computation trade-off. With just 7.8 million parameters and 0.7 billion FLOPS, it reaches around 79.1% top-1 accuracy on ImageNet, surpassing larger models such as ResNet-152 and Inception-V3, as shown in figure 2.1.1. Further, real hardware experiments verify that EfficientNetB1 is 5.7 times faster than ResNet-152 on a standard Central Processing Unit (CPU) for inference, making it highly adequate for real-time applications in figure 2.1.2.

Model	Top-1 Acc.	Top-5 Acc.	#Params	Ratio-to-EfficientNet	#FLOPs	Ratio-to-EfficientNet
EfficientNet-B0	**77.1%**	**93.3%**	**5.3M**	**1x**	**0.39B**	**1x**
ResNet-50 (He et al., 2016)	76.0%	93.0%	26M	4.9x	4.1B	11x
DenseNet-169 (Huang et al., 2017)	76.2%	93.2%	14M	2.6x	3.5B	8.9x
EfficientNet-B1	**79.1%**	**94.4%**	**7.8M**	**1x**	**0.70B**	**1x**
ResNet-152 (He et al., 2016)	77.8%	93.8%	60M	7.6x	11B	16x
DenseNet-264 (Huang et al., 2017)	77.9%	93.9%	34M	4.3x	6.0B	8.6x
Inception-v3 (Szegedy et al., 2016)	78.8%	94.4%	24M	3.0x	5.7B	8.1x
Xception (Chollet, 2017)	79.0%	94.5%	23M	3.0x	8.4B	12x
EfficientNet-B2	**80.1%**	**94.9%**	**9.2M**	**1x**	**1.0B**	**1x**
Inception-v4 (Szegedy et al., 2017)	80.0%	95.0%	48M	5.2x	13B	13x
Inception-resnet-v2 (Szegedy et al., 2017)	80.1%	95.1%	56M	6.1x	13B	13x
EfficientNet-B3	**81.6%**	**95.7%**	**12M**	**1x**	**1.8B**	**1x**
ResNeXt-101 (Xie et al., 2017)	80.9%	95.6%	84M	7.0x	32B	18x
PolyNet (Zhang et al., 2017)	81.3%	95.8%	92M	7.7x	35B	19x
EfficientNet-B4	**82.9%**	**96.4%**	**19M**	**1x**	**4.2B**	**1x**
SENet (Hu et al., 2018)	82.7%	96.2%	146M	7.7x	42B	10x
NASNet-A (Zoph et al., 2018)	82.7%	96.2%	89M	4.7x	24B	5.7x
AmoebaNet-A (Real et al., 2019)	82.8%	96.1%	87M	4.6x	23B	5.5x
PNASNet (Liu et al., 2018)	82.9%	96.2%	86M	4.5x	23B	6.0x
EfficientNet-B5	**83.6%**	**96.7%**	**30M**	**1x**	**9.9B**	**1x**
AmoebaNet-C (Cubuk et al., 2019)	83.5%	96.5%	155M	5.2x	41B	4.1x
EfficientNet-B6	**84.0%**	**96.8%**	**43M**	**1x**	**19B**	**1x**
EfficientNet-B7	**84.3%**	**97.0%**	**66M**	**1x**	**37B**	**1x**
GPipe (Huang et al., 2018)	84.3%	97.0%	557M	8.4x	-	-

Figure 2.1.1 Performance Results of EfficientNet on ImageNet

	Acc. @ Latency		Acc. @ Latency
ResNet-152	77.8% @ 0.554s	GPipe	84.3% @ 19.0s
EfficientNet-B1	78.8% @ 0.098s	EfficientNet-B7	84.4% @ 3.1s
Speedup	**5.7x**	**Speedup**	**6.1x**

Figure 2.1.2 Comparison of Inference Latency

Overall, thanks to its light-weight structure, high precision, and speedy inference, EfficientNetB1 performs excellently on transfer learning tasks, especially on medical image classification tasks like distinguishing lung cancer from CT images. With its low resource consumption and strong generalization capability, it can be an excellent backbone model in clinic deployment scenarios where timely and accurate diagnosis is vitally important.

2.1.3 Fuzzy C-Means (FCM) for Medical Image Segmentation

The Fuzzy C-Means algorithm evolved from the work [23] in the late 1970s and early 1980s, topping off in his classic book Pattern Recognition with Fuzzy Objective Function Algorithms (1981). Unlike hard clustering algorithms such as k-means, where each point is assigned to a single cluster, FCM introduced the concept of fuzzy partition where every point belongs to multiple clusters with varying membership grades. This was later formalized by a fuzzy objective function that minimized within-cluster distance and maximized between-cluster separation. Addition of a fuzzification parameter allowed control of the "softness" of cluster boundaries, and FCM thereby became especially effective for applications where class boundaries are ill-defined [23].

The fundamental principle of the Fuzzy C-Means (FCM) algorithm is to minimize a fuzzy objective function, i.e., weighted sum of squared distances from cluster centers to pixels, with weights as the degree of membership of each pixel. The algorithm begins with an arbitrarily initialized membership matrix and iterates step-by-step by initially computing cluster centers as weighted averages of the data points. Membership values are recalculated using the inverse distance to each of the cluster centers, with correction through the fuzzification parameter m to control the softness of clustering. These two steps—recomputing cluster centers and reassigning memberships—are repeated until the change in the membership matrix is less than a threshold value, which suggests that convergence has occurred. By this iterative process, each pixel attains partial membership in multiple clusters, thus making FCM capable of capturing overlapping or fuzzy boundaries, as typically seen in medical scans [23].

Medical imaging often involves inherent uncertainty, noise, and intensity inhomogeneity, particularly in modalities such as CT and MRI. FCM penetrated medical image segmentation from the time its fuzzy membership approach could handle fuzzy borders better compared to crisp methodologies. Brain MR images were first taken up, proceeding to lung CT scans where tumor and nodule boundaries are often indistinct. By the 2000s, FCM was one of the most popular unsupervised cluster algorithms applied to medical segmentation tasks due to its ability to deal well with uncertainty [14].

In tumor segmentation, FCM is useful by capturing fuzzy interfaces between tumor and normal tissue. Studies [14] and [17] established that FCM successfully segmented lung nodules from CT scans without requiring annotated training images. The algorithm could separate regions of interest (tumors) from the surrounding tissues by making use of differences in intensity despite the presence of artifacts and noise. However, its spatial information absence and noise sensitivity remain the limitations, often the source of spurious positives unless it is used in combination with morphological processes or spatially constrained like FRFCM (Fast and Robust FCM).

The completeness of FCM has led to applications in a wide range of medical segmentation applications. These include brain tumor segmentation, lesion identification in multiple sclerosis, breast cancer screening, and lung tumor segmentation. In lung CT imaging alone, FCM has been used both for solitary pulmonary nodules and diffuse tumor detection. As an unsupervised method, the requirement for large annotated datasets is removed, hence being desirable for clinical settings where labelling is costly and time-consuming. Moreover, hybrid approaches have emerged where FCM is blended with neural networks, SVMs, or morphological processing to improve accuracy and stability in clinical practice [14].

In sum, Fuzzy C-Means has established itself as an innovative clustering algorithm for the application area of medical image segmentations with the ability to model uncertainty as well as capture blurred edges. Despite its noisy sensitivity and blindness for the spatial information which indicate its weakness, the algorithm continues to grow with its new variants such as FRFCM and combined methodology using fuzzy/clustering with machine learning or morphological algorithms. These newer expansions equip FCM with its long-term value as well as flexibility for addressing the intrinsical complex issues of tumor segmentations as well as the entire medical imaging application.

2.2 Review of the Existing Systems

2.2.1 Existing System A: Lung Cancer Segmentation [26]

This Kaggle-based system [26] provides a two-stage deep learning pipeline for lung cancer segmentation and detection from CT scans. The system begins with a classification stage using a pre-trained VGG16 model that has been fine-tuned to perform binary classification—predicting whether a given CT scan is diagnosed with lung cancer or not. The model achieved a test accuracy of approximately 80%. If the prediction probability exceeds a given threshold value (normally 0.5), the image is passed to the segmentation stage. For segmentation, a U-Net model is employed to create a pixel-wise binary mask that identifies the areas of the potential tumor. The resultant mask is resized to be the same size as the input image and superimposed on the CT scan to visually indicate the area of the detected tumor. As shown in Figure 2.2.1, the system generates a side-by-side comparison of the original CT scan and segmented tumor regions (highlighted in red). Although it integrates simple classification and segmentation steps nicely, it lacks subtype classification, morphological postprocessing, and clinical feature extraction.

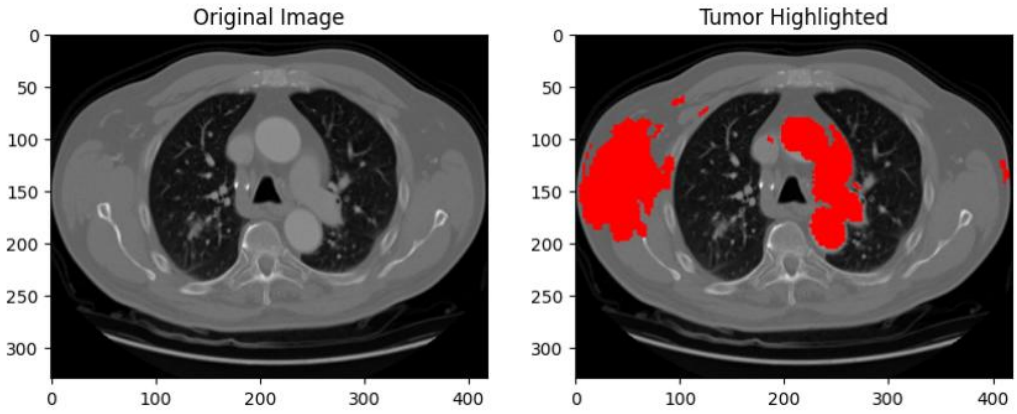


Figure 2.2.1 Tumor segmentation output —original CT scan (left) and tumor region highlighted (right) [26]

2.2.2 Existing System B: Lung Cancer Streamlit Application [27]

This Streamlit platform [27] combines two methods of lung cancer detection: one is machine learning-based using structured clinical data, and the other is Convolutional Neural Network (CNN) based on CT images. The platform, as indicated in Figure 2.2.2a and 2.2.2b, supports users either uploading CT images or entering clinical factors such as age, smoking, passive smoking, fatigue, diet, and others. The structured data is passed through a variety of ML models—Support Vector Machines, Logistic Regression, Decision Tree, and K-Nearest Neighbors. The models are selectable through the sidebar, and best-performing models (SVM, Decision Tree, KNN) are each 100% accurate on their individual test sets, while Logistic Regression comes in at 95.5%. A prediction confidence score and chosen ML model are clearly displayed after user input.

The CNN module takes user-uploaded CT images and produces a classification output like "Normal" or "Lung Cancer" with confidence (e.g., 99.06% normal), making it visually accessible for end-users. Although this arrangement has an intuitive interface with explicit result panels, it does not include essential features like tumor segmentation, subtype classification (e.g., adenocarcinoma vs squamous cell), or structural reporting output like lesion measurement or lobe location. Therefore, while excellent as a demo, it possesses shallow applicability to actual clinical decision support systems.

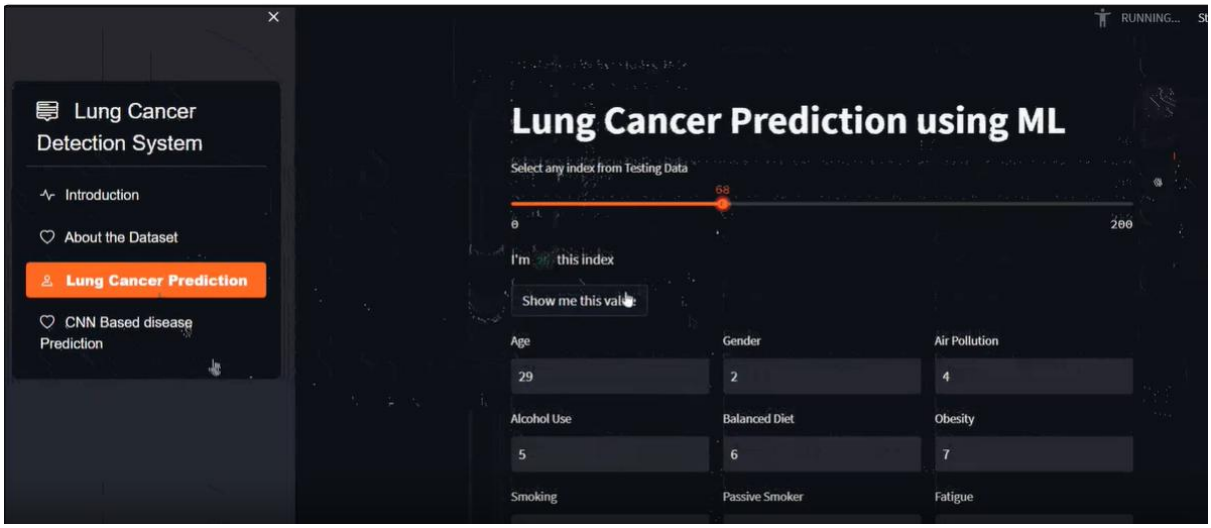


Figure 2.2.2a Lung Cancer Prediction using Structured Clinical Data with ML Models

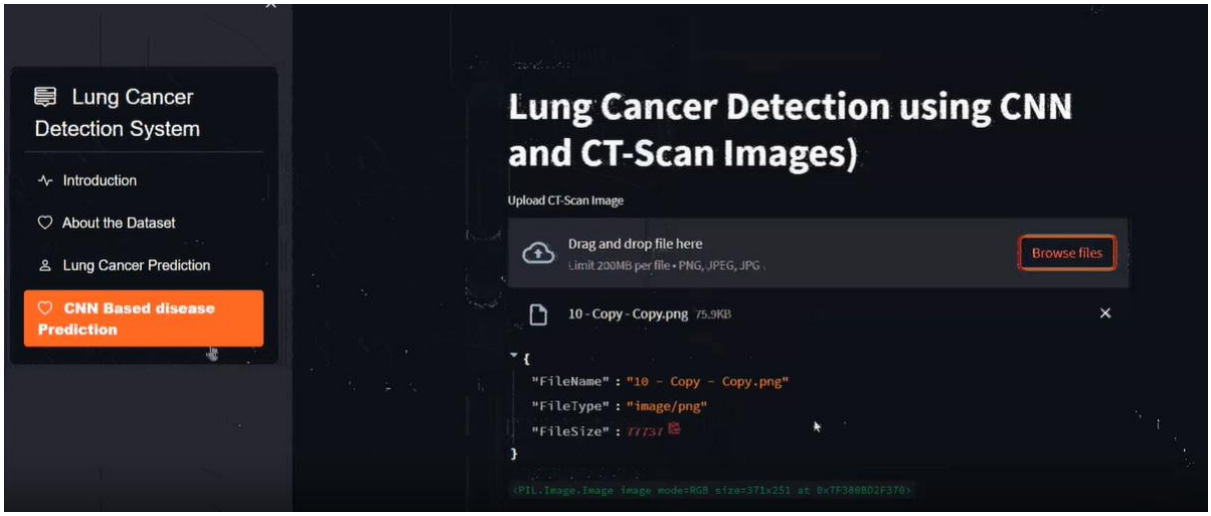


Figure 2.2.2 b Cancer Prediction Interface using CNN-Based Image Classification in Streamlit Application

2.2.3 Existing System C: U-Net Segmentation on Lung Tumor [28]

The GitHub-based system [28] that provides a completely automated segmentation system to identify lung tumors from CT scans using deep learning. The project, developed from the Medical Segmentation Decathlon dataset, employed a traditional U-Net architecture trained on 2D slices to generate pixel-level tumor masks. The system includes clearly defined data preprocessing steps, grayscale conversion, resizing, normalization, and train-validation split. It is trained using binary cross-entropy loss with oversampling methods for handling tumor-background class imbalance. The final output of the model is a binary segmentation mask that is thresholded and overlaid on the original CT slice for visualization of tumor areas. The system achieves a Dice score of 0.878 on test images. Although the pipeline demonstrates nice tumor localization and mask generation, it doesn't include morphological postprocessing (e.g., noise removal or shape filtering) and lacks a classification module for tumor types or malignancy grades. The segmented output of U-Net model, as shown in Figure 2.2.3, outlines the tumor region in red overlay on the original CT scan.

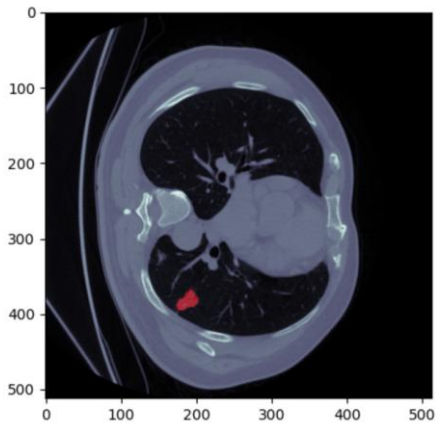


Figure 2.2.3 Segmented region of lung tumor using U-Net model with MSD dataset training

2.2.4 Existing System D: Interactive Lung Cancer Detection System [30]

This Streamlit web application offers a simple interface for predicting the probability of having lung cancer based on standard clinical and lifestyle input data like age, gender, smoking, fatigue, yellow fingers, worry, chronic diseases, and chest pain. Users can select from various machine learning models (e.g., Decision Tree, Logistic Regression, KNN), and get output like "Low chances of Lung Cancer" according to their input as shown in figure 2.2.4a. The system also indicates the decision-making process with interactive decision trees and reports train/test accuracy numbers for reasons of transparency as shown in figure 2.2.4b. Although this system is better in terms of usability and ease of use, it lacks scope. It is unable to interpret CT images, segment the tumor, or classify cancer subtypes. It also does not offer interpretation functionalities like lesion overlays, morphological analysis, or structured clinical reporting (like PDF or AIM XML reports). Compared to image-based CAD systems, it cannot visualize or localize tumors and is therefore less useful in clinical radiology settings where spatial context is essential. Thus, it is more of a risk screening tool than a diagnostic tool.

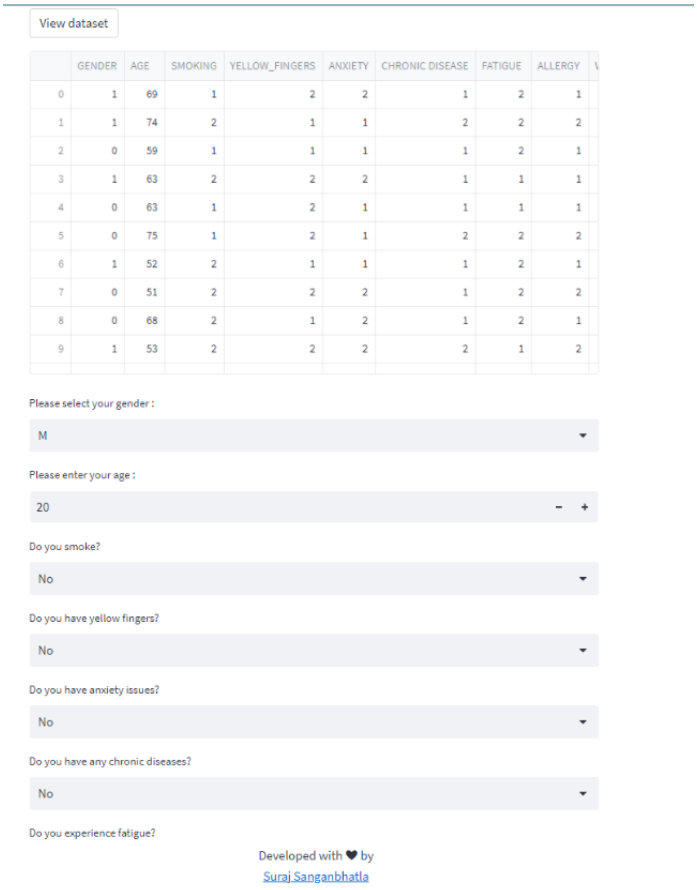


Figure 2.2.4a Lung Cancer Detection System Interface

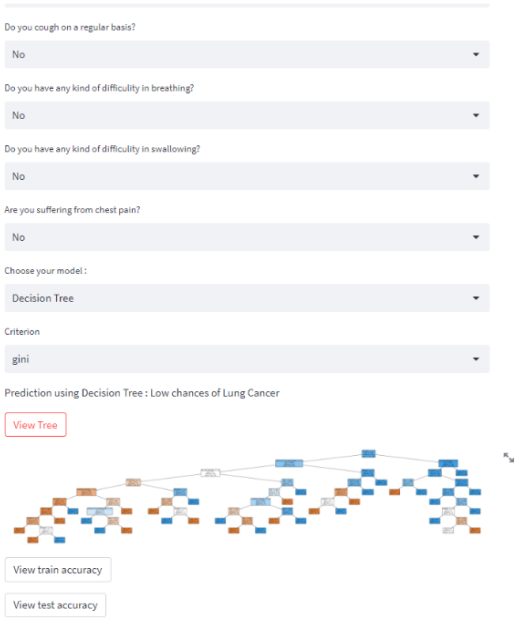


Figure 2.2.4b Results of the classification model

2.2.5 Existing System E: Lung Infection Detection App [31]

This Streamlit program takes input of a user uploading CT scan images and returning a result of lung disease classification, such as "Type 2 disease." As demonstrated by Figure 2.2.5, the user interface is minimalistic and easy to use, allowing for effortless image uploads using drag-and-drop functionality. Once an image has been uploaded, the system processes the scan using a pre-trained deep learning model and presents the prediction in addition to the uploaded image.

Yet, system output is restricted to a single label for a disease type with no confidence scores or explainability metrics. There are no tumor segmentation, measurement tools, subtype classification (e.g., adenocarcinoma), nor formalized output formats such as XML or PDF. The system architecture and training are not published, precluding reproducibility and transparency. Therefore, even though the tool shows embryonic classifying capability, the tool is not equipped with the multi-output, modular, and clinically interpretable characteristics necessary in actual diagnostic assistance.

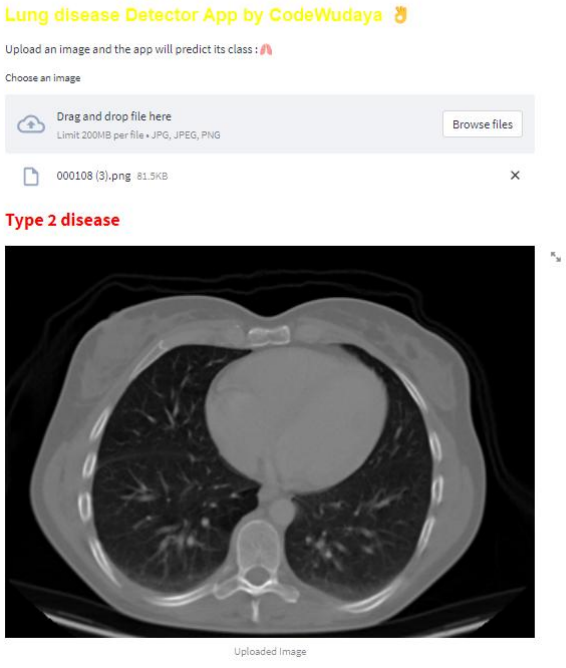


Figure 2.2.5 Screenshot of the Lung Infection Detection App showing CT image upload and "Type 2 disease" prediction.

2.2.6 Existing System F: DeepHealth Lung (Aidence / DeepHealth) [32]

DeepHealth Lung, previously Veye Lung Nodules by Aidence, is a CE-marked AI solution that is integrated within radiology workflows for computer-aided detection, monitoring, and reporting of pulmonary nodules on CT scans. It is currently deployed across multiple clinical settings, including more than 40 NHS Trusts within the United Kingdom under the Targeted Lung Health Checks initiative. The system offers computerized nodule segmentation, volumetry, volume doubling time (VDT) computation, and formatted reporting straight into PACS, assisting the radiologists in being consistent and decreasing reporting times by up to 50% [32].

The system's strengths lie in its robust clinical integration, its ability to handle large-scale CT volumes, and workflow efficiency orientation. As shown in Figure 2.2.6, Radiologists are supported by features like prior scan comparison, nodule growth alerts, and guideline-based structured output (e.g., Fleischner or BTS). Above all, DeepHealth Lung does not even need clinicians to upload images manually because it is plugged into hospital IT systems.

As solid as it is, DeepHealth Lung is primarily focused on nodule detection and tracking and not for histological subtype classification (e.g., adenocarcinoma, squamous cell carcinoma). It segments all the scans, cancer and non-cancer, which could cause redundant computation in the case of normal scans. It also is not open-source and a commercial product to be licensed for use, and it leaves very little room for its underlying system or AI model interpretability.

Figure 2.2.6 Example of DeepHealth Lung Reporting Interface

2.3 Related Works

Lung cancer remains one of the world's most fatal cancers, and accurate detection and segmentation from CT scans are vital for improving diagnostic results. Early attempts brought together conventional feature extraction coupled with machine learning classification. Lobo and Guruprasad [17] introduced one of the first combined methods, a two-stage conventional method for lung cancer detection that used fuzzy C-Means (FCM) for segmentation and Support Vector Machines (SVM) for classification. Using the Gray-Level Co-occurrence Matrix (GLCM), their system first extracted texture features such as contrast, correlation, and energy. The SVM then used these features to classify CT images as either cancerous or non-cancerous. FCM was used to locate the tumour if it was determined to be cancerous. This technique struggled to accurately isolate tumours and lacked the flexibility and depth found in contemporary deep learning architectures, despite producing a moderate accuracy of 79.17% and displaying distinct segmentation masks.

To address the limitations in cluster shape modeling, Afshar et al. [24] recognized the inherent drawback of FCM and K-means in forcing regular cluster shapes on tumors. They proposed an improved segmentation pipeline that first separated the lung region from the background through snake optimization, then employed Gustafson-Kessel (GK) clustering to cater to tumors' irregular shapes. Quantified on CT scans in the LIDC dataset using entropy analysis, their method consistently outperformed FCM and K-means, with two-cluster GK clustering providing the most consistent tumour contouring. the reliance on semi-automatic snake initialization limited the system’s full automation, and the study acknowledged that the development of the framework into three-dimensional modeling and automatic contour detection would be necessary for clinical use in practice. Despite these restrictions, the research demonstrated how clustering improvements could appreciably boost the robustness of tumour segmentation and pave the way for more advanced automated techniques.

Similarly, Dhaware and Pise [25] developed feature-based classification. Their system enhanced CT lung images using Contrast Limited Adaptive Histogram Equalization (CLAHE), used twelve texture features derived from GLCM and seven shape-based features, and then used sequential forward selection (SFS) for feature reduction. A Bayesian classifier was used to classify normal and abnormal cases, and after an image had been classified as abnormal,

FCM segmentation was applied for tumour segmentation. This pipeline delivered higher classification accuracy compared to standard methods, demonstrating the ability to combine statistical texture analysis with probabilistic classification. Nevertheless, as with the earlier methods, it was limited by the reliance on manually engineered features and standalone segmentation procedures, which limited scalability to larger collections and more complex clinical scenarios.

In the context of the literature, most of the traditional lung nodule detection pipelines typically follow a segmentation-first strategy, where the lung region is segmented before candidate nodule extraction and classification. These pipelines mostly use hand-engineered features (e.g., shape, texture, intensity) and machine learning classifiers such as SVM, Random Forest, or KNN. Image preprocessing methods like CLAHE, Gaussian, and Wiener filtering for improving image clarity are followed by segmentation methods like thresholding, active contours, or fuzzy connectivity. While high sensitivity has been obtained in some settings, these approaches do not generalize well across datasets and struggle with small or irregular nodules. False positives are a significant problem, usually dealt with by an extra classification stage [29].

Researchers like Song et al. [15] shifted from handcrafted features to convolutional neural networks (CNNs) as deep learning began to take over the medical imaging space. In order to automatically learn spatial and textural features for benign-malignant classification, their work trained CNNs directly on lung nodule patches from CT images. This was a major change, increasing automation and decreasing human bias in feature extraction. In addition to increasing classification accuracy, the CNN-based framework provided a repeatable technique with publicly accessible datasets. However, segmentation and tumour localization—two crucial steps for accurate treatment planning—were not covered in the study; it only concentrated on classification.

For more effective spatial modeling and tumour localization, later systems developed with 3D CNNs and hybrid architectures. Such advanced networks possessed the ability of handling volumetric data and learning context-aware features, which resulted in better segmentation accuracy. The majority of these models continued to adopt the segmentation-first approach—using U-Net or its variants to generate tumour masks before classification. For example, in

architectures like Faster R-CNN, a Region Proposal Network (RPN) segments candidate regions first, followed by classification. Multi-tasking networks such as NoduleNet go one step further and unify segmentation and classification within one U-Net backbone for end-to-end training and implicit false positive reduction [29].

Later, multi-task architectures gained more popularity with increasing GPU power and the availability of larger annotated medical datasets. Humayun et al. [16] proposed a better hybrid framework consisting of a U-Net model with various backbones—VGG16, ResNet50, and Xception—for correct lung region segmentation, in combination with a Binary Image Representation (BIR)-enhanced CNN for predicting the severity of lung injury. The setup had incorporated contrast enhancement (CLAHE), normalization, and resizing during preprocessing, and the performance was evaluated using metrics such as IoU, Dice coefficient, F1 score, and accuracy. The best performance was done by VGG16-based U-Net backbone with a segmentation accuracy of 98.36% and Dice score of 0.9363. The BIR-enhanced VGG16 CNN performed best in classification with an accuracy of 97.83%. Though computationally demanding, the model demonstrates great promise in dual-task medical image analysis, most notably in accurate delineation of cancerous regions and ranking the severity of injury.

Lightweight, high-performance architectures that strike a balance between accuracy and efficiency have become more popular in recent years. For example, Lung-EffNet, a deep learning pipeline that uses a refined EfficientNetB1 model to classify lung cancer types from CT images—normal, adenocarcinoma, large cell carcinoma, and squamous cell carcinoma—was proposed by Raza et al. [1]. The model, which was trained on a 1,000-image dataset from Kaggle, produced high F1-scores for the majority of classes and a 94% test accuracy; however, because of its visual ambiguity, large cell carcinoma was still more challenging to classify. The need for more integrated pipelines was highlighted by the fact that, despite its effectiveness in classification, this method did not extend to segmentation or structured annotation.

Briefly, lung cancer detection models have evolved from handcrafted feature-based pipelines towards highly coupled deep learning models. Despite this, the conundrum remains on how to balance accuracy, interpretability, and usability by clinicians. Our current work is a contribution towards such trends in terms of proposing a modular framework that combines

EfficientNetB1 as the classifier and FCM-based segmentation with the aim of obtaining lightweight, interpretable, and scalable performance for real-world clinical environments.

2.4 Critical Remarks of Previous Works and Existing Systems

Lobo and Guruprasad [17] presented one of the initial hybrid pipelines for detecting lung cancer that included fuzzy C-Means (FCM) segmentation and Support Vector Machine (SVM) classification. The system was successful in utilizing Gray-Level Co-occurrence Matrix (GLCM) features such as contrast and energy, and demonstrated the efficacy of feature extraction-classification synergy. However, the approach was heavily reliant on hand-crafted features and separate processing steps. The FCM segmentation was inaccurate for irregular tumor edges and the low classification accuracy rate of 79.17% reflected the model's failure to scale and adapt as well as existing deep learning architectures.

Afshar et al. [24] enhanced standard clustering approaches with a segmentation-first pipeline that applied snake optimization followed by Gustafson-Kessel (GK) clustering. The approach was better suited to model irregular tumor shapes and achieved greater contour consistency on the LIDC dataset. The pipeline, however, was semi-automatic in that it needed snake initialization, limiting full automation. The pipeline also lacked integration of classification modules as well as advancing to 3D modeling, hence not achieving clinical deployment levels.

Dhaware and Pise [25] proposed a statistical image analysis pipeline that incorporated CLAHE-based preprocessing, GLCM texture features, and shape features for Bayesian model-based classification. Sequential forward selection (SFS) was utilized to carry out feature dimensionality reduction, and FCM was utilized to segment subsequent to classification. Even though this approach improved accuracy in classification, the implementation of handcrafted features and isolated segmentation limited it to be utilized in real-time and big datasets or in clinical practice.

Continuing to deep learning, Song et al. [15] proposed the use of Convolutional Neural Networks (CNNs) for automatic spatial and textural feature learning from lung CT patches without requiring manual feature design. This improved classification accuracy and reproducibility with public datasets. However, the research was restricted to classification only

without segmentation and tumor localization—essential for clinical interpretation and treatment planning. Reduction of false positives and explainability of the model were not considered.

Humayun et al. [16] proposed a multi-task deep learning pipeline based on a U-Net segmentation backbone integrated with a Binary Image Representation (BIR)-enhanced CNN for severity prediction. They tried out different encoder backbones—VGG16, ResNet50, and Xception—with excellent segmentation performance (Dice score of 0.9363) and classification accuracy (97.83%). While the structure was high in accuracy, its high computational cost may prove to be a constraint when deployed to low-resource settings. The structure also did not incorporate subtype classification and multimodal clinical feature fusion.

Raza et al. [1] suggested a light-weight EfficientNetB1-based classifier for the detection of four lung cancer types from CT scans. The model was good in terms of overall accuracy (94%) and F1-scores and would therefore be suitable for real-time deployment. Nevertheless, it did not have a segmentation module integrated into it and therefore wasn't as interpretable in case of clinical diagnosis. The system also wasn't good at identifying visually confusable tumor types like large cell carcinoma and had unstructured annotation outputs.

Existing System A [26] is a two-step deep learning pipeline of Kaggle based on U-Net segmentation and VGG16-based classification. It gives a tumor mask on the CT scan with a visual indication of where the tumor is, providing a rough estimate of tumor location. While this system has an end-to-end process from localization to prediction, it possesses only moderate performance (~80%) with no subtype classification, morphological postprocessing, and structured output for clinical use.

Existing System B [27], a Streamlit-based application, integrates structured clinical information with image-based CNN classification. It had excellent test accuracy for several machine learning algorithms like SVM, Decision Tree, and KNN (all 100%). The CNN also performed well with a validation accuracy of 94.5%. Although it has the luxury of being dual-mode input, the system does not include segmentation, spatial tumor localization, and clinical interpretability. The 100% test scores are also questioned for overfitting and lack of external validation.

Existing System C [28], starting from the Medical Segmentation Decathlon dataset, offers an end-to-end computerized U-Net segmentation pipeline for lung tumor detection. It has satisfactory preprocessing and class imbalance control with a high Dice score of 0.878. However, it is limited to simple segmentation without any classification step or morphological postprocessing like shape filtering or noise removal. Web interface or real-time prediction capability is also lacking in it.

The system D [30] is highly interactive and easy to use with its Streamlit user interface and provides prediction results from a vast array of clinical and lifestyle factors like smoking, fatigue, chronic illness, and anxiety. It comprises support for various machine learning algorithms (like Decision Tree, KNN) and decision explanation via decision trees. This ease makes it appropriate for general screening and public health education applications. However, it lacks image-based diagnosis and thus no spatial context, tumor localization, or visual interpretability of disease. It lacks segmentation, subtype classification, or medical imaging. Its applicability in radiology or clinical diagnosis is limited given the absence of CT scan input support or structured output formats.

For system E [31], the tool has a simple drag-and-drop interface for uploading CT images, with instant feedback of class prediction output. It reduces the barrier to participation for users looking for rapid image-based predictions and provides a feeling for how much is possible from a deep learning model in visual classification problems. However, output from prediction is restricted to a general disease label (e.g., "Type 2 disease") without subtype differentiation, segmentation mask, confidence measure, or explainability. Lesion size and shape reporting is not offered by the system, nor structured diagnostic reporting in PDF or XML format. Data used for training as well as model architecture are also not made available, thus the system is neither reproducible nor clinically reliable.

Lastly, system F [32] stands out with its full integration into hospital PACS and diagnostic workflows. It supports automated nodule detection, volumetry, VDT calculation, and guideline-driven follow-up recommendation. The interface allows radiologists to compare prior scans, assess lesion growth, and access structured output consistent with clinical protocols. Although it offers very good detection and tracking, it is not classifying cancer into

histological subtype or making inferences about aggressiveness of cancer. However, it does not possess conditional segmentation reasoning (i.e., always segments all the scans) and can over-use resources on non-pathological cases. It is a closed-source commercial package with little transparency or user modification.

A summarized comparison of these previous works and existing systems and the proposed solution is presented in Table 2.4.

Reference/System	**Techniques**	**Strengths**	**Weaknesses**	**Proposed Solution**
Lobo & Guruprasad [17]	FCM + SVM (Traditional ML)	Early hybrid approach; utilized GLCM features	Hand-crafted features; weak FCM segmentation; relatively good SVM accuracy	Replaces SVM with high-accuracy EfficientNetB1; improves FCM with spatial filters, morphology, and heuristics
Afshar et al. [24]	GK Clustering + Snake Optimization	Performed well on irregular tumor shapes	Not fully automated; no classification; lacks 3D or real-time use	Introduces fully automated pipeline of classification and segmentation; no manual initialization
Dhaware & Pise [25]	CLAHE + GLCM + SFS + Bayesian Classifier	Combined texture + shape features; probabilistic output	No real-time use; only post-classification used; low generalizability	Uses a EfficientNetB1 for generalizable classification; segmentation only for positive cases in order to save computation

Song et al. [15]	CNN on CT nodule patches	Lung nodule binary classificat ion that works well and has good spatial and textural learning	No segmentation or localization; no explainability	Adds tumor localization using modified FCM; outputs overlaid masks and tumor centroids for interpretation
Humayun et al. [16]	U-Net + BIR-enhanced CNN	Dual-task framework; high accuracy & Dice score	More GPU-intensive requirement; no subtype or structured reporting	Offers a lightweight pipeline; segmentation only after classification; XML export for review integration
Raza et al. [1]	EfficientNetB1 classifier	94% classificat ion accuracy, good generalisa tion, and fine tuning on actual CT scans	No segmentation or localization; no structured o utput	Adds segmentation step using modified FCM to provide tumor localization and boundary masks
Existing System A [26]	VGG16 + U-Net	Easy and interpretable; pixel-wise segmentation	Poor classification accuracy (80%); lacks subtype or morphology	Enhances classification using EfficientNetB1 (94.6%); incorporates advanced postprocessing to segmentation
Existing System B [27]	ML (SVM/KNN) + CNN in Streamlit	High test accuracy; integrates	No tumor segmentation or class label by	Adds segmentation and interpretable outputs (mask +

		clinical + imaging data	subtype; risk of overfitting	XML); allows strong CT-based inference
Existing System C [28]	U-Net on 2D slices (MSD)	High Dice score; fully automatic	No classification; no postprocessing; no UI/web deployment	Adds classification module (EfficientNetB1); adds support for postprocessing and XML export and integrated with Streamlit application
Existing System D [30]	Streamlit + ML (Decision Tree, SVM,KNN,etc.)	Simple UI, risk prediction from clinical/lifestyle data; model interpretability via decision tree	No image analysis; no tumor localization or segmentation; no subtype classification or structured output	Adds image-based CT analysis, subtype classification, tumor segmentation, structured PDF & AIM XML report generation
Existing System E [31]	CT-based Image Classifier in Streamlit	Fast prediction UI; simple deployment	Lacks segmentation, subtype classification, interpretability, structured reporting	Adds subtype classification (EfficientNetB1), FCM-based segmentation, morphological postprocessing, report & XML export in integrated Streamlit dashboard

DeepHealth Lung [32]	Nodule Detection Volumetry VDT	Real clinical deployment; PACS integration; growth tracking; structured reports	No subtype classification; no conditional segmentation; commercial license required	Adds subtype classification (EfficientNetB1); conditional segmentation (FCM); lightweight, interpretable dashboard

Table 2.4 Critical Comparison of Previous Works and Existing Systems with the Proposed Solution

Chapter 3
System Methodology/Approach OR System Model

This chapter outlines the methodology adopted for the development of the proposed lung cancer detection system. The project applies the Cross-Industry Standard Process for Data Mining (CRISP-DM) methodology, a structured yet adaptable process in six iterative stages: business understanding, data understanding, data preparation, modelling, evaluation, and deployment. Each stage has been tailored to address the requirements of developing a light and modular classification-first pipeline that combines deep learning for cancer detection with segmentation for tumour localisation based on clustering. In addition, system analysis and design methodologies, including use case diagram and activity diagram, are presented to represent user interactions and the overall system flow. Together, these form an overall methodology that guides both technical implementation and practical usability of the proposed solution.

3.1 CRISP-DM Framework

3.1.1 Business Understanding

To achieve a clear business understanding for this project, we will begin by discovering background of the project in an effort to establish the clinical and technical context for the detection of lung cancer. This will involve review of the existing problems in diagnosis, in particular the reliance on CT scans and limitations of radiologist manual interpretation. After developing this context, we will define precisely what are the problems that need to be solved, such as the inefficiencies of existing CAD pipelines which are only focusing on classification or repeat segmentation on standard scans.

After establishing the main issues, we will then move on to describe the motivation of the project. This will be focused on the need for a light, modular, and clinically feasible solution that mirrors true real-world workflows and optimizes computational performance by allowing segmentation only when cancer is detected. This will be followed by documenting the business objectives, which will specify the objectives of creating an EfficientNetB1-based classifier,

incorporating a better FCM segmentation module, and deploying the system through an interactive dashboard.

Finally, we will establish the scope of business by determining what will be a part of the project, i.e., proof-of-concept model development and deployment of dashboards, and what will not be a part, i.e., 3D volumetric segmentation and direct PACS integration. The output of this business understanding phase will be an explicitly defined project scope and objectives that motivate the subsequent technical effort in subsequent CRISP-DM phases, consistent with the broader objective of delivering a lightweight and clinician-modifiable lung cancer detection system.

3.1.2 Data Understanding

To establish a clear understanding of the data, we plan to begin by reviewing the dataset selected for this project, which is the publicly released 2D Chest CT Scan Image Dataset on Kaggle. This will include discussing the form of the dataset, including the number of images, train, validation, and test sets split, and class distribution among normal and cancer classes. We will also verify the image types, resolutions, and quality for any inconsistencies or noise that would interfere with model training.

Second, we will review the dataset for any issue such as class imbalance and contrast or brightness level variation. Based on this, we can determine whether data augmentation or preprocessing steps would be necessary in order to improve model robustness. We will also review the suitability of this dataset for segmentation and classification tasks, observing that while it provides labelled classes for classification, it does not provide ground truth tumour masks for segmentation.

From these observations, we will note the key characteristics and drawbacks of the dataset. This will inform the data preparation plan, in particular with respect to resizing, normalisation, augmentation, and segmentation preprocessing steps. The expected outcome of this phase will be a comprehensive analysis of the dataset, including its strengths and weaknesses, and the optimal manner in which it can be leveraged in the classification-first pipeline. This

identification will guide the design of appropriate preprocessing and modeling methods in subsequent CRISP-DM stage.

3.1.3 Data Preprocessing

In preparing the dataset for modelling, we plan to begin by normalising all CT images to a standard resolution of 240×240 pixels. Resizing will enhance uniformity in the dataset and compatibility with EfficientNet input requirements. Normalisation of pixel intensity values to the range [0,1] will also be done to stabilise the training process and improve convergence.

After standardisation, we will apply data augmentation techniques such as rotation, flipping, and brightness adjustment to artificially increase the size of the dataset along with variability. This will prevent overfitting and improve the generalisation ability of the classification model, especially considering the small size of the medical dataset.

For the classification experiments, we will prepare the data in stratified 5-fold cross-validation splits in such a manner that cancerous and non-cancerous cases are well represented in both validation and training sets. This will provide us with a robust framework for the evaluation of the classification model.

For segmentation, we will preprocess the input images by applying preprocessing filters to highlight potential tumour regions. Preprocessing will include lung mask filtering to isolate the lung region, fixed thresholding on membership maps, and morphological operations such as small object removal and hole filling. We will also apply heuristic filters such as brightness, size, and shape constraints to strengthen tumour candidates and remove false positives.

The result of this phase will be a cleaned, standardised, and augmented dataset that will be suitable for both classification and segmentation. By carefully preparing the data, we will ensure that the models trained in the next CRISP-DM stage have the best possible chance of achieving high accuracy, robustness, and interpretability.

3.1.4 Modelling

In the modelling phase, we plan on developing and implementing a two-stage classification-first pipeline. Classification will be the initial stage where we will train an EfficientNetB1 model to classify multi-type of lung cancer CT scans. EfficientNetB1 is chosen because it has a fine balance between accuracy and computational cost, which is perfect for medical imaging tasks. To provide a basis of comparison, we shall also attempt to experiment with EfficientNetB0 so that we can contrast the compromises in stability and performance.

Segmentation will be the task of the second stage of the pipeline. If a classifier labels an image as cancerous, then segmentation module would be called for. This module will be built from an adapted fuzzy C-Means (FCM) clustering algorithm. Spatial and morphological enhancements such as Otsu thresholding, small object removal, hole filling, and shape-based heuristics (e.g., brightness, eccentricity, solidity) will be added to improve segmentation quality. These will help tumour localise more accurately and reduce false positives. Segmentation output will give tumour masks, centroid markers, and overlays for interpretability.

The modelling approach will therefore take advantage of deep learning classification and unsupervised segmentation clustering power, ensuring that the system ensuring that the system remains lightweight and modular. The final output from this phase will be a working classification-first pipeline that not only identifies cancer presence but also provides interpretable tumour localisation when required.

3.1.5 Evaluation

In the evaluation phase, we plan to start by comparing the performance of EfficientNetB1 and EfficientNetB0 on the classification task. The two models will be trained and validated through stratified 5-fold cross-validation and their mean accuracy over the folds computed. This will allow us to compare the two models in terms of overall stability and performance. On the basis of the comparison, the model with higher mean accuracy will be selected as the best classifier. Once selected, the model will be tested on the held-out test set in order to provide an account of its ultimate accuracy and provide an unbiased estimation of its generalisation ability.

For the segmentation step, there are no quantitative ground truth masks available in the data. Therefore, evaluation would be limited to qualitative visual inspection of overlay images. The tumour masks and centroid markers produced by the enhanced fuzzy C-Means segmentation module would be overlaid on the original CT scans to see if the detected areas are visually consistent with plausible locations of a tumour.

The result of this evaluation stage will be a final comparison between EfficientNetB1 and B0 using cross-validation, confirmation that the model selected is doing well on the test set, and visual inspection that the output of the segmentation is interpretable and is clinically meaningful.

3.1.6 Deployment

In the deployment phase, we plan to integrate the classification and segmentation modules as an interactive application presenting a demonstration of the system's proposed functionality. This will be achieved by developing a Streamlit dashboard that simulates a clinical decision-support tool. The dashboard will allow users to upload chest CT scan images, which will then pass through the classification-first pipeline. If cancer is detected by the classifier, segmentation module will be triggered to generate tumour masks and centroid overlays.

The dashboard will provide various outputs to enable interpretability, including the original CT scan, derived tumour mask, heatmap visualisations of FCM membership maps, and overlay images with annotated tumour regions and centroid markers. In addition to visual outputs, the system provides automated reporting functionality, such as exporting tumour size measurements and annotations as PDF reports for the clinician to view, and AIM-compliant XML files to enable integration into hospital information systems. There is also a local SQLite database to store case information so that users may access and view past results when required. Finally, users or end-users can validate the result of segmentation by either accepting, modifying, or rejecting suggested results for clinical reliability and flexibility.

The deployment will therefore focus on demonstrating usability and clinical relevance rather than clinical integration in the real-time setting. The deliverable of this phase should be an operational prototype dashboard that displays the whole classification-first workflow, provides understandable results for radiologists, and generates structured reports for documentation and review. This will elicit the utility value of the system and show how it can be used in real healthcare practice.

3.2 Use Case Diagram and Description

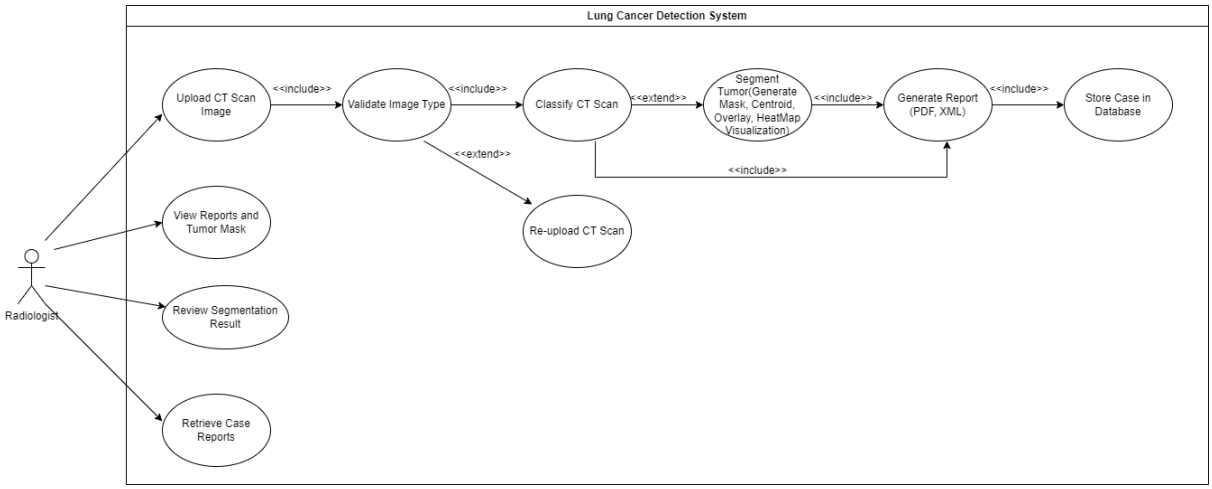


Figure 3.2.1 Use Case Diagram of the Lung Cancer Detection System

The Lung Cancer Detection System as shown in figure 3.2.1 is designed to help radiologists detect and localise lung cancer at an early stage with the use of CT scans. The system begins once the radiologist has uploaded an image of a CT scan. The system validates the file type to verify that the input is a legitimate CT scan. In case of failure, the radiologist can re-upload a valid or re-taken scan.

On validation, the system classifies the CT scan by an EfficientNetB1-based deep learning model into different type of cases like Adenocarcinoma, Squamous Cell Carcinoma, Large Cell Carcinoma and Normal. When the scan is classified as cancerous, the system extends its operation by performing tumor segmentation by a modified fuzzy C-Means clustering method. This segmentation gives tumor masks, centroids, overlays, and heatmaps that can be manually inspected visually.

Radiologists can view the results and tumor masks directly in the system and then review the output of segmentation by accepting, editing, or rejecting the results appropriately. In clinical applications, automated reporting functionality is incorporated within the system to allow users to review and download structured reports in PDF and XML formats.

Each case is stored in a database along with its reports and segmentation output. The radiologist can later retrieve earlier cases from the database for observation, comparison, or follow-up analysis.

The use case design ensures that the system provides efficient classification, conditional localisation of tumor, and clinically interpretable reporting with modularity and light weight execution.

3.3 Activity Diagram

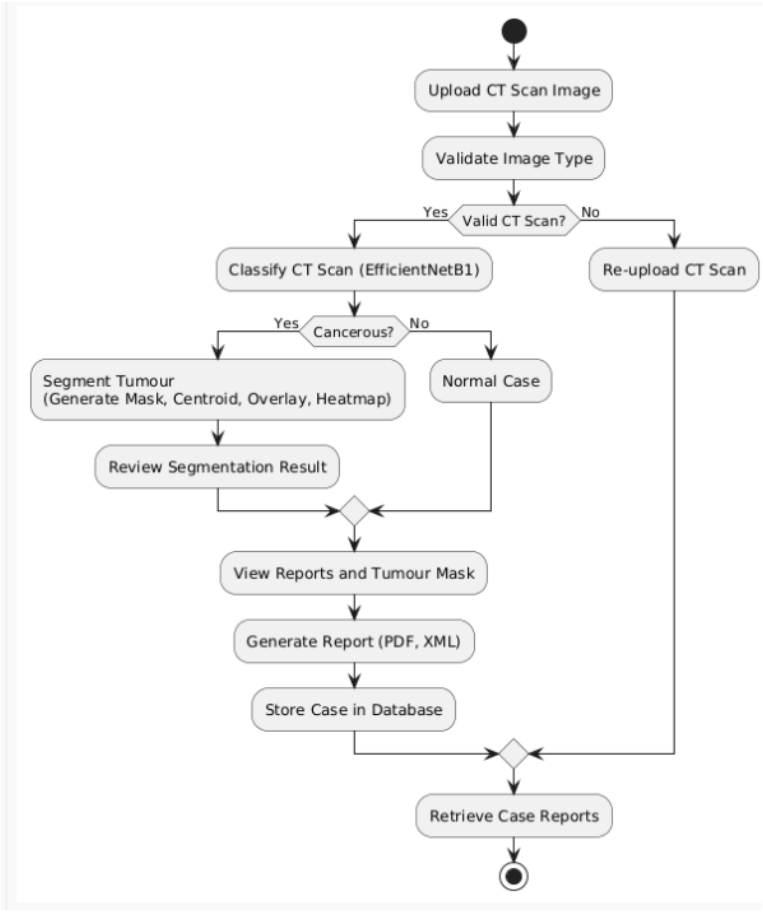


Figure 3.3.1 Activity Diagram of Lung Cancer Detection System

The activity diagram illustrates the overall work flow of the proposed lung cancer detection system. It begins when the radiologist uploads a CT scan image. The system then validates the type of image. When the file uploaded is not valid, the radiologist is prompted to re-upload a valid CT scan, looping the process back to the uploading stage.

If the scan is valid, the system proceeds with classification through the EfficientNetB1 model. At this stage, two outcomes are possible: if the scan falls in the normal category, the case is labeled as a normal case; if the scan falls in the cancerous category, the system triggers the segmentation module. Segmentation is carried out using an enhanced Fuzzy C-Means (FCM) algorithm to generate tumour masks, centroids, overlays, and heatmaps. The results of segmentation are reviewed by the radiologist, who decides whether they are acceptable or require adjustment.

Following this, the radiologist can view the complete set of reports and tumour masks. The system also supports generating structured reports in PDF and XML formats for clinical use. Once the results and reports are finalized, the case is saved automatically in the database for record and future analysis.

Finally, the radiologist is able to access case reports saved previously in the database for reviewing, comparing, or following up. This enables the system to provide not only real-time analysis but continuity as well for patient management

Chapter 4

System Design

The system design phase captures the overall architecture, requirements, operations, and desired workflow of the proposed Lung Cancer Detection Dashboard. The chapter seeks to clarify how the classification-first pipeline and its supporting modules were conceptualized, structured, and integrated into a comprehensive medical dashboard that simulates a clinical workflow in real life.

4.1 System Overview

The system was designed as a lightweight, interactive, and modular medical dashboard created using Streamlit. The dashboard unifies lung cancer classification, tumour segmentation, lesion measurement, radiologists' review, automatic reporting, and case management into a single application. It was designed to provide clinicians with an intuitive interface mirroring clinical diagnostic workflow, while remaining computationally efficient and feasible for deployment on standard hardware.

4.2 Dashboard Specifications

The dashboard was developed with a clear set of functional and technical specifications to ensure both usability and reliability. The frontend is built using Streamlit, a Python-based framework that allows the system to be accessed directly through a web browser, providing an interactive and user-friendly environment. In the backend, EfficientNetB1 is employed for image classification and a modified Fuzzy C-Means (FCM) algorithm is implemented for tumor segmentation to provide efficient and accurate analysis of CT scans. To provide support for data handling, a light-weighted SQLite database is employed for handling small-scale storage and retrieval processes. The platform supports dual reporting outputs in the form of PDF reports for human-readable clinical summaries and AIM XML files for machine-readable lesion annotations, to suit both research and clinical applications. Input is designed to be flexible, allowing CT scan images in .png, .jpg, and .bmp formats with file sizes of up to 200 MB, making the platform versatile and capable of handling high-resolution medical imaging data.

4.3 System Architecture

The system architecture adopts a modular design (Figure 4.3.1), ensuring that each component functions independently while remaining integrated within a unified workflow. The workflow begins with the User Input Layer, which collects patient information as well as CT scan images. These inputs are processed by the Classification Module, where EfficientNetB1—supplemented with EfficientNetB0 as a baseline comparator—categorises the scans into one of four classes: Adenocarcinoma, Large Cell Carcinoma, Squamous Cell Carcinoma, or Normal. If cancer is detected, the Segmentation Module is activated, employing a modified Fuzzy C-Means (FCM) clustering algorithm to identify the tumour regions. In this case, the Lesion Measurement Module calculates significant tumour characteristics such as area, diameter, volume, side, and lobe location to facilitate quantitative evaluation. The findings are then passed to the Reporting Module, which creates human-readable clinical reports in PDF format and machine-readable annotations in AIM XML format. To facilitate clinical decision-making, a Radiologist Review Module is also provided that allows specialists to accept, reject, or manually edit the AI-derived findings. Finally, all cases are organised and stored in the Case Browser Module which uses SQLite for efficient storage, retrieval, and maintenance of past records.

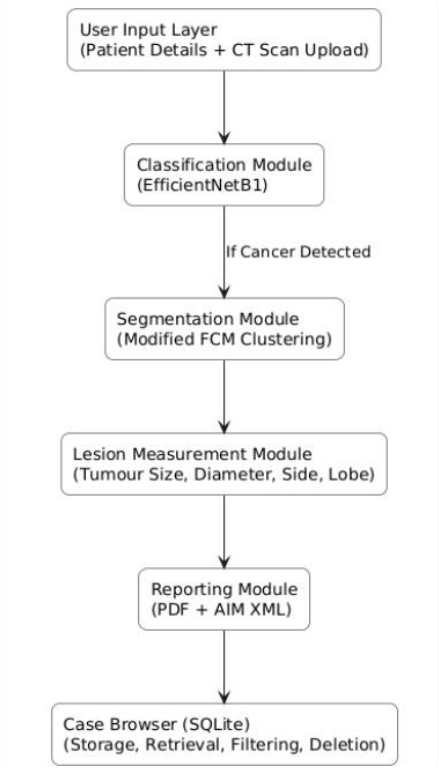


Figure 4.3.1 System Architecture of the Proposed Lung Cancer Detection Dashboard

4.4 Planned Features and Functions

The dashboard is constructed with four main functional modules, with each designed to serve a particular purpose to ensure there is a smooth user experience and clinical functionality. The Introduction Module provides background information on lung cancer and highlights the importance of AI in early cancer detection and presents worldwide statistics and clinical reasons behind new users before they access the predictive capabilities. The About the Dataset Module presents transparency through dataset specification, i.e., the four diagnostic classes used in the project, and presents dataset distribution visually through bar graphs and tables to enable comprehension of the data basis of the system. At the core lies the Prediction Module, which enables users to input patient metadata and upload CT scans for analysis. This module performs classification using EfficientNetB1 and conditional segmentation through the modified Fuzzy C-Means algorithm, producing lesion-level overlays, tumour masks, and centroid visualisations. It also provides quantitative lesion measurement such as area, volume, side, and lobe location, along with PDF and AIM XML reporting. Most importantly, the module allows for radiologist review through acceptance, rejection, or manual editing of AI output to foster machine intelligence and human expertise collaboration. Finally, the Case Browser Module is an SQLite-based case management system that presents processed cases in a filterable and searchable table. It allows searching by Patient ID, type of cancer, or scan date, and retrieval of diagnostic reports and segmentation overlays. Additionally, a "Danger Zone" safeguard is included to securely manage bulk deletion of records, ensuring controlled handling of sensitive clinical data.

4.5 Planned Workflow

The workflow of the designed system was developed to closely simulate a clinical decision-support environment (Figure 4.5.1), and it could be easily integrated into radiological practice. The process begins with the radiologist uploading a CT scan and inputting the corresponding patient information. First, the system verifies the uploaded file to be valid before classification by EfficientNetB1. In cases where cancer is predicted, the modified Fuzzy C-Means (FCM) segmentation is automatically triggered to localise tumour regions. Results are presented visually in the form of overlays, tumour masks, and centroid markers, supplemented with quantitative lesion measurements in terms of area, volume, and location. The system

subsequently generates structured reports in PDF and XML formats, which are stored in the database for later retrieval. The radiologist is then able to review the AI predictions, confirming or denying the results, or manually revise the segmentation masks to enhance accuracy. Finally, all case records—including images, measurements, and reports—are securely stored in the Case Browser, allowing easy retrieval, filtering, and review of past cases.

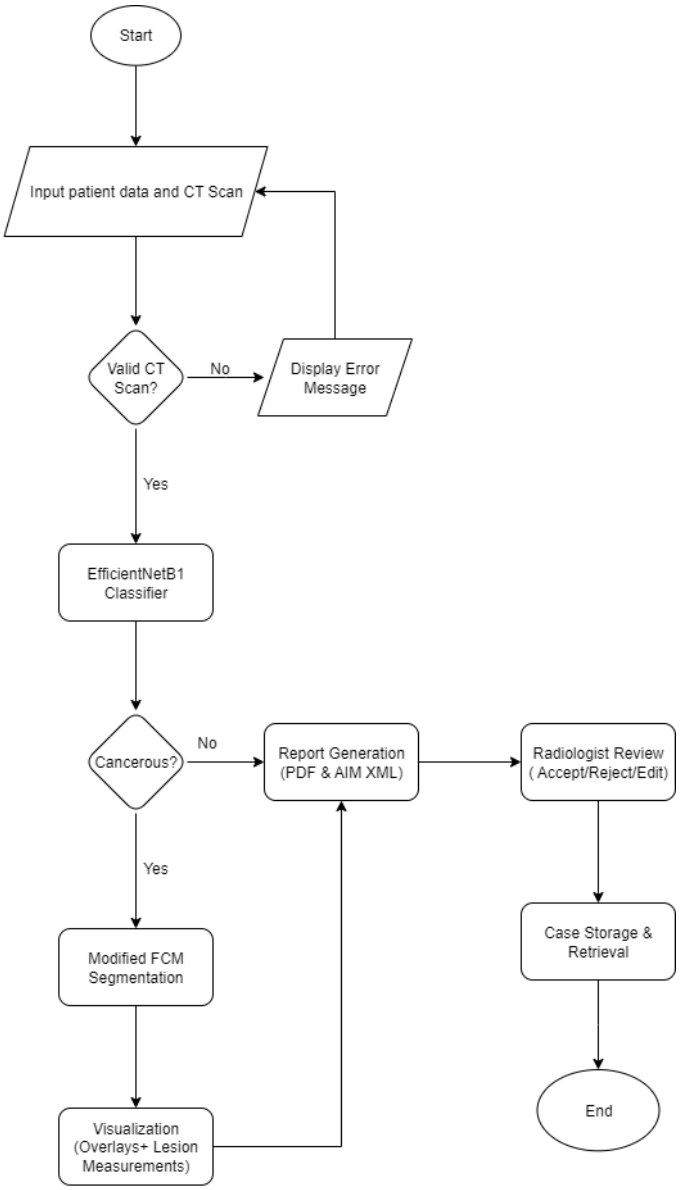


Figure 4.5.1 Planned Workflow of the Lung Cancer Detection System

4.6 Design Considerations

The design of the system was guided by several key considerations to balance technical efficiency with clinical practicality. Lightweight deployment was prioritized through utilizing EfficientNetB1, a lightweight but highly precise deep model, with SQLite, a light database suitable for small-scale storage, in an effort to make it deployable on modest hardware. In an effort to maintain the workflow as clinically relevant, classification-first was utilized, not performing unnecessary segmentation in normal scans and reducing the diagnostic process. Interpretability was addressed by incorporating segmentation overlays and quantitative lesion measurements, enabling clinicians to better understand and trust the AI's predictions. The dashboard was also built with usability in mind, offering intuitive navigation and clear, structured report outputs to support ease of use in a clinical environment. Finally, scalability was considered by designing the system to be proof-of-concept ready with SQLite, while leaving room for future integration with more robust clinical data systems such as PACS or SQL Server as deployment needs expand.

Chapter 5

System Implementation

This chapter provides the detailed implementation of the proposed Lung Cancer Detection System, following the CRISP-DM framework as introduced in Chapter 3. Each section describes how the planned design of Chapter 4 was implemented into a working prototype, from all phases of the pipeline from business understanding until deployment. The chapter begins with the implementation stages of business and data understanding, wherein the clinical rationale, characteristics of the dataset, and preprocessing strategies come into play again in implementation context. It then continues to outline the modelling stage, focusing on EfficientNetB0 and B1 classifying model creation and its modified Fuzzy C-Means segmentation component. The evaluation stage details testing classification and segmentation capability using quantitative indicators and qualitative observation. Finally, the deployment stage is discussed, including the integration and design of the interactive Streamlit dashboard, such as prediction, reporting, radiologist review, storage, and retrieval of cases. Overall, the series shows how the system was sequentially deployed to achieve the project goals and simulate a clinically suitable diagnostic process.

5.1 Crisp-DM Implementation

5.1.1 Business Understanding

Under the business understanding, the project began with a review of the history of lung cancer detection to establish both the technical and clinical context. Lung cancer remains a leading cause of cancer death worldwide, and survival is closely linked to the stage of diagnosis. Computed Tomography (CT) scans emerged as the most appropriate non-invasive imaging method for detecting pulmonary nodules and irregular growths [2]. Another challenge is that manual interpretation of CT scans by radiologists is time-consuming, subjective, and error-prone, especially when it comes to handling huge imaging volumes or detecting subtle early-stage tumours [3]. This necessitated the need for computer-aided diagnosis (CAD) systems that provide accurate, efficient, and interpretable results to support radiologists in early diagnosis and treatment planning.

From this review, various limitations of current CAD systems were established. Classification-only pipelines were seen to have no tumour localisation, which was clinically limiting. Segmentation-first pipelines were wasting resources in carrying out segmentation even for normal scans, whereas parallel pipelines were carrying out both classification and segmentation in parallel, leading to redundant computation [2][3]. These limitations showed that there was a need for a classification-first pipeline that only triggers segmentation when cancer is suspected, thus conserving resources and increasing efficiency.

From these difficulties, the project motivation was defined as the need to create an ultra-lightweight, modular, and clinically feasible diagnostic system that mirrors real-world work processes and computational efficiency. The project aimed not only to provide correct classification but also to produce tumour localisation for interpretability, which is essential for clinical adoption.

The business goals were then determined as: (1) To develop an EfficientNetB1-based deep learning classification model to differentiate between normal and cancerous CT scans with high accuracy, with EfficientNetB0 as a performance benchmark. (2) To design and implement a conditional Fuzzy C-Means (FCM) segmentation module, activated only in the event of cancer detection, to provide tumour masks, centroid overlays, and interpretability. (3) To deploy the system through an interactive Streamlit dashboard with integrated classification, segmentation, case storage, and reporting features.

The project scope was defined to include proof-of-concept development over a publicly accessible 2D Kaggle CT scan dataset. Tasks such as 3D volumetric segmentation, DICOM image processing, real-time PACS integration, and clinical trials were out of scope. This assisted in maintaining the system lightweight and feasible within available resources.

The final deliverables for this project are: (1) A classification-first pipeline that combines EfficientNetB1-based classification with optimized FCM segmentation. (2) An interactive Streamlit dashboard to simulate clinical workflows. (3) Visual outputs as tumour masks, overlays, and heatmaps. (4) Structured reporting in PDF and XML formats for clinical decision-making. (5)A local SQLite database for storing and retrieving processed cases.

This business insight served to keep subsequent phases of CRISP-DM focused on the end goal of delivering a lightweight, modular, and clinically flexible system for lung cancer detection and localisation.

5.1.2 Data Understanding

At data understanding, the Kaggle 2D Chest CT Scan Image Dataset was examined to assess its structure, size, and suitability for the proposed classification-first pipeline. The data set had already been split into three subsets: training, validation, and test. To quantify the distribution of the dataset, a Python function (GetDatasetSize) was employed to iterate across each directory and count the images in each class. The four classes which were accessible to employ were: Normal, Adenocarcinoma, Squamous Cell Carcinoma, and Large Cell Carcinoma.

Figure 5.1.1and table 5.1 illustrates the output of this analysis. The training set included the highest number of images, with the testing and validation sets following thereafter. Specifically, the Squamous Cell Carcinoma class contained 195 training images, 23 validation images, and 120 test images, and the Normal class contained 115 training images, 13 validation images, and 51 test images. The Adenocarcinoma class contained 148 training images, 21 validation images, and 54 test images, and the Large Cell Carcinoma class contained 155 training images, 15 validation images, and 90 test images.

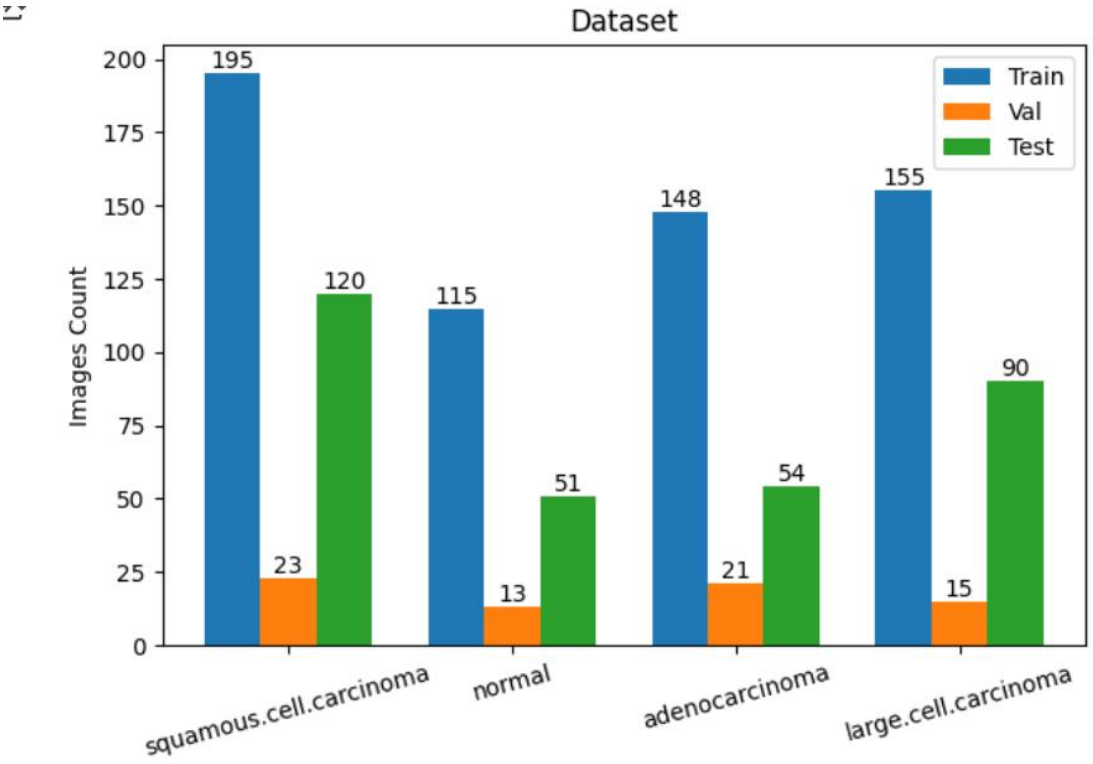


Figure 5.1.1 Dataset Distribution Across Train, Validation, and Test Sets

Class	Training	Validation	Test	Total
Squamous Cell Carcinoma	195	23	120	338
Normal	115	13	51	179
Adenocarcinoma	148	21	54	223
Large Cell Carcinoma	155	15	90	260
Total	**613**	**72**	**315**	**1000**

Table 5.1 Number of Images per Class in Train, Validation, and Test Sets

This division revealed that although the overall dataset was balanced, there was some imbalance among classes, with Normal and Squamous Cell Carcinoma having less samples compared to Adenocarcinoma and Large Cell Carcinoma in the train set. In order to better illustrate the same, a bar graph was plotted contrasting the class-wise image counts among the train, validation, and test sub-sets.

From this study, some significant findings were revealed: (1) The dataset was sufficiently diverse to train and evaluate a classification model across four categories. (2) The dataset was split in a 70:10:20 ratio, which provided a good split for training, validation, and testing. (3) There was slight class imbalance, which could potentially affect model training. (4) The images were stored as .png, with variations in brightness and resolution, as would be the case with real CT scans.

These findings emphasized the need for data preparation techniques such as image resizing, normalisation, and augmentation to make the data uniform and strong for modelling. The data was valid for classification tasks, but since no ground truth segmentation masks were present, the segmentation module was merely qualitatively tested using visual checks.

5.1.3 Data Preprocessing

At data preparation, several preprocessing and data augmentation tasks were undertaken to ensure dataset consistency, robustness, and balance between classes at model training time. Since the initial dataset consisted of images of varying sizes, all CT images were resized to a uniform resolution of 240×240 pixels for EfficientNetB1 and 224×224 pixels for EfficientNetB0. Resizing provided a consistent input shape and ensured compatibility with the respective model architectures.

Pixel intensity values were similarly normalized to the [0,1] range and subsequently preprocessed further using the EfficientNet preprocessing function (preprocess_input) to be within the scale necessary for the pre-trained ImageNet weights. The action stabilized training and assisted convergence.

Selective data augmentation methods were applied to the training set to address the moderate class imbalance that had been identified under data understanding. The improvement included random rotations (±10°), zooming (±10%), horizontal and vertical shifts (5%), and faked brightness variability (0.9–1.1). These detailed changes, summarized in table 5.2 artificially increased the number of examples observed during training, namely augmenting the sparsely-represented classes such as Normal and Large Cell Carcinoma with increased variability.

Importantly, the augmentation was carried out only for the training set to prevent data leakage, while validation and test sets remained unchanged except for resizing and preprocessing.

Techniques	Parameter Range	Purpose
Rotation	±10°	Adds orientation variability, improves robustness to different scan angles.
Zoom	Up to 10%	Simulates variation in tumour size and magnification.
Width Shift	±5% of image width	Introduces positional variability along the horizontal axis.
Height Shift	±5% of image height	Introduces positional variability along the vertical axis.
Brightness Adjustment	0.9 – 1.1	Accounts for variation in scan contrast and brightness levels.
Preprocessing Function	preprocess_input (EfficientNet)	Normalises input according to ImageNet-trained EfficientNet requirements.

Table 5.2 Summary of Data Augmentation Techniques for Training Set

There was also a visualisation pipeline implemented to study augmented samples of the training set. Sample images of different classes (Normal, Adenocarcinoma, Large Cell Carcinoma, and Squamous Cell Carcinoma) as shown in figure 5.1.2 confirmed that augmentation preserved anatomical structure with the injection of variability, thereby reducing the risk of overfitting and improving generalisation.

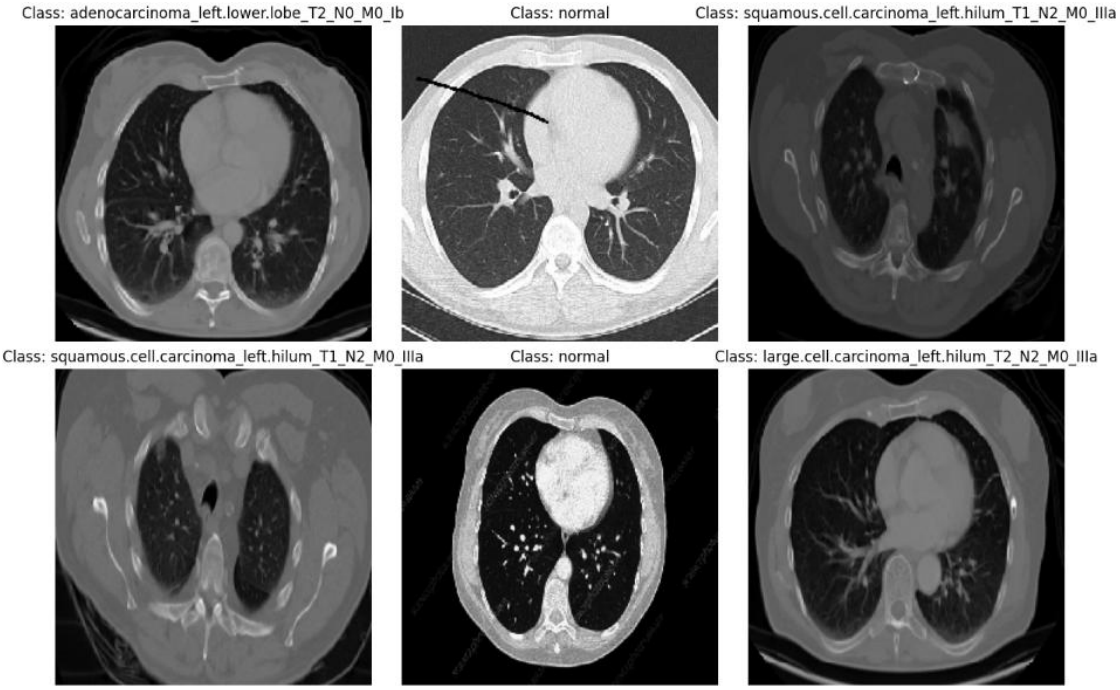


Figure 5.1.2 Sample CT Images from Training Dataset After Preprocessing and Visualisation

For segmentation procedures, additional preprocessing filters were applied to grayscale CT images before clustering. These included lung masking for isolating the lung region, fixed thresholding of membership maps, and morphological operations such as small object elimination, filling of holes, and opening by a structuring disk. Heuristic filtering was also included, with restrictions on brightness, area size, and shape features (e.g., eccentricity, solidity), to remove false positives and filter tumour candidates.

Overall, the data preparation phase produced a clean, standardised, and augmented dataset. This preprocessing not only ensured consistency across different CT images but also mitigated the effect of class imbalance by enriching the training set with more diverse examples. As a result, the prepared dataset was well-suited for robust classification experiments and for producing clinically meaningful segmentation overlays.

5.1.4 Modelling

During the Modelling phase of the project, a two-stage classification-first pipeline was designed and developed that comprised (i) an image classification module and (ii) a conditional segmentation module for tumour localisation.

EfficientNetB0 and EfficientNetB1 were employed for the classification task for both the variants of the EfficientNet family. Both models were pre-trained with pre-trained ImageNet weights and then fine-tuned with a custom classification head made up of a Global Average Pooling layer, a Dropout layer (rate = 0.5) for regularization, and a Dense layer with four softmax outputs representing the classes in the dataset: Normal, Adenocarcinoma, Squamous Cell Carcinoma, and Large Cell Carcinoma. Training was performed in two stages, beginning with full-layer training followed by fine-tuning with a reduced learning rate (1e-4). The models were trained using the Adam optimiser (learning rate = 0.001), categorical cross-entropy loss, and accuracy as the primary performance metric. To prevent overfitting and convergent stability, a few callbacks were employed, including EarlyStopping to terminate training when validation loss plateaus, ModelCheckpoint to save the best model weights, and ReduceLROnPlateau to decrease learning rate dynamically when validation performance plates.

In terms of performance, EfficientNetB0 reached ultimate training accuracy of 99.35% and peak validation accuracy of 95.83% at epoch 20, although validation accuracy varied across epochs, which suggests moderate instability. EfficientNetB1, with slightly lower peak validation accuracy of 90.28% at epoch 13, exhibited smoother convergence and more stable generalisation across folds. The results therefore indicate EfficientNetB0 with greater validation accuracy but increased stability of performance in practice potentially from EfficientNetB1. The results are shown in figure 5.1.3 and figure 5.1.4.

For instances tagged as cancer, a conditional segmentation component was invoked for interpretability and localisation of the tumour. It was implemented in the guise of an improved Fuzzy C-Means (iFCM) clustering algorithm with several improvements over the original process. These included pixels intensity normalisation to 0–1, lung masking to separate pulmonary regions from background structures, and thresholding of the FCM membership map

to isolate candidate tumour. Post-processing was also optimized by the application of morphological operations such as small object removal, hole filling, and disk filter opening. To reduce false alarms, region size, brightness, eccentricity, and solidity-based heuristic filters were applied with the introduction of an optional lobe-targeting mechanism to restrict detection to specific lung regions where applicable. The segmentation module generated four key outputs: (i) the original CT scan, (ii) the FCM membership map, (iii) the binary tumour mask, and (iv) an overlay image with centroid markers, thereby augmenting interpretability and diagnostic assistance.

Overall, modelling successfully offered an efficient yet modular pipeline where EfficientNetB0 and EfficientNetB1 act as a classification backbone, and a tailored FCM-based segmentation module works as an interpretability layer for positive instances of cancer. This configuration strikes a balance between computation efficiency, classification accuracy, and tumour localisation based on the objectives of the project in terms of accurate and interpretable lung cancer detection.

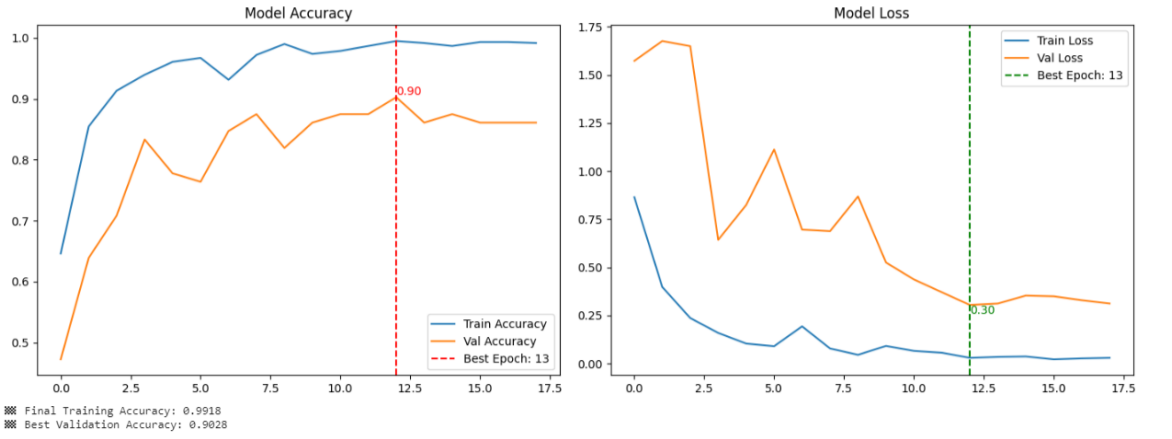


Figure 5.1.3 Training and validation accuracy or loss of EfficientNetB0

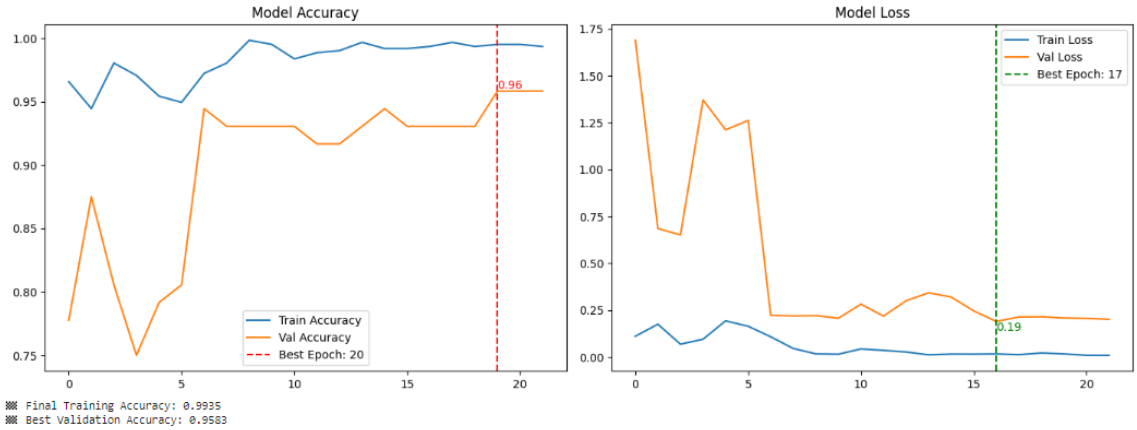


Figure 5.1.4 Training and validation accuracy or loss of EfficientNetB1

5.1.5 Evaluation

The evaluation phase compared the performance of the segmentation and classification modules to assess whether the proposed pipeline met its intended objectives.

EfficientNetB0 and EfficientNetB1 classification models were evaluated using a combination of training/validation performance curves, confusion matrices, k-fold cross-validation, and test set accuracy. As shown in Figure 5.1.3 and Figure 5.1.4, EfficientNetB0 achieved a best validation accuracy of 90.28% at epoch 13, and EfficientNetB1 achieved a best validation accuracy of 95.83% at epoch 20. Although EfficientNetB0 had better validation accuracy, it also had more fluctuation in validation performance, but EfficientNetB1 had smoother and more stable convergence across folds.

Confusion matrices and classification reports also provided more details on class-level performance. EfficientNetB0 presented high precision and recall for most classes but had slight misclassification in the squamous cell carcinoma class (Figure 5.1.5a). EfficientNetB1, while having a slightly lower validation accuracy, yielded a more balanced classification between classes, as revealed in Figure 5.1.5b.

To improve reliability, 5-fold cross-validation was performed on the combined training and validation sets. The results, summarised in Figure 5.1.5c, showed that EfficientNetB0 achieved a mean accuracy of 96.85% ± 0.0140, with EfficientNetB1 achieving 97.04% ± 0.0302. The results highlighted that EfficientNetB1, despite a lower single-run validation accuracy, offered

more consistent generalisation across folds. As such, EfficientNetB1 was selected as the final classification model. The model was then evaluated on the independent test set, achieving a test accuracy of 94.60% with a test loss of 0.227, confirming its ability to generalise well to new unseen data.

Six random CT images from the validation dataset were visually examined, with both ground truth and predicted labels displayed. As observed in Figure 5.1.5d, most of the predictions aligned with the ground truth correctly, and visual examination confirmed the stability and reliability of EfficientNetB1 for real-world classification tasks.

The segmentation module was evaluated qualitatively since there were no ground truth tumour masks available in the dataset. The enhanced Fuzzy C-Means (FCM) algorithm produced four significant outputs: the original CT scan, the membership map with tumour boundaries, the final binary tumour mask, and an overlay image with centroid markers for interpretability. This is shown in Figure 5.1.5e. Visual inspection demonstrated that the module successfully localised suspicious tumour regions in the lungs and removed false positives through morphological and heuristic refinements.

Although quantitative measures such as Dice similarity or Intersection-over-Union (IoU) could not be computed since segmentation labels were not available, the qualitative outputs were clinically reasonable. The overlays provided radiologists interpretable localisation outputs that complemented classification predictions.

In conclusion, the evaluation confirmed that the classification-first pipeline achieved outstanding performance in terms of both classification and interpretability. EfficientNetB1 was chosen as the final classifier due to its stable convergence and reliable performance across folds, with 94.60% accuracy on the independent test set. Clinical utility was also enhanced by the FCM-based segmentation module by providing tumour localisation overlays to facilitate radiologists in diagnosis and treatment planning.

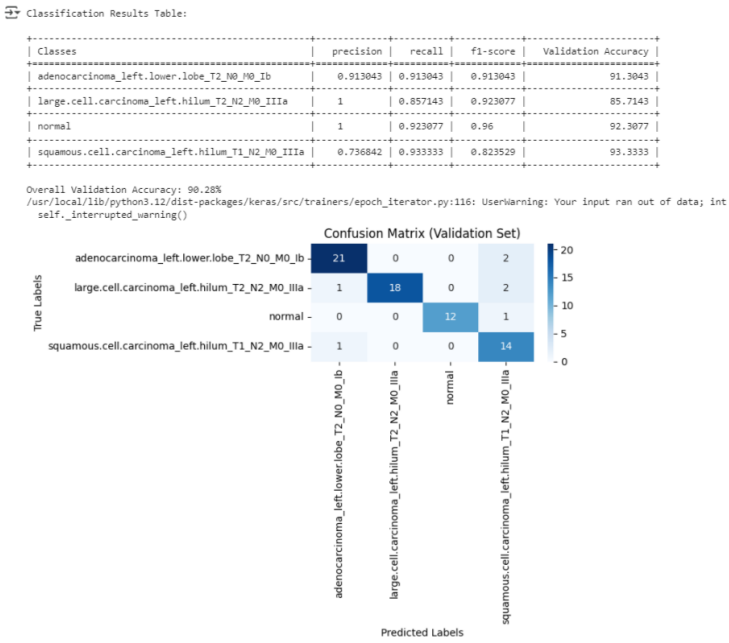


Figure 5.1.5 a: Confusion Matrix and Classification Report for EfficientNetB0

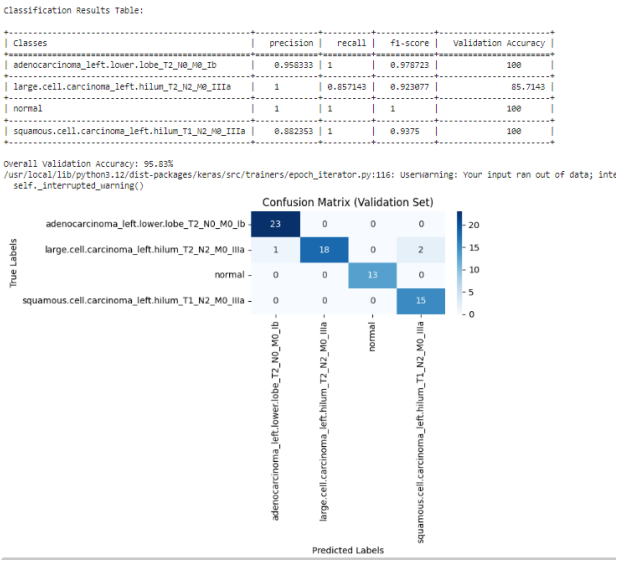


Figure 5.1.5b: Confusion Matrix and Classification Report for EfficientNetB1

Fold-wise Comparison:

Fold	B0_Accuracy	B0_Loss	B1_Accuracy	B1_Loss
1	0.948148	0.208047	0.977778	0.196293
3	0.977778	0.117419	0.925926	0.242668
4	0.977778	0.108983	0.992593	0.0569412
5	0.97037	0.130337	0.985185	0.0803422

Summary:
B0 Mean Accuracy: 0.9685 ± 0.0140
B1 Mean Accuracy: 0.9704 ± 0.0302

Figure 5.1.5c: Fold-wise Comparison of EfficientNetB0 and B1 Across 5-fold Cross-validation

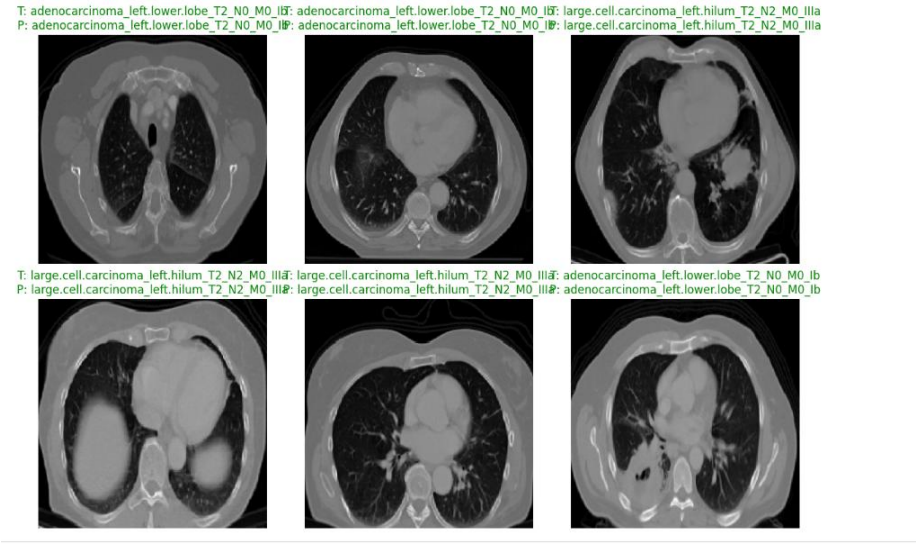


Figure 5.1.5d: Visual Classification Results of EfficientNetB1 on Random Validation Images

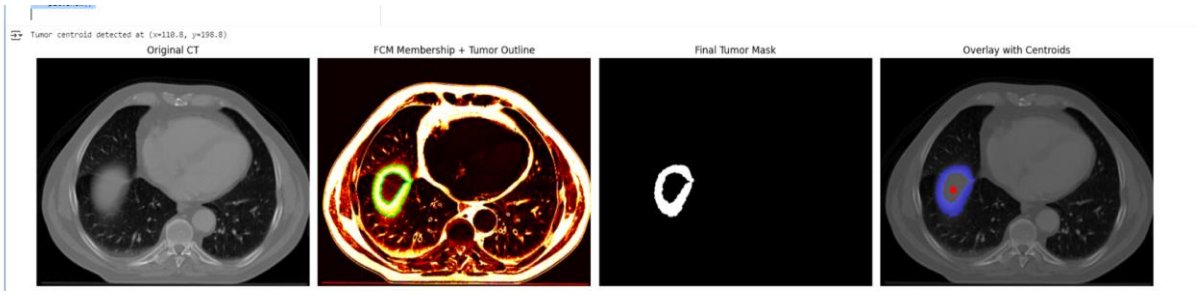


Figure 5.1.5e: Segmentation Outputs: Original CT, Membership Map, Tumour Mask, and Overlay with Centroids

5.1.6 Deployment

Introduction Module

The Introduction module of the system implemented is the entry point for users, providing important background knowledge and context about lung cancer, along with the use of artificial intelligence in medical imaging. The module was designed to be informative and interactive and help users such as radiologists, healthcare practitioners, and even non-technical users understand the importance of early detection and the function of the system.

In this module, the users are first provided with an introduction to lung cancer as a leading cause of cancer-related deaths worldwide, supported by visual images of the human lungs along with global statistics as shown in figure 5.1.6a and 5.1.6b. The module also highlights the limitations of the traditional diagnostic methods, including the insufficiency of chest X-rays, the manual interpretation requirement of CT scans, and the ineffectiveness of conventional CAD systems. These issues point to the need for developing light, AI-based detection tools.

To bridge this gap, the module introduces the promise of deep learning. The module explains how newer AI architectures like EfficientNetB1 (for classification), FCM-based segmentation, and transfer learning techniques enhance diagnostic accuracy and efficiency compared to traditional pipelines. The module also lists the types of lung cancer covered in this project—Adenocarcinoma, Squamous Cell Carcinoma, Large Cell Carcinoma, and Normal cases—providing users with a clear picture of the system's scope.

The purpose of this module is to orient the user before proceeding to the prediction and case management part of the system. It ensures that users have a sufficient level of background information on lung cancer detection and the system’s underlying approach.Users can navigate through the introduction, discover global context, understand system motivation, before proceeding to the dataset exploration and usage of the prediction feature with informed expectations.

Figure 5.1.6a Introduction Page of the Dashboard

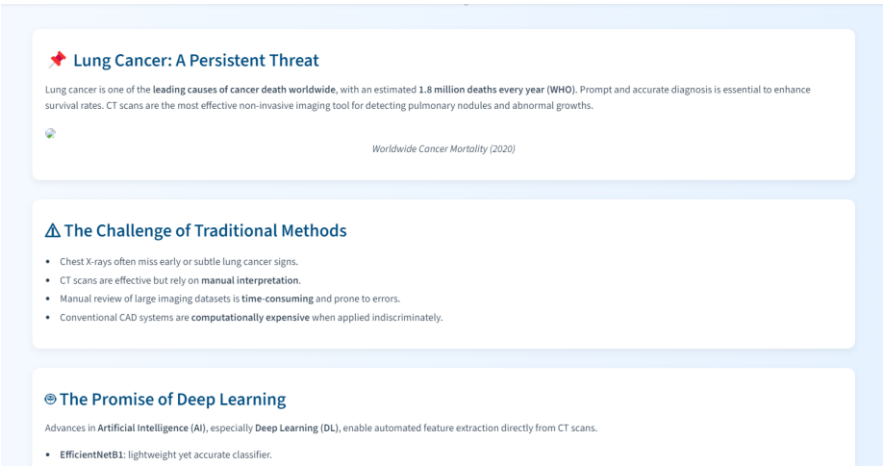


Figure 5.1.6b Lung Cancer Information

About The Dataset Module

The About the Dataset module was developed to provide users with an appropriate overview of the dataset that was used in the development of the lung cancer detection system. The purpose of this module is to foster transparency by allowing users to understand the data foundation on which the classification and segmentation models were established. By revealing dataset details in a visual and interactive manner, it helps users—particularly radiologists, researchers, and medical students—to build trust in the system's accuracy and limitations.

This module first explains the structure of the dataset, which includes three types of lung cancer—Adenocarcinoma, Large Cell Carcinoma, and Squamous Cell Carcinoma—and normal CT scan images. The specifications point out that the dataset consists of image file formats (.png) instead of DICOM, making it lightweight and ready for use with deep learning models without any need for conversion. It also explains the splitting of the data into training (70%), validation (10%), and test (20%) datasets for maintaining proper model development and unbiased testing.

The module additionally provides graphical summaries, i.e., distribution tables and bar plots (Figure 5.1.6 c and Figure 5.1.6d), of images per class in the training, validation, and testing datasets. The plots allow users to immediately identify dataset balance, confirm the adequacy of samples per class, and ascertain that no class is under-represented.

From a user perspective, this module serves as a reference hub. Users can determine which types of lung cancer are present in the dataset and review dataset distribution by different splits. Users can also use the graphs and tables as a quick validation of how representative and balanced the dataset is.

Overall, the About the Dataset module ensures that users are not only interacting with a black-box AI system but also understand the underlying data that powers the predictions, reinforcing trust and interpretability in clinical or research applications.

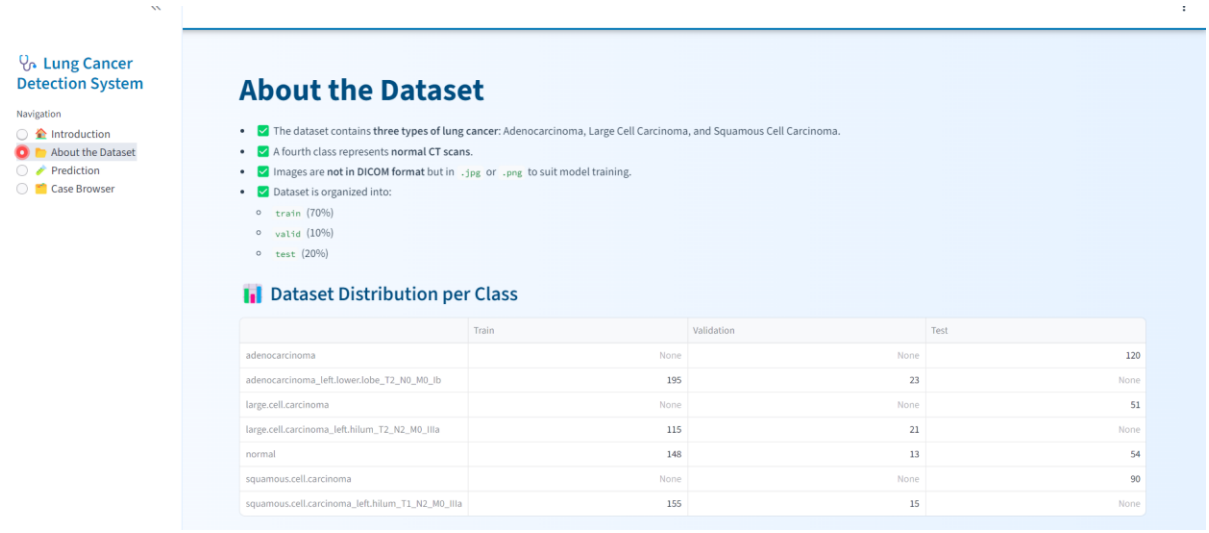


Figure 5.1.6 c Dataset Distribution Table by Class (Train, Validation, Test)

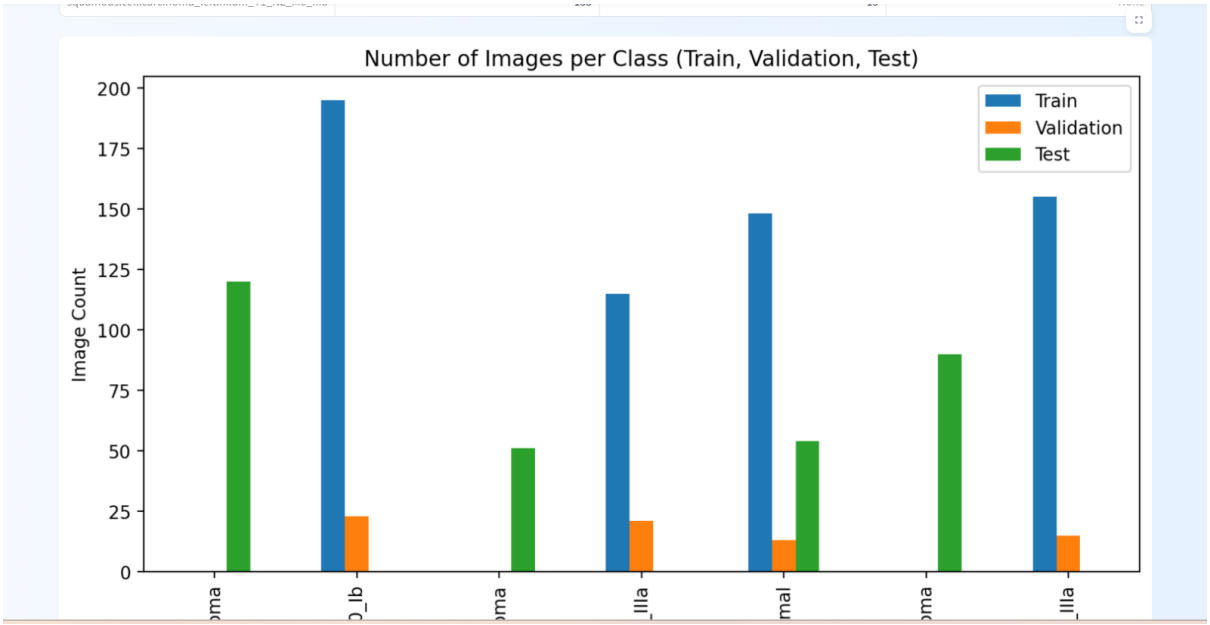


Figure 5.1.6d Bar Chart Showing Image Counts per Class Across Train, Validation, and Test Splits

Prediction Module

Prediction module is the central functional component of the deployed Lung Cancer Detection System, where users directly engage with the AI pipeline to perform the lung cancer classification and tumor segmentation as part of a clinical workflow. Its main purpose is to provide an intuitive interface that allows medical experts to read CT scans, get automated results, and generate structured diagnostic reports. As reflected in Figure 5.1.6e, the interface prompts the user to enter basic patient data, including Patient ID, Patient Name, and Scan Date, prior to uploading CT scan images in .png, .jpg, or .bmp formats (up to 200 MB). This ensures that every prediction is appropriately linked to a patient record for traceability and clinical context. Once an image is uploaded, the system processes the scan via two significant tasks. The first task is classification, where the EfficientNetB1 model predicts whether the scan belongs to one of four classes: Adenocarcinoma, Large Cell Carcinoma, Squamous Cell Carcinoma, or Normal. The predicted label is shown with a confidence measure in order to provide real-time diagnostic output, as in Figure 5.1.6f. The second task is segmentation, which is triggered automatically if the scan is predicted to be cancer. At this stage, the FCM segmentation pipeline is employed to detect suspicious tumor regions and generate four outputs: the original CT image, membership map with tumor outlines, binary mask of tumors, and overlay image with centroids for better interpretability (Figure 5.1.6g). Aside from visualization, the system also computes quantitative measurements of lesions such as tumor area, diameter, volume, involved lung side (left or right), and lobe location, as shown in Figure 5.1.6h.

For documentation and for downstream purposes, the module includes a reporting module wherein users can download an HTML or PDF report stating patient metadata, classification results, and segmentation visualizations (Figure 5.1.6i). It also creates an AIM XML file with lesion annotations and measurements in a formatted, machine-readable form for PACS integration or research databases (Figure 5.1.6j). When there is a normal scan prediction made by the classifier, the segmentation process is skipped, with the system instead generating a concise report stating that no tumors are found, but still logging the case (Figure 5.1.6k). To guarantee clinical validity, an integrated workflow of radiologist review is integrated in the module. As Figure 5.1.6l illustrates, the radiologists can accept, reject, comment, or edit an AI prediction using an editing mode to edit tumor masks manually (Figure 5.1.6m). Any edited

mask is stored and a new PDF/XML report is generated (Figure 5.1.6n), which ensures that there is a collaborative process between AI predictions and expert validation.

The module also incorporates robust error-handling mechanisms. If the uploaded file does not resemble a valid CT scan—for instance, if a non-medical image is uploaded—the system immediately raises a warning and guides the user on acceptable file types and scan formats, as depicted in Figure 5.1.6o. It maintains the workflow to ensure only valid medical scans are processed. Overall, the Prediction module is an entire pipeline that integrates classification, segmentation, lesion quantification, structured reporting, and expert review. By enabling both positive and normal case workflows with valid input protections, it demonstrates an end-to-end solution not only technically solid but also clinically viable.

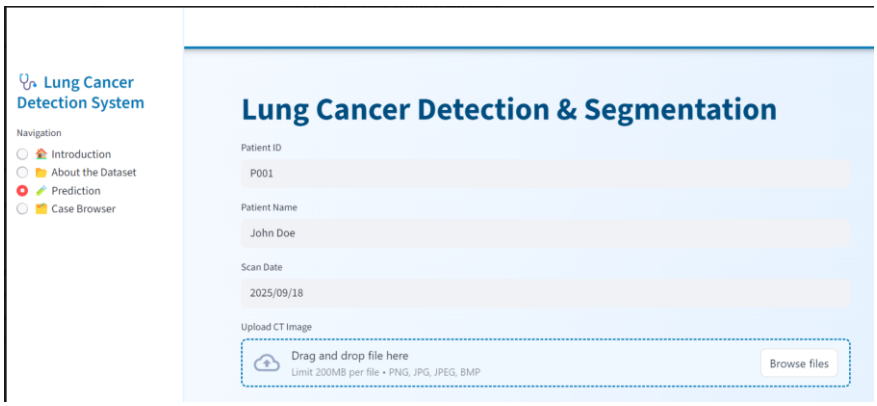


Figure 5.1.6e Prediction input interface (Patient ID, Name, Scan Date, image upload)

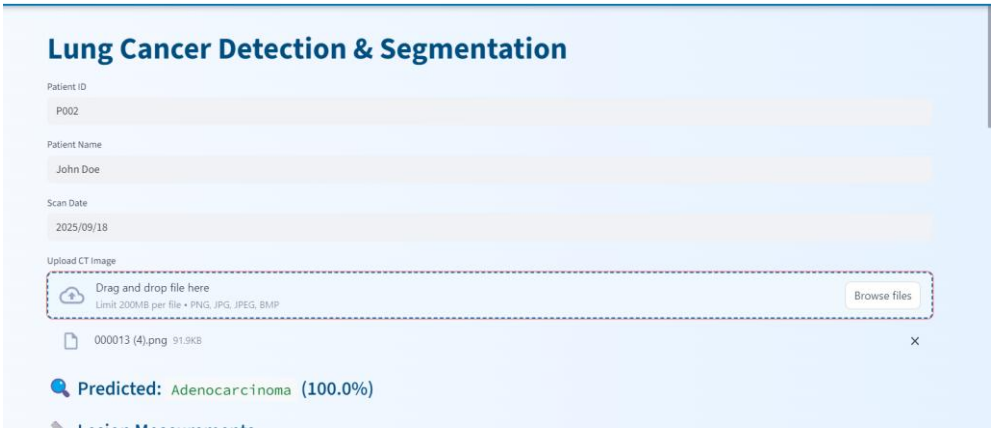


Figure 5.1.6f Classification result with predicted label and confidence score

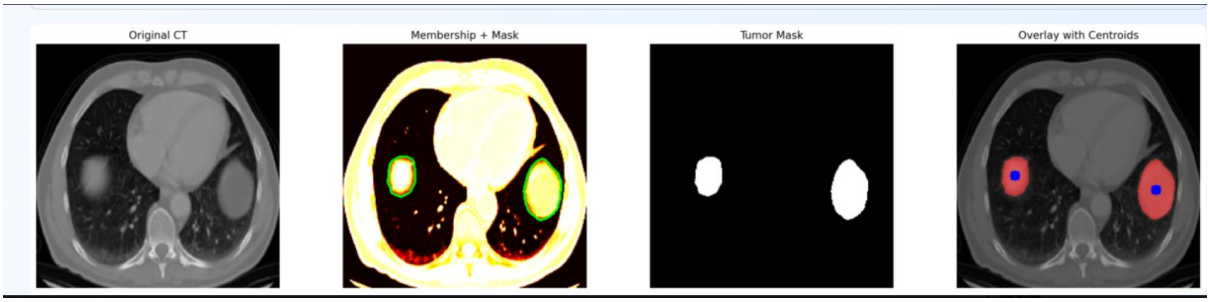


Figure 5.1.6g Tumour segmentation outputs (Original CT, Membership + Mask, Tumor Mask, Overlay with Centroids)

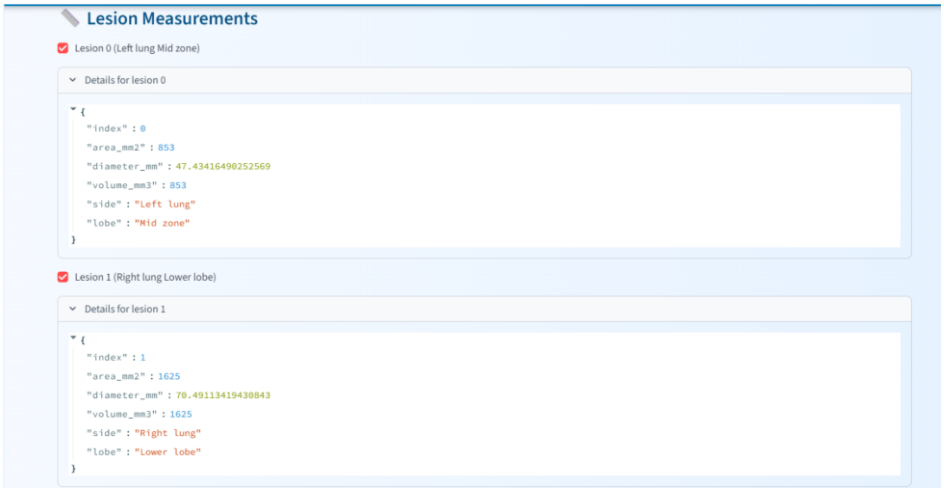


Figure 5.1.6 h Lesion measurements (area, diameter, volume, side, lobe)

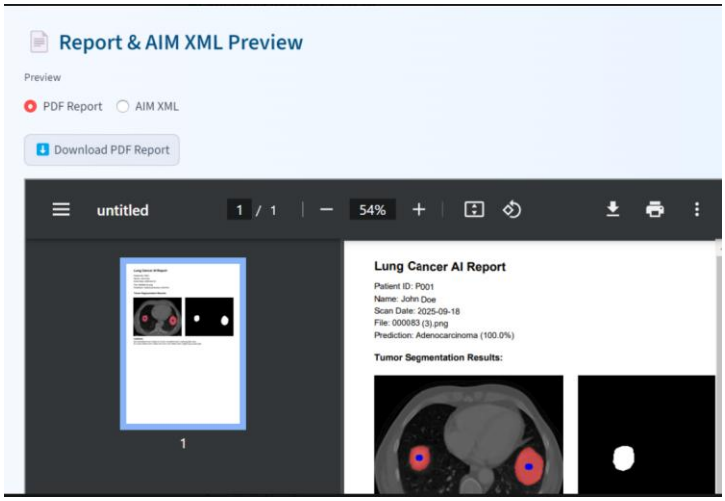


Figure 5.1.6i Generated PDF report preview

Figure 5.1.6j AIM XML annotation output

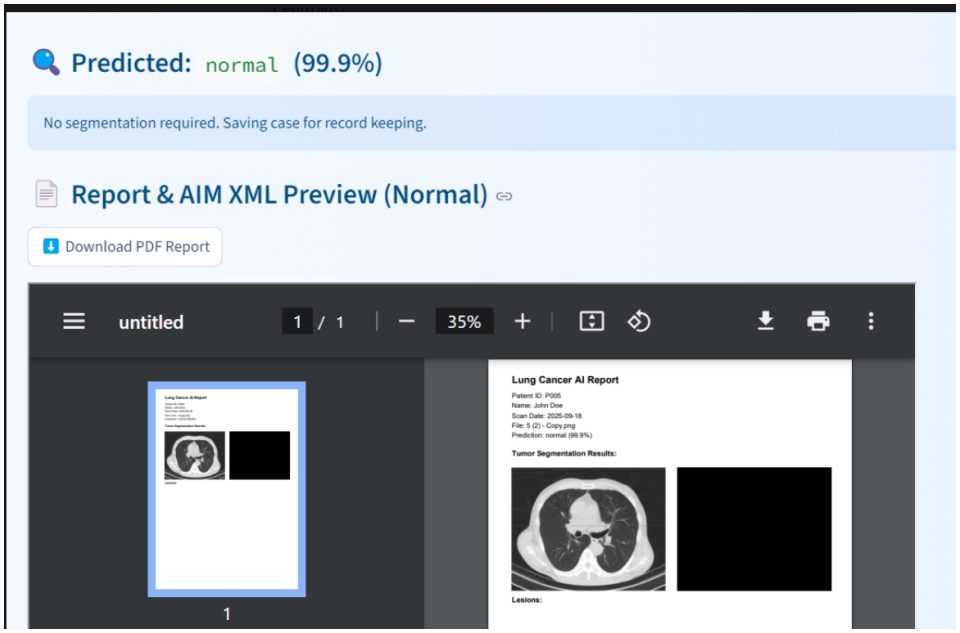


Figure 5.1.6k Normal case workflow with simplified report

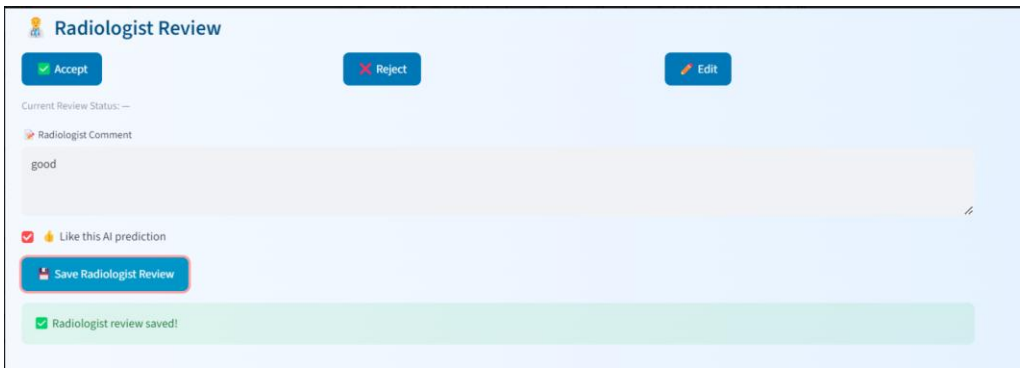


Figure 5.1.6l Radiologist review interface (accept/reject with comments)

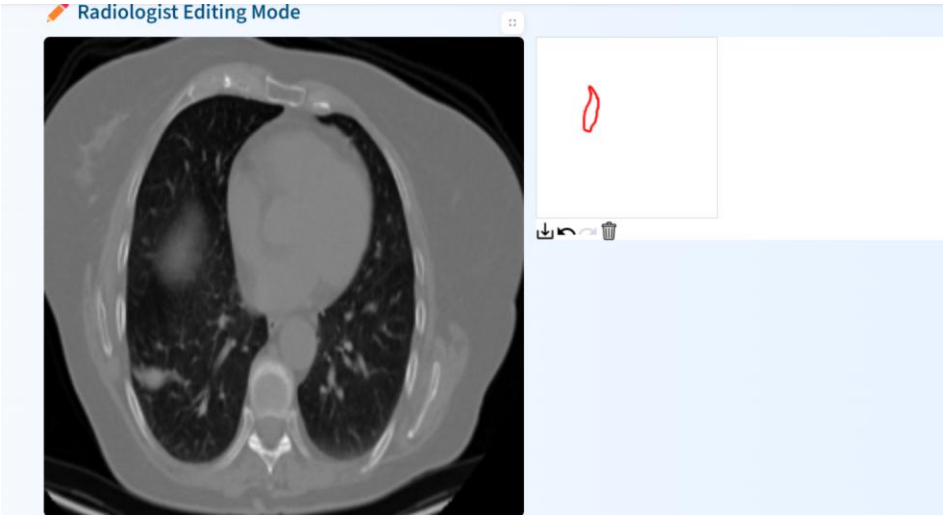


Figure 5.1.6m Radiologist editing mode with manual segmentation

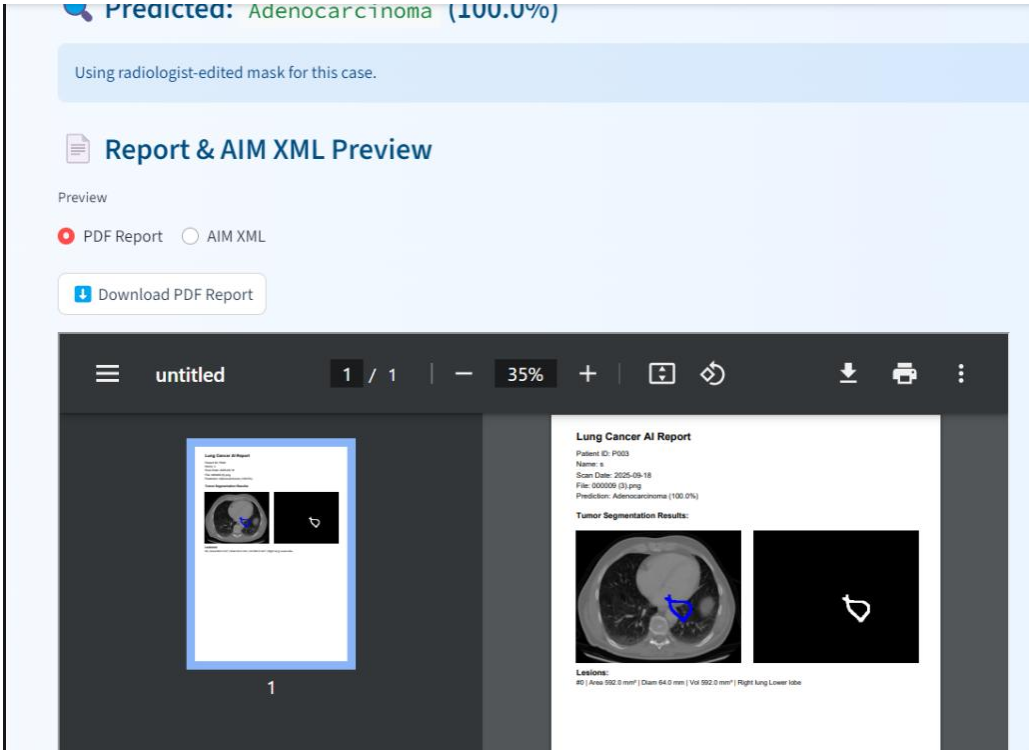


Figure 5.1.6n Revised report using radiologist-edited mask

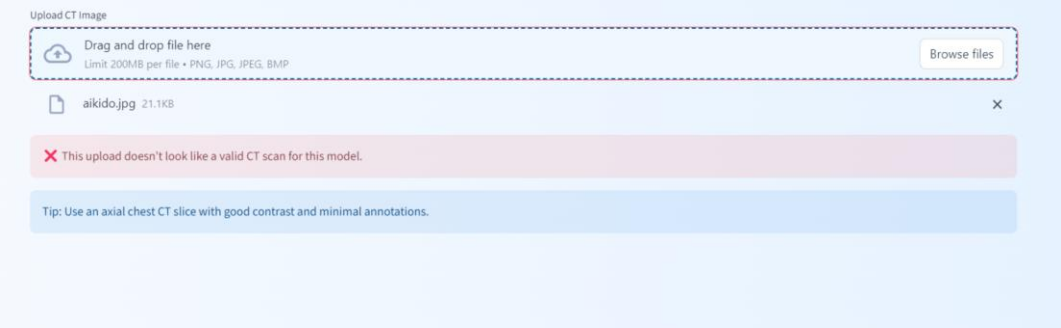


Figure 5.1.6o Error handling for invalid CT scan uploads

Case Browser Module

The Case Browser is the patient case record management system of the Lung Cancer Detection System and serves users to store, retrieve, review, and manage all past patient cases run through the AI pipeline. Its fundamental purpose is to build a structured and searchable repository of predictions, segmentation outputs, and generated reports such that clinical workflows are traceable and reproducible.

As shown in Figure 5.1.6q, the Case Browser presents all previously processed cases in a tabular format, with patient metadata such as Patient ID, name, date of scan, predicted class, confidence score, review status, radiologist comments, and paths to the relevant PDF and AIM XML reports. This structure ensures that every processed scan remains accessible for future verification or research purposes. Users can select one or more cases to perform batch operations, such as downloading reports in bulk as ZIP files (Figure 5.1.6r) or exporting lesion annotation in XML form for integration into PACS systems or research databases (Figure 5.1.6rs.

The module also supports detailed case inspection. By clicking on a specific record, users can view the associated diagnostic report, which includes the predicted cancer type, segmentation overlays, and quantitative lesion measurements (Figure 5.1.6 t). This functionality makes the module a central point for reviewing AI-generated findings alongside traditional radiological workflows.

For administrative purposes, the Case Browser also includes record management features such as selective deletion of patient records and bulk deletion of all cases when necessary. A "Danger Zone" confirmation system is employed to prevent accidental loss of information before performing bulk deletions (Figure 5.1.6u).

A key feature of the module is its filtering feature. Users can search for patient records using Patient ID, filter cases by cancer type (e.g., Adenocarcinoma, Large Cell Carcinoma), or restrict the view by scan date range to concentrate on restricting certain cases of interest (Figure 5.1.6v). This allows efficient navigation within large datasets, which makes it easier to retrieve relevant patient records for diagnostic reading or research analysis.

Overall, the Case Browser module enhances the system by transforming raw predictions into a structured medical archive. It enables radiologists and clinical staff to filter by patient ID, cancer type, or scan date, download standardized reports in bulk, and manage historical case data effectively. By combining case traceability, report accessibility, and safe deletion controls, this module ensures that the system aligns with clinical standards for record keeping and case management.

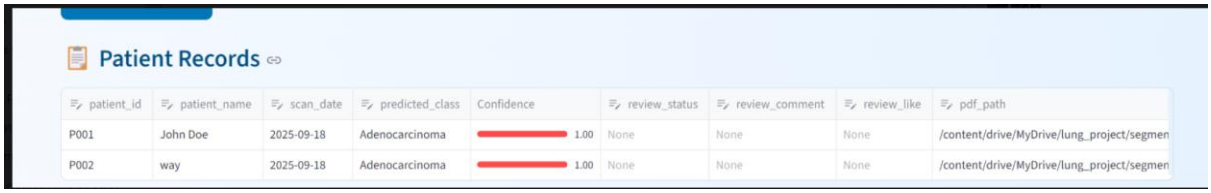


Figure 5.1.6q Patient records table displaying metadata and report paths

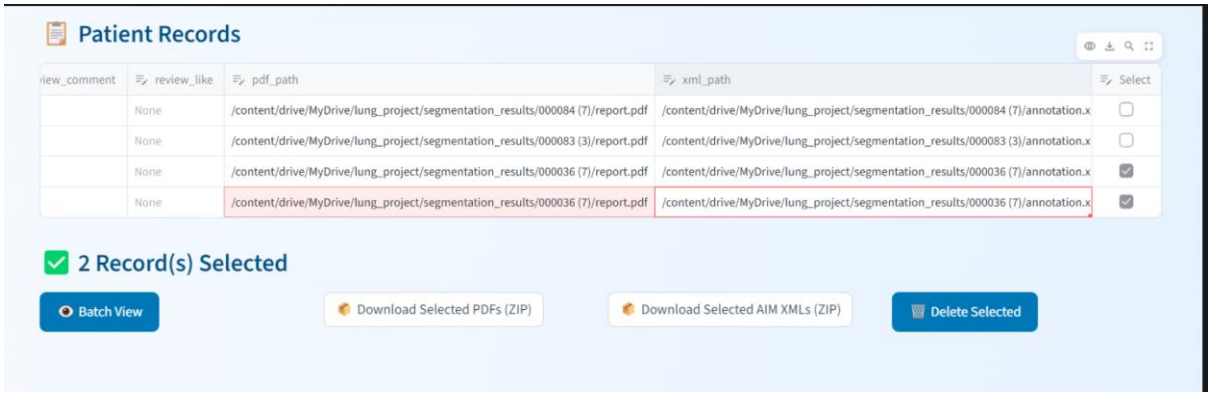


Figure 5.1.6r Batch view and bulk PDF download interface

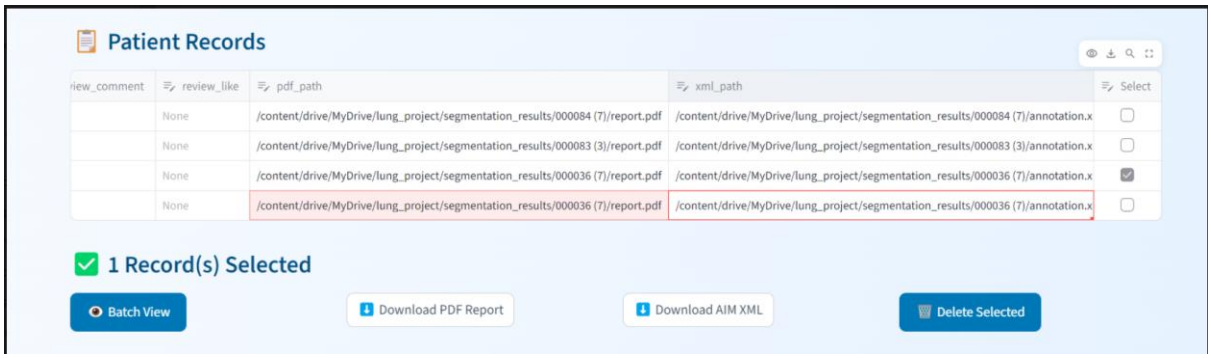


Figure 5.1.6s Example ZIP export of AIM XML annotations

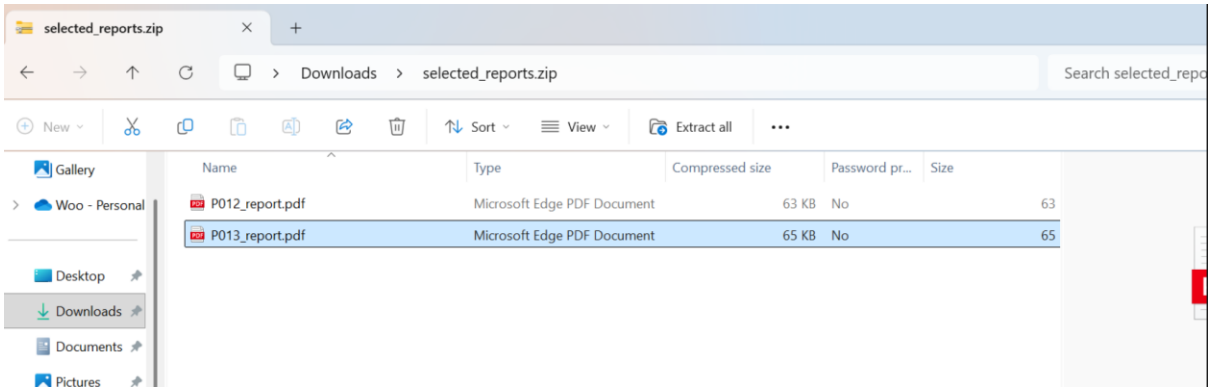


Figure 5.1.6 t Example diagnostic reports retrieved from Case Browser

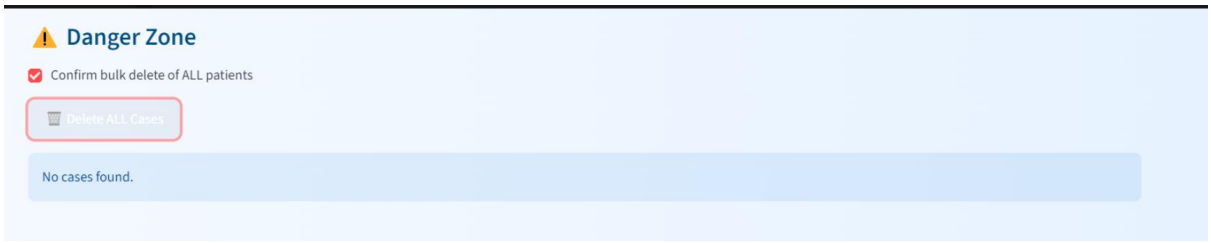


Figure 5.1.6 u Danger Zone interface for safe bulk deletion of all cases

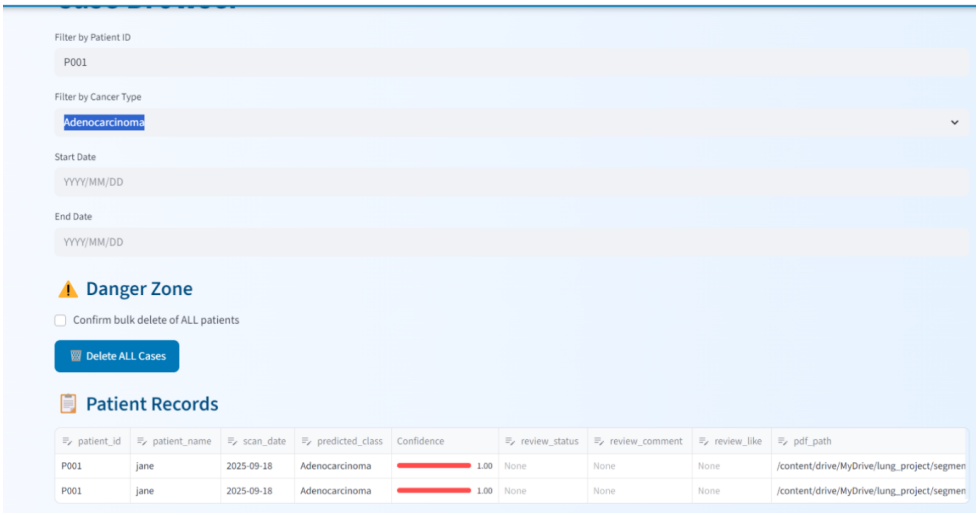


Figure 5.1.6v Filtering options by Patient ID, Cancer Type, and Date Range

Database Implementation

SQLite was selected as the database solution for case storage and retrieval due to its file-based and lightweight nature and the fact that it can be readily integrated into the Streamlit framework. The database schema was set up to store key patient and prediction information such as Patient ID, Patient Name, Scan Date, Predicted Class, Confidence Score, Review Status, Radiologist Comments, and filenames of the generated PDF and AIM XML reports. Each time a new prediction is processed by the system, the corresponding record is automatically inserted into the SQLite database, ensuring that all diagnostic outputs are traceable and persist beyond a single session. The database also supports the fundamental retrieval operations that are built into the Case Browser module for filtering previous cases by Patient ID, cancer type, or scan date range, and viewing or downloading reports for a patient or in bulk. Administrative functions such as record deletion are also supported with safeguards such as the "Danger Zone" warning feature to prevent deletion of case records in error. While SQLite was sufficient for this proof-of-concept system, particularly for local deployments and small-to-medium datasets, a more robust database infrastructure such as a full-scale SQL server or direct PACS integration would be required in clinical settings to handle larger imaging datasets and concurrent multi-user access.

Chapter 6

System Evaluation and Discussion

This chapter offers the evaluation and discussion of the proposed Lung Cancer Detection System to determine whether it meets the requirements and goals that were set earlier in the project. The chapter begins with the system testing plan and performance metrics used to evaluate all modules of the pipeline, including classification, segmentation, reporting, management of databases, workflow for the dashboard, and review by radiologists. It then outlines the test environment and results, giving both quantitative metrics, such as classification accuracy and cross-validation score, and qualitative metrics, such as segmentation overlay and correctness of the dashboard functions. The chapter then gives an account of the problems encountered in preparing data, modelling, and deploying, and evaluates to what extent the problems were successfully resolved. An in-depth objectives assessment is then given, relating every project goal to its corresponding outcomes. Finally, the chapter concludes with a summarizing remark on the overall performance of the system, referring to its strengths, weaknesses, and potential areas for improvement in the future.

6.1 System Testing and Performance Metrics

To ensure that the proposed Lung Cancer Detection System met its intended objectives, system testing was conducted on all the major components of the pipeline, i.e., the classification models, the segmentation module, the automated reporting features, the case storage/retrieval system, the dashboard workflow, and the radiologist review functionality. Each module was tested on relevant metrics and methods to establish both quantitative performance and qualitative accuracy.

For the classification module, two variants of the EfficientNet architecture (B0 and B1) were tested to determine their ability in distinguishing between normal and cancerous CT scans, and between the different subcategories of lung cancer. Among the metrics used for evaluation were standard machine learning metrics of accuracy, precision, recall, and F1-score, along with the use of confusion matrices for analysis at a class level. Furthermore, 5-fold stratified cross-validation was performed on the training and validation sets to offer reliability and generalisation to different data splits. Test set accuracy and loss were also recorded to verify

performance on unseen data. These measures were selected since they provide a sense of both overall performance as well as the system's ability to handle imbalanced or difficult-to-classify examples.

For the segmentation module, testing was necessarily qualitative, as there were no ground truth tumour masks available in the dataset. The Fuzzy C-Means (FCM) segmentation output was visually inspected to see whether or not the tumour candidates discovered were clinically plausible regions. Assessment involved examining overlays, membership maps, binary tumour masks, and centroid markers. This approach ensured that the output from segmentation was interpretable by radiologists without the need for quantitative metrics like Dice similarity or IoU.

The reporting system was tested for accuracy and completeness of the outputs generated. This involved making sure that the PDF reports accurately reflected patient metadata, classification results, segmentation overlays, and lesion measurements, and the AIM XML files contained properly formatted lesion annotations and quantitative attributes. The tests confirmed that both formats were consistent, reproducible, and ready for downstream use, for instance, PACS integration or research data archiving.

For the case storage and management module, functional testing was carried out on the SQLite database implementation for validation of core operations. This included insertion of new case records on prediction, retrieval and filtering of cases by Patient ID, cancer type, and scan date, and deletion of records both selectively and in bulk. Special attention was devoted to testing the "Danger Zone" bulk deletion protection to ensure that accidental data loss was not feasible without explicit user confirmation. These tests confirmed that the database afforded reliable storage, easy retrieval, and secure deletion of past cases, and thereby assured traceability and accountability of the diagnostic process.

Dashboard-level testing was also conducted to ensure workflow functionality in various scenarios. Specifically, when a scan was normal, the segmentation module was skipped and a simplified report with no tumours was generated. When an invalid file had been uploaded (for example, a non-CT image), an error message asking the user to upload a valid medical image

was displayed by the system. These tests ensured that the dashboard provided robust input validation and consistent workflow outputs.

Finally, the radiologist review functionality was tested to ensure proper integration of human oversight into the AI workflow. Three scenarios were validated: (i) when the radiologist accepted the AI result, the case was marked as accepted and stored without modification; (ii) when the radiologist rejected the AI result, the case was tagged as rejected with the radiologist's comments recorded in the database; and (iii) when the radiologist chose to manually edit a tumour mask, the revised mask was saved, and updated PDF/XML reports were generated to reflect the corrections. These tests confirmed that the system effectively supported collaborative AI–human decision making, improving clinical accountability and reliability.

In summary, the system was tested through a combination of quantitative metrics (accuracy, precision, recall, F1-score, cross-validation performance, test set accuracy) and qualitative/functional checks (visual overlays, report correctness, database operations, workflow handling, and radiologist review). This combination of methods was chosen to capture both the statistical performance of the deep learning models and the practical usability of the deployed system in a simulated clinical workflow. Table 6.1 shows the summary of test cases.

Test Case	**Input Type**	**Expected Behaviour**
Classification – Normal Scan	Valid CT scan with no tumour	System predicts "Normal", segmentation skipped, simplified report generated.
Classification – Cancer Scan	Valid CT scan with tumour	System predicts cancer subtype, segmentation triggered, lesion mask/overlays produced.
Invalid CT Scan Upload	Non-medical image or unsupported file type	System rejects input, displays error message, prompts correct file upload
Report Generation	Valid classified case (Normal or Cancer)	PDF and AIM XML reports generated with correct metadata and outputs.

Database – Insert	New case prediction	Patient details, predictions, and report paths stored in SQLite database
Database – Retrieve/Filter	Query by Patient ID, cancer type, or scan date	Correct records retrieved and displayed in Case Browser.
Database – Delete (Selective)	Delete individual patient record	Selected case removed from database, others unaffected.
Database – Delete (Bulk)	Activate "Danger Zone" bulk delete	Confirmation required, all cases deleted only after user approval.
Dashboard Workflow	User uploads CT scan with patient details	Predictions linked to patient record; workflow consistent end-to-end.
Radiologist Review – Accept Result	Radiologist accepts AI prediction	Case marked as "Accepted"; no changes made; stored report remains final.
Radiologist Review – Reject Result	Radiologist rejects AI prediction	Case marked as "Rejected"; radiologist comment stored; report updated accordingly.
Radiologist Review – Reject Result	Radiologist rejects AI prediction	Case marked as "Rejected"; radiologist comment stored; report updated accordingly.

Table 6.1 Summary of System Test Cases

6.2 Testing Setup and Result

6.2.1 Classification Testing Result Module

For the classification testing, the outcomes were both EfficientNetB0 and EfficientNetB1 could distinguish between the normal and cancer CT scans, as well as classify the three subtypes of lung cancers. As can be seen in Figure 5.1.3 and Figure 5.1.4, EfficientNetB0 achieved a best validation accuracy of 90.28% at epoch 13, while EfficientNetB1 achieved a best validation accuracy of 95.83% at epoch 20. Despite the higher validation accuracy of EfficientNetB0, its training curves were more fluctuating in terms of epochs, indicating less stability in the process of convergence. EfficientNetB1, on the other hand, experienced smoother training dynamics and more uniform generalisation across folds.

Class-level performance was further examined using confusion matrices and class reports. EfficientNetB0 had satisfactory precision and recall metrics for all classes but experienced some misclassification in the squamous cell carcinoma class (Figure 5.1.5a). EfficientNetB1, although lower in peak validation accuracy, offered a more balanced classification performance for all four classes, as evidenced by Figure 5.1.5b.

To further validate consistency of models, 5-fold stratified cross-validation was conducted over the combined training and validation data sets. The results, as combined in Figure 5.1.5c, showed that EfficientNetB0 had a mean accuracy of 96.85% ± 0.0140 and EfficientNetB1 had a mean accuracy of 97.04% ± 0.0302. These results indicated EfficientNetB1, although slightly lower in one-off validation, yielded greater consistency and generalisation across different data partitions.

On the holdout test set, the last model selected EfficientNetB1 had a test accuracy of 94.60% with test loss = 0.227, confirming its performance on unseen data classification. The stability of EfficientNetB1 across folds and its reliable test performance justified its selection as the primary classifier for the system.

6.2.2 Segmentation Testing Result Module

The segmentation module provided output that was consistent with the intended behavior set during system testing. With the enhanced Fuzzy C-Means (FCM) clustering algorithm, the system now generated four outputs for each scan: (i) the raw CT image, (ii) the membership map with tumour boundaries, (iii) the binary tumour mask, and (iv) an overlay image with centroid dots for readability, as presented in Figure 5.1.5e.

On visual inspection, overlays always detected suspicious tumour regions in the lung fields. Artefacts and background noise were removed in a few cases, leaving only clinically plausible tumour candidates that aligned with expected anatomical locations. Centroid markers provided more information to allow radiologists to readily identify the central location of the segmented lesion.

Despite the lack of ground truth segmentation masks to compute quantitative performance metrics such as Dice similarity or Intersection-over-Union (IoU), the qualitative evaluation ensured segmentation outputs were clinically meaningful and complemented the classification outputs. Localizing tumour candidates only when cancer was detected, the module demonstrated its effectiveness as a conditional interpretability layer, adding transparency and diagnostic support to the overall pipeline.

6.2.3 Dashboard Testing Result Module

The dashboard module was tested to ensure smooth end-to-end workflow and user interaction across different scenarios, including normal scans, cancerous scans, invalid uploads, report generation, database operations, and radiologist review.

For the normal scan classification test, input was a correct CT scan without any tumours. The expected behavior was for the system to label the scan "Normal," skip segmentation, and generate a simple report confirming there were no tumours. The resulting output was as expected, with the dashboard skipping segmentation correctly and generating the appropriate report output, confirming that the workflow handled non-cancer cases as anticipated as shown in figure 6.2.3a.

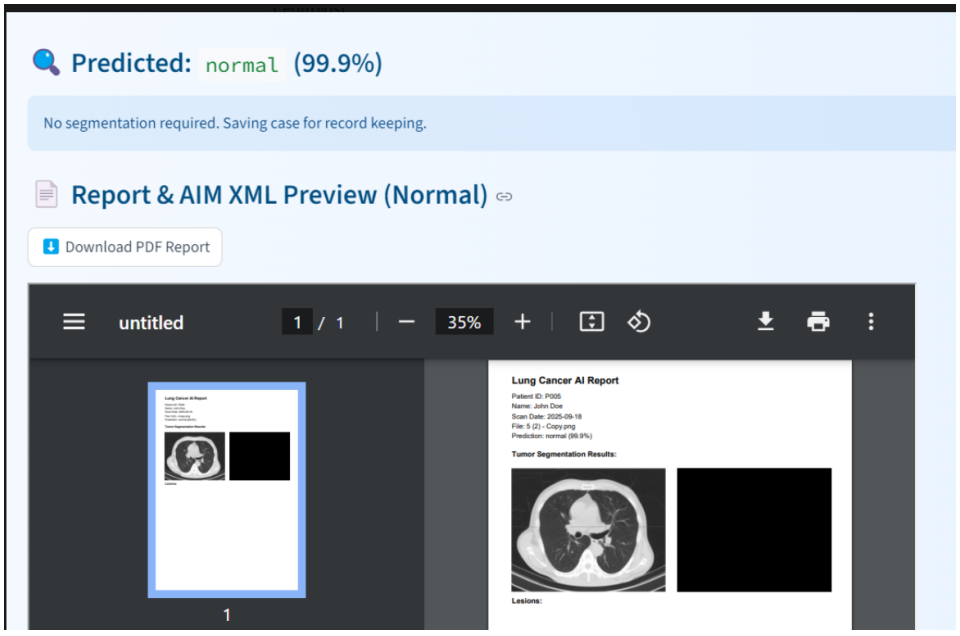


Figure 6.2.3a Normal case

For the classification test on cancerous scans, the input type was a valid CT scan with visible tumour features. The expected behaviour was for the system to predict the correct cancer subtype, trigger the segmentation module, and generate tumour masks, overlays, and lesion measurements as shown in figure 6.2.3b, figure 6.2.3c and figure 6.2.3d. The actual result fulfilled this expectation, as the system produced accurate subtype predictions, visually interpretable overlays, and complete lesion measurement outputs.

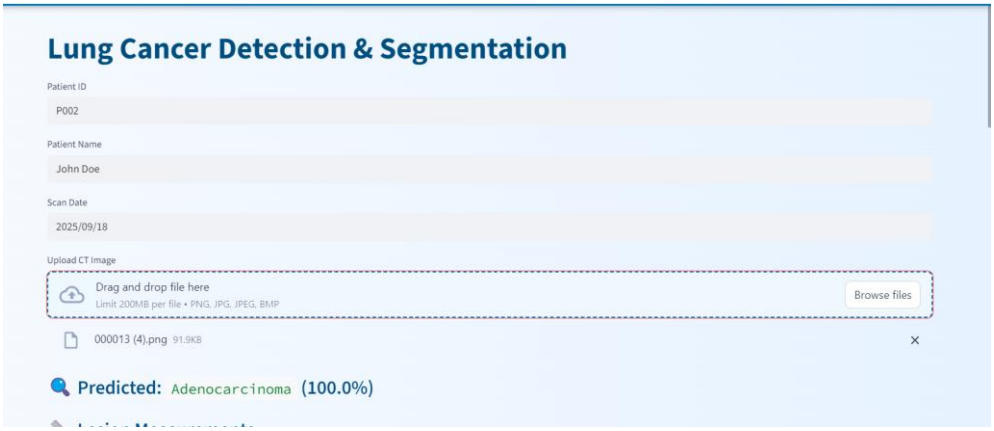


Figure 6.2.3b Prediction of the cancer type

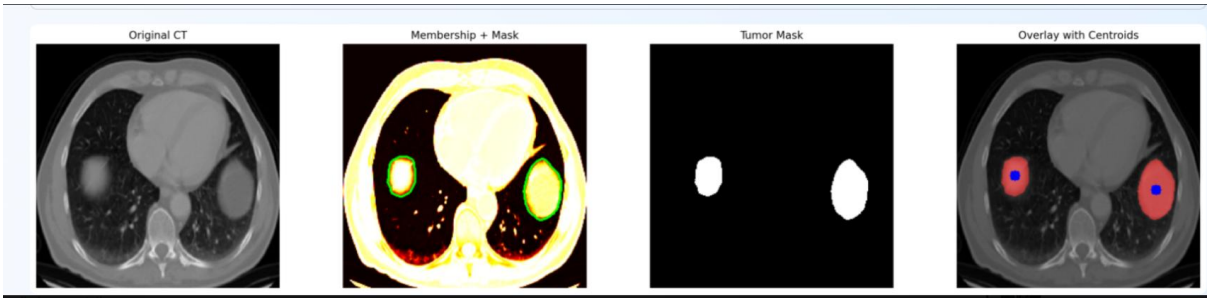


Figure 6.2.3c Segmentation output

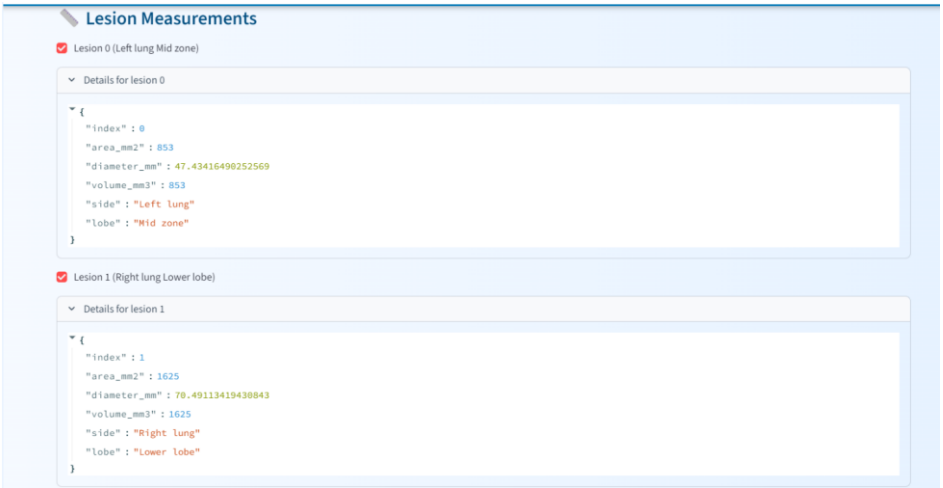


Figure 6.2.3d Lesion Measurements

For improper upload of CT scans, when the input was an unsupported file type or non-medical image, the expected behavior was for the system to reject the file and give notification to the

user in the form of an error message. The actual result was as expected since the dashboard displayed a warning message with instructions to upload the correct CT image, with robust input validation as shown in figure 6.2.3e.

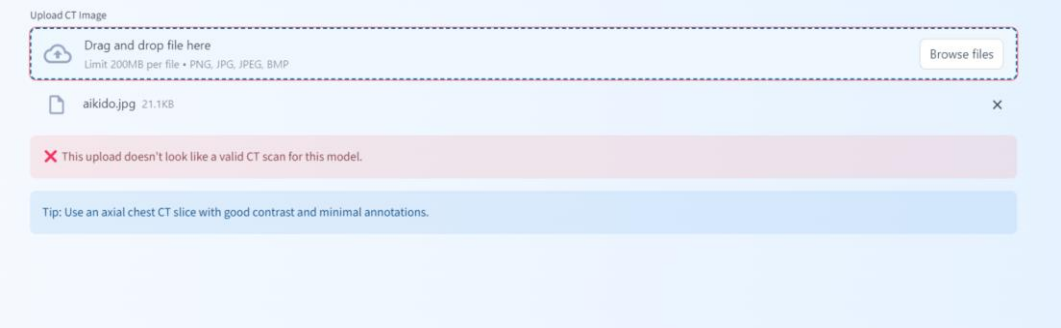


Figure 6.2.3e Invalid CT Scan Case

For the report generation test, valid classified cases (both normal and cancerous) were used as input. The expected behaviour was the generation of both PDF and AIM XML reports containing patient details, classification results, segmentation visualisations, and lesion measurements where applicable. The actual result met these requirements, with reports generated accurately and in the correct format, confirming that the reporting feature was reliable as shown in figure 6.2.3f-g.

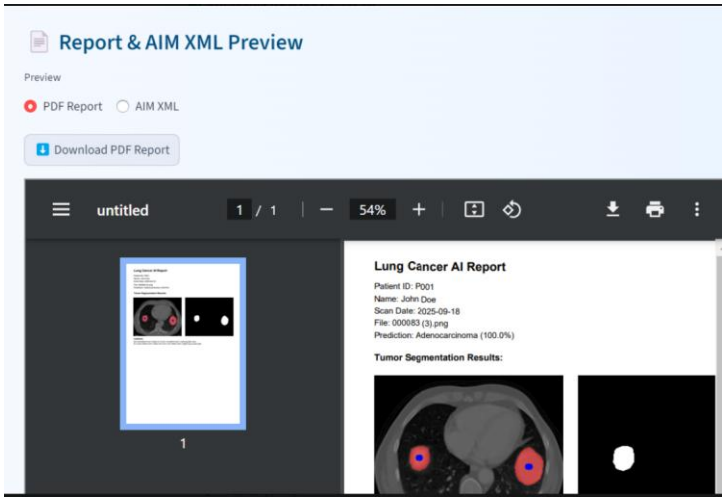


Figure 6.2.3f Lung Cancer PDF Report

Figure 6.2.3g Lung Cancer AIM XML

With regard to the database operations, new prediction records were inserted into the SQLite database. What was anticipated was that all patient data, predictions, and report file paths would be permanently stored. The actual result confirmed this, as entries appeared immediately in the Case Browser and could be retrieved later as shown in figure 6.2.3h. Filtering tests, using Patient ID, cancer type, and scan date, returned correct and relevant records, fulfilling the expectation as shown in figure 6.2.3i. Also, deletion tests confirmed that selective record deletions affected only the desired entries as shown in figure 6.2.3j-k, while bulk deletions caused the "Danger Zone" confirmation window before deleting all cases as shown in figure 6.2.3l. These findings were consistent with the expected behavior, ensuring traceability and safety.

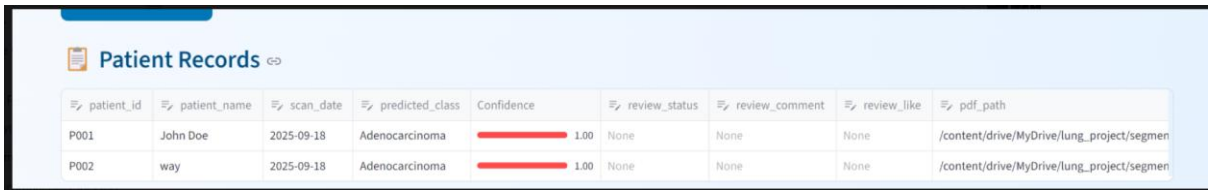


Figure 6.2.3h Patient Records Stored

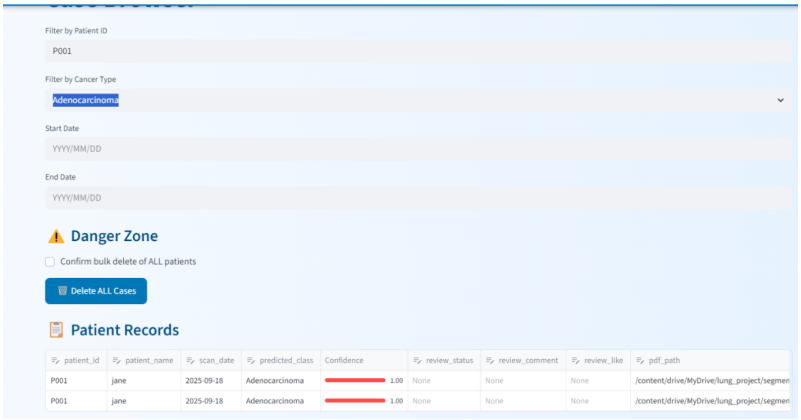


Figure 6.2.3i Filter Case

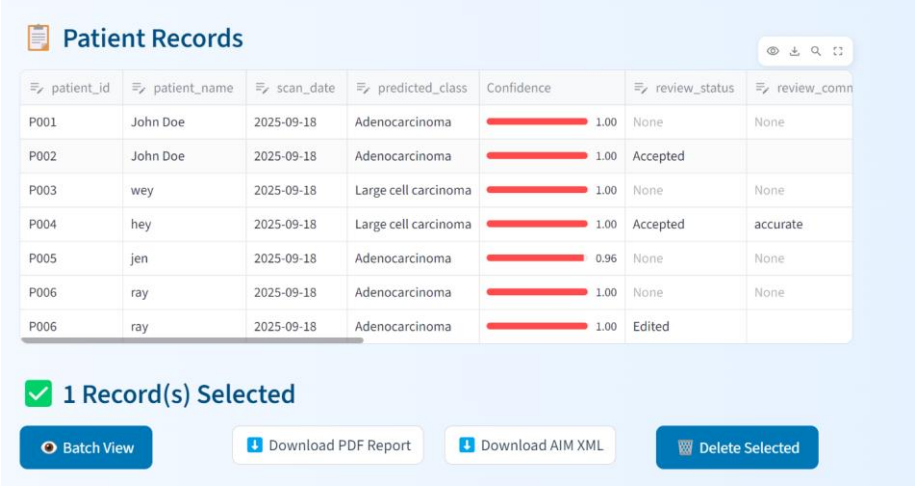


Figure 6.2.3j P001 record exists before deleted

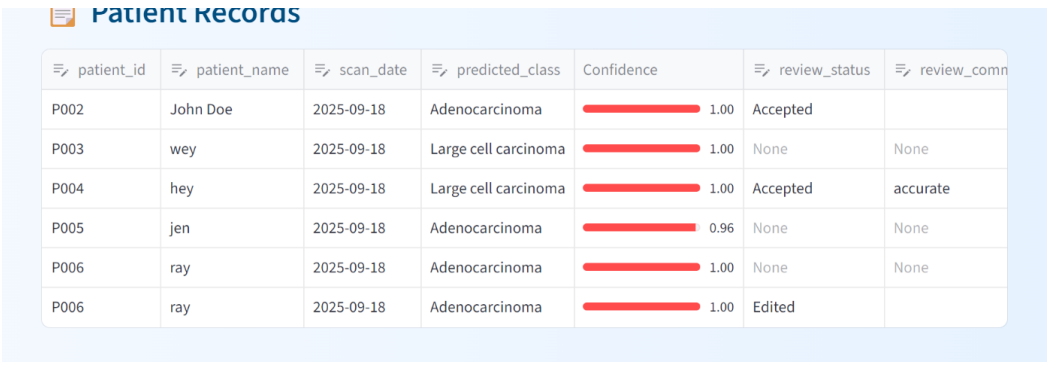


Figure 6.2.3k P001 record deleted

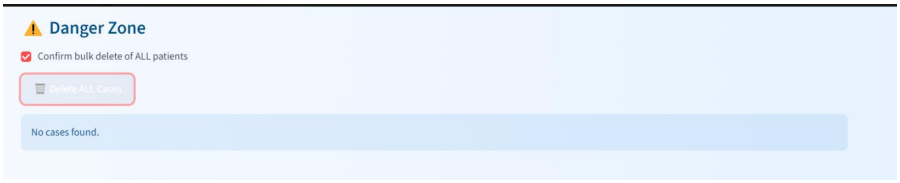
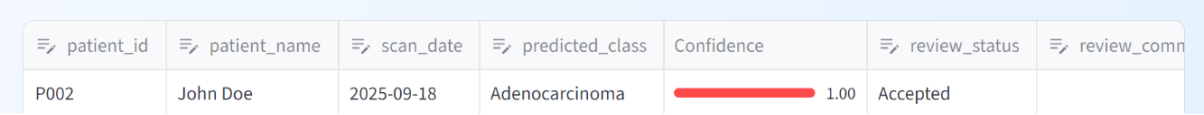


Figure 6.2.3l All patient records deleted

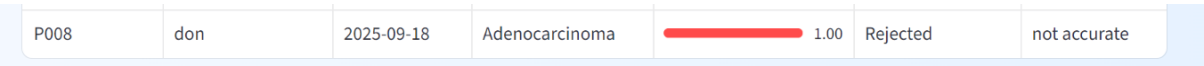
For the testing of the dashboard workflow, when patient data and CT scans were uploaded, the predictions were always correlated with the correct metadata, and the end-to-end process—from upload to report generation—operated flawlessly. Both normal and cancer workflows executed as expected, and the invalid inputs were correctly prevented, ensuring that the user interface was fault-free.

Finally, for the radiologist review section, three test scenarios were conducted. In the scenarios in which the radiologist approved the AI prediction, the case was saved in the accepted state without modification, as required (figure 6.2.3m). If the radiologist did not approve the prediction, the system correctly logged the rejection along with the radiologist comments as shown in figure 6.2.3n. For manual tumour mask amendments, the required behaviour was that the modified mask would be saved and be included in updated PDF and XML reports as shown in figure 6.2.3o-p. The actual results confirmed that these updates were properly saved and integrated, demonstrating that the system worked well in enabling collaborative AI–human decision-making.



patient_id	patient_name	scan_date	predicted_class	Confidence	review_status	review_comn
P002	John Doe	2025-09-18	Adenocarcinoma	1.00	Accepted	

Figure 6.2.3m Approved status



P008	don	2025-09-18	Adenocarcinoma	1.00	Rejected	not accurate

Figure 6.2.3n Rejected status with comment

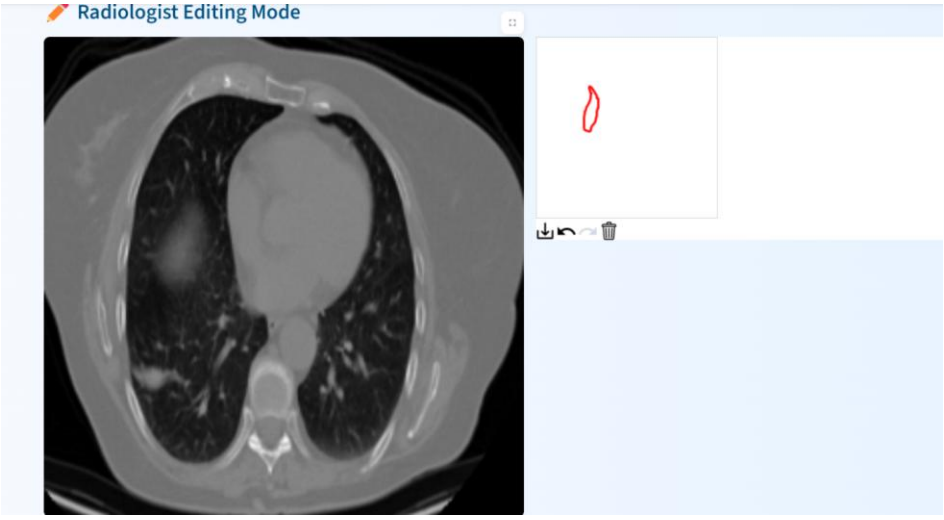


Figure 6.2.3o Editing Mode

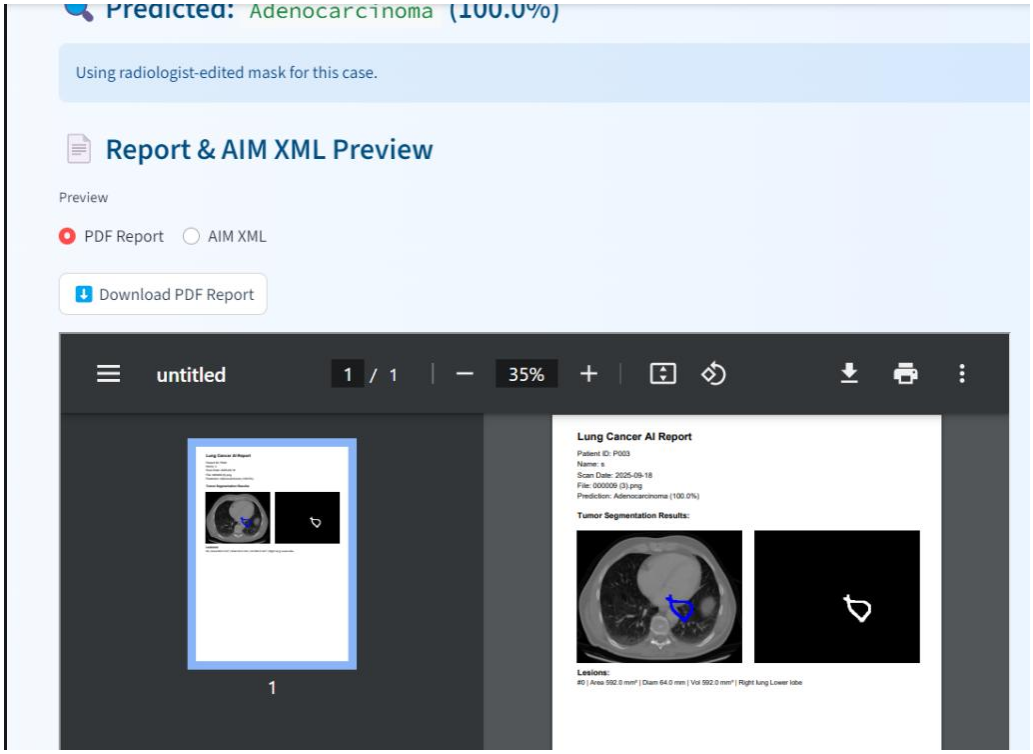


Figure 6.2.3p Updated reports with edited mask

In general, across all test scenarios—spanning from classification and segmentation to reporting, database management, dashboard workflow, and radiologist review—the actual results were consistent with the anticipated behaviours. This confirmed that the system was functionally complete, robust, and consistent with its clinically designed workflow model. Table 6.2 shows the summary of test results.

Test Case	**Input Type**	**Expected Behaviour**	**Actual Result**
Classification – Normal Scan	Valid CT scan with no tumour	System predicts "Normal", segmentation skipped, simplified report generated.	EfficientNetB1 correctly classified normal scans; segmentation skipped; simplified reports produced as expected.

Classification – Cancer Scan	Valid CT scan with tumour	System predicts cancer subtype, segmentation triggered, lesion masks/overlays generated.	EfficientNetB1 correctly classified cancer cases; FCM segmentation triggered; outputs (mask, overlay, centroids) successfully generated.
Invalid CT Scan Upload	Non-medical image/unsupported file	System rejects input, displays error message, prompts correct upload.	Error message displayed; invalid scans rejected; workflow maintained
Report Generation	Valid classified case	PDF and AIM XML reports generated with correct metadata and outputs.	Reports consistently generated; PDF contained classification + segmentation results; XML contained lesion annotations.
Database – Insert	New case prediction	Patient details, predictions, and report paths stored in SQLite database.	Records inserted correctly; all fields (patient ID, name, scan date, predictions, report paths) stored.
Database – Retrieve/Filter	Query by Patient ID, cancer type, or scan date	Correct records retrieved and displayed in Case Browser.	Queries returned correct cases; filtering by ID, cancer type, and date worked as expected.
Database – Delete (Selective)	Delete individual patient record	Selected case removed from database, others unaffected.	Individual deletions successful; no impact on unrelated records.

Database – Delete (Bulk)	Activate "Danger Zone" bulk delete	Confirmation required; all cases deleted only after approval.	Confirmation prompt worked; all cases deleted only after explicit user confirmation.
Dashboard Workflow	User uploads CT scan with details	Predictions linked to patient record; workflow consistent end-to-end.	Dashboard linked patient info to predictions; workflow operated smoothly for both normal and cancer scans.
Radiologist Review – Accept Result	Radiologist accepts AI prediction	Case marked as "Accepted"; stored report finalised.	Acceptance recorded; report stored as final version.
Radiologist Review – Reject Result	Radiologist rejects AI prediction	Case marked as "Rejected"; radiologist comments stored in database.	Rejections recorded correctly; comments stored and linked to case.
Radiologist Review – Manual Editing	Radiologist edits tumour mask	Edited mask saved; revised PDF/XML reports generated with updates.	Manual edits saved successfully; updated reports generated and stored.

Table 6.2 Summary of Test Results

6.3 Project Challenges

During the development of the system, several challenges were encountered at the data, modelling, and deployment stages. The first important challenge was that ground truth segmentation masks did not exist in the dataset, so quantitative metrics such as Dice similarity or Intersection-over-Union (IoU) could not be utilized to evaluate the segmentation module. This required reliance on qualitative visual inspection, which, although clinically plausible, limited the ability to quantify segmentation accuracy.

Another difficulty was class imbalance within the dataset. Although the dataset overall was balanced across the four classes, some subtypes such as squamous cell carcinoma and normal cases had fewer training examples. This imbalance would have affected classification performance, and the necessity to use augmentation strategies and stratified k-fold cross-validation to improve generalisation.

Deployment challenges included integrating four elements—classification, segmentation, reporting, and database storage—into a single Streamlit dashboard without affecting its robustness and usability. Error handling had to be explicitly implemented to handle invalid inputs and system crashes. Additionally, proof-of-concept storage was implemented using SQLite, but it is not designed for large-scale clinical environments, meaning future migration to a more robust database or PACS integration would be necessary.

6.4 Objectives Evaluation

The evaluation of project outcomes against the stated objectives confirmed that all goals were successfully achieved:

1. **EfficientNetB1 Classification Model**
   Objective: To develop a lung cancer classification model using EfficientNetB1 that accurately distinguishes between normal and various types of cancerous chest CT scans.
   Result: Achieved with a final independent test accuracy of 94.60%, confirming the model's ability to accurately differentiate normal scans from cancerous cases, including subtypes.
2. **Comparison with EfficientNetB0**
   Objective: To evaluate and compare the classification performance of EfficientNetB1 against a lighter baseline model (EfficientNetB0) using K-fold stratified cross-validation.
   Result: Successfully performed. EfficientNetB0 achieved a mean accuracy of 96.85% ± 0.0140, while EfficientNetB1 achieved 97.04% ± 0.0302, demonstrating that EfficientNetB1 generalised more consistently across folds despite slightly lower single-run validation accuracy.
3. **Modified Fuzzy C-Means Segmentation**
   Objective: To implement a modified Fuzzy C-Means (FCM) segmentation module to generate tumour masks only when cancer is detected.
   Result: Successfully implemented. For cancerous CT scans, the system generated tumour masks, overlays, and centroid markers, providing interpretable localisation without unnecessary computation for normal cases.
4. **Interactive Streamlit Dashboard**
   Objective: To develop an interactive Streamlit-based medical dashboard that supports lung cancer prediction, tumour segmentation, lesion measurement, radiologist review, and automated reporting for streamlined clinical decision support.
   Result: Fully developed and deployed. The dashboard integrated all modules into a seamless clinical simulation environment, enabling classification, segmentation, lesion quantification, radiologist review, and automated report generation.

5. **Tumour Visualisation and AIM XML Annotation**
   Objective: To visualise tumour segmentation by overlaying masks on CT scans and generate structured Annotation Image Markup (AIM) XML annotations for potential clinical integration.
   Result: Achieved successfully. The system generated segmentation overlays, binary masks, membership maps, and structured AIM XML annotations, enabling interpretability and potential PACS integration.
6. **SQLite-Based Case Storage**
   Objective: To implement an SQLite-based case storage system that records patient details, predictions, radiologist reviews, and generated reports for easy retrieval.
   Result: Implemented successfully. The system persistently stored case records with full functionality for retrieval, filtering, and safe deletion, ensuring case traceability and accountability in the diagnostic workflow.

6.5 Concluding Remark

This project was successful in developing and deploying a classification-first pipeline for the identification of lung cancer from CT scans using the integration of EfficientNetB1-based classification, conditional FCM-based segmentation, structured reporting, and case management on a Streamlit dashboard. The system demonstrated excellent classification accuracy, radiologist-interpretable segmentation overlays, and workflow functions such as PDF/XML reporting and integration of radiologist review.

While the system performed satisfactorily as a proof-of-concept, defects such as the absence of annotated segmentation ground truth and reliance on minimalist SQLite database suggest areas of future improvement. Scaling to larger annotated sets, incorporating quantitative segmentation evaluation, and migration to enterprise-class PACS or SQL environments would make the system more scalable and clinically feasible.

In short, the system exhibited strong technical feasibility, addressed gaps in existing CAD pipelines, and provided an efficient, modular, and clinically relevant platform for supporting radiologists in early lung cancer detection and decision-making.

Chapter 7

Conclusion and Recommendation

This chapter provides the concluding remarks and recommendations derived from the development, implementation, and evaluation of the proposed Lung Cancer Detection System. Section 7.1 presents the overall conclusion of the project by summarising its achievements, outcomes, and contributions to the field of computer-aided diagnosis. Section 7.2 outlines several recommendations for future improvements, focusing on technical enhancements, dataset considerations, clinical integration, and scalability for real-world deployment. Together, these sections highlight the extent to which the project objectives were fulfilled while also identifying opportunities for further refinement and extension of the system.

7.1 Conclusion

This project has effectively developed and deployed a classification-first pipeline for detecting and localising lung cancer from CT scans. The system combined EfficientNetB1-based classification with a conditional Fuzzy C-Means (FCM) segmentation module, enabling both accurate cancer detection and interpretable tumour localisation. Deployed in a Streamlit dashboard, the system supported end-to-end workflows like case prediction, lesion measurement, radiologist review, automatic report generation, and SQLite-based case storage.

The performance results indicated that EfficientNetB1 achieved a final standalone test accuracy of 94.60% with consistent performance between folds as verified by validation through 5-fold cross-validation. Although EfficientNetB0 was slightly better in single-run validation accuracy, its training was not as stable, so EfficientNetB1 was the more stable choice for deployment. The segmentation module, while qualitatively evaluated in the absence of ground-truth masks, produced overlays, binary masks, and centroid annotations that were clinically plausible and gave interpretability to classification results.

Functionality testing also guaranteed that the dashboard behaved consistently across different situations. Normal scans were correctly classified and skipped unnecessary segmentation, cancerous scans triggered segmentation with accurate overlays and lesion measurements, and invalid uploads were correctly rejected with error messages. Reporting features generated PDF

and AIM XML formats consistently, and case browser facilitated insertion, retrieval, filtering, and secure removal of patient records firmly. Radiologist review workflow encouraged accountability with the facility to accept, reject, or edit manual masks, and amended reports were generated automatically.

Overall, the project delivered a lightweight, modular, and clinically feasible proof-of-concept system that addressed significant limitations of existing CAD pipelines by avoiding redundant computation, preserving interpretability, and promoting collaborative AI–human decision-making. While not yet deployable for immediate clinical application, the system demonstrated high technical feasibility and set a solid foundation for further research and development.

7.2 Recommendation

Although the project successfully achieved its objectives, several areas for improvement and future work are recommended to strengthen both technical performance and clinical applicability. One of those is supplementing the annotated segmentation dataset because the current dataset lacked ground-truth tumour masks, which limited evaluation to qualitative visual inspection. Supplementing such resources as LIDC-IDRI or MSD Lung Tumour would allow quantitative validation through such measures as Dice similarity and IoU. Scaling up from 2D slice-by-slice analysis to 3D volumetric segmentation would also improve tumour localisation and volumetric measurement and render the system clinically relevant to treatment planning. Infrastructure level, while SQLite was sufficient for proof-of-concept, larger clinical deployments will require transitioning to enterprise-grade databases such as SQL Server or PostgreSQL, or PACS integration to support bigger data sets, multi-user environments, and real-time hospital usage. For deployability and scalability purposes, the system could be designed for cloud infrastructure with GPU acceleration support, remote access, and integration with hospitals, and containerization (e.g., Docker) would ensure portability and big-data deployment. A next important step would be to improve model generalisation via training and validation across a set of, ideally multi-institutional, varied datasets to minimize variability dependence on imaging hardware, acquisition protocols, and patient demographics. More advanced techniques such as transfer learning and federated learning would be capable of increasing generalisability even more without the need for centralized data sharing. From an ease-of-use perspective, the user interface can be enhanced by real-time heatmaps, case

comparison features, and audit trails for radiologist modifications so the system would become even closer to clinical decision-support requirements. Finally, for clinical and regulatory approval, subsequent studies would involve clinical trials and medical device regulations such as FDA or CE certification and incorporating explainable AI (XAI) techniques to further increase transparency, interpretability, and trust by clinicians.

REFERENCES

[1]
R. Raza et al., “Lung-EffNet: Lung cancer classification using EfficientNet from CT-scan images,” Engineering Applications of Artificial Intelligence, vol. 126, pp. 106902–106902, Nov. 2023, doi: https://doi.org/10.1016/j.engappai.2023.106902.

[2]
R. Javed, T. Abbas, Ali Haider Khan, A. Daud, A. Bukhari, and Riad Alharbey, “Deep learning for lungs cancer detection: a review,” Artificial Intelligence Review, vol. 57, no. 8, Jul. 2024, doi: https://doi.org/10.1007/s10462-024-10807-1.

[3]
P. Srivastava, A. R. Mishra, and S. S. Chauhan, “Deep Learning-Based Lung Cancer Detection Enhancing Early Diagnosis and Treatment Outcomes,” Mar. 2024, doi: https://doi.org/10.1109/icdt61202.2024.10489161.

[4]
M. Wahengbam, M. Sriram and T. G. Singh, "Deep Learning-based Early Lung Cancer Detection using Multi-Model Image Fusion Technique," 2023 2nd International Conference on Automation, Computing and Renewable Systems (ICACRS), Pudukkottai, India, 2023, pp. 884-889, doi: 10.1109/ICACRS58579.2023.10405217.

[5]
“Chest CT-Scan images Dataset,” *www.kaggle.com*. https://www.kaggle.com/datasets/mohamedhanyyy/chest-ctscan-images

[6]
T. Kadir and F. Gleeson, “Lung cancer prediction using machine learning and advanced imaging techniques,” *Translational Lung Cancer Research*, vol. 7, no. 3, pp. 304–312, Jun. 2018, doi: https://doi.org/10.21037/tlcr.2018.05.15.

[7]

F. Ge, Y. Zhang, J. Xu, A. Muhammad, J. Song, and D.-J. Yu, “Prediction of disease-associated nsSNPs by integrating multi-scale ResNet models with deep feature fusion,” *Briefings in Bioinformatics*, vol. 23, no. 1, Nov. 2021, doi: 10.1093/bib/bbab530.

[8]
W. Sun, B. Zheng, and W. Qian, “Computer aided lung cancer diagnosis with deep learning algorithms,” *Proceedings of SPIE, the International Society for Optical Engineering/Proceedings of SPIE*, vol. 9785, p. 97850Z, Mar. 2016, doi: 10.1117/12.2216307.

[9]
S. Wang, L. Dong, X. Wang, and X. Wang, “Classification of pathological types of lung cancer from CT images by deep residual neural networks with transfer learning strategy,” *Open Medicine*, vol. 15, no. 1, pp. 190–197, Jan. 2020, doi: 10.1515/med-2020-0028.

[10]
T. Sajja, R. Devarapalli, and H. Kalluri, “Lung cancer detection based on CT scan images by using deep transfer learning,” *Traitement Du Signal*, vol. 36, no. 4, pp. 339–344, Oct. 2019, doi: 10.18280/ts.360406.

[11]
M. Tan and Q. V. Le, “EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks,” in *Proceedings of the 36th International Conference on Machine Learning (ICML)*, Long Beach, CA, USA, 2019, pp. 6105–6114.

[12]
M. Humayun, R. Sujatha, S. N. Almuayqil, and N. Z. Jhanjhi, “A Transfer Learning Approach with a Convolutional Neural Network for the Classification of Lung Carcinoma,” *Healthcare*, vol. 10, no. 6, p. 1058, Jun. 2022, doi: 10.3390/healthcare10061058.

[13]

H. F. Al-Yasriy, M. S. Al-Husieny, F. Y. Mohsen, E. A. Khalil, and Z. S. Hassan, “Diagnosis of lung cancer based on CT scans using CNN,” *IOP Conference Series Materials Science and Engineering*, vol. 928, no. 2, p. 022035, Nov. 2020, doi: 10.1088/1757-899x/928/2/022035.

[14]
S. Sivakumar and C. Chandrasekar, “Lung Nodule Segmentation through Unsupervised Clustering Models,” *Procedia Engineering*, vol. 38, pp. 3064–3073, Jan. 2012, doi: 10.1016/j.proeng.2012.06.357.

[15]
Q. Song, L. Zhao, X. Luo, and X. Dou, "Using deep learning for classification of lung nodules on computed tomography images," *Journal of Healthcare Engineering*, vol. 2017, Article ID 8314740, 7 pages, Aug. 2017, doi: 10.1155/2017/8314740.

[16]
A. Golkarieh, F. Bayrami, and R. A. Lashaki, "Enhancing Lung Cancer Detection: Segmentation and Classification of CT Images Using U-Net with Pretrained Backbones and BIR Model," *Research Square*, preprint, Oct. 2024. [Online]. Available: https://doi.org/10.21203/rs.3.rs-5232211/v1

[17]
P. Lobo and S. Guruprasad, “Classification and Segmentation Techniques for Detection of Lung Cancer from CT Images,” *2018 International Conference on Inventive Research in Computing Applications (ICIRCA)*, pp. 1014–1019, Jul. 2018, doi: 10.1109/icirca.2018.8597273.

[18]
C. Liu *et al.*, “Automatic detection of pulmonary nodules on CT images with YOLOv3: development and evaluation using simulated and patient data,” *Quantitative Imaging in Medicine and Surgery*, vol. 10, no. 10, pp. 1917–1927, 2020.

[19]

S. Li *et al.*, "Predicting Lung Nodule Malignancies by Combining Deep Convolutional Neural Network and Handcrafted Features," *Physics in Medicine & Biology*, vol. 64, no. 17, p. 175012, 2019.

[20]
N. Khosravan and U. Bagci, "S4ND: Single-Shot Single-Scale Lung Nodule Detection," *Proc. MICCAI 2018*, pp. 794–802, 2018.

[21]
A. J. Alberg and J. M. Samet, "Epidemiology of lung Cancer," in *Elsevier eBooks*, 2010, pp. 1098–1115. doi: 10.1016/b978-1-4160-4710-0.00046-8.

[22]
American Lung Association, "Lung Cancer Screening: Early Detection Decreases Mortality," *American Lung Association*, 2022. [Online]. Available: https://www.lung.org/blog/lung-cancer-screening-saves-lives

[23]
J. C. Bezdek, *Pattern Recognition with Fuzzy Objective Function Algorithms*. 1981. doi: 10.1007/978-1-4757-0450-1.

[24]
P. Afshar, A. Ahmadi, and M. H. F. Zarandi, "Lung tumor area recognition in CT images based on Gustafson-Kessel clustering," *2022 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE)*, Jul. 2016, doi: 10.1109/fuzz-ieee.2016.7737980.

[25]
B. U. Dhaware and A. C. Pise, "Lung cancer detection using Bayasein classifier and FCM segmentation," Sep. 2016, doi: https://doi.org/10.1109/icacdot.2016.7877572.

[26]

aadityagupta007, “lung cancer segmentation,” *Kaggle.com*, Feb. 23, 2025. https://www.kaggle.com/code/aadityagupta007/lung-cancer-segmentation (accessed Sep. 07, 2025).

[27]
vedantkadam, “GitHub - vedantkadam/Lung_Cancer_Streamlit: Lung Cancer Detection using Machine Learning,” *GitHub*, 2023. https://github.com/vedantkadam/Lung_Cancer_Streamlit (accessed Sep. 07, 2025).

[28]
mrshamshir, “GitHub - mrshamshir/Lung-Tumor-Segmentation: Automatically lung tumor segmentation in CT scan images.,” *GitHub*, 2023. https://github.com/mrshamshir/Lung-Tumor-Segmentation.git (accessed Sep. 07, 2025).

[29]
Y. Li, X. Yang, X. Chen, E. Fu, G. Ren, and Y. Mu, “Automatic Detection of Lung Nodules in Computed Tomography (CT) Images: A Systematic Review”, *SJISR*, vol. 7, no. 1, pp. 72–89, Feb. 2025, doi: 10.54691/a3577023.

[30]
Iamssuraj, “GitHub - iamssuraj/Interactive-Lung-Cancer-Detection-System: An ML based webapp, using Python modules. Here’s the live working app: https://lung-cancer.streamlit.app/,” *GitHub*. https://github.com/iamssuraj/Interactive-Lung-Cancer-Detection-System

[31]
codeWudaya, “GitHub - codeWudaya/Lung-Infection-Detection-app: It’s a Deep Learning based Lung Infection detector that is deployed in Web in streamlit. Technology used is Convolutional Nural Network ,tensorflow / keras ,python,” *GitHub*. https://github.com/codeWudaya/Lung-Infection-Detection-app

[32]
“DeepHealth Lung - DeepHealth,” *DeepHealth*, Jul. 24, 2025. https://deephealth.com/population-health/deephealth-lung/

APPENDIX

POSTER