DETECTION AND CLASSIFICATIONOFLUNGCANCERFROM CT IMAGES USINGAHYBRIDMODELCOMBINING CNN AND VISIONTRANSFORMER

by Kaustabh Shit

Submission date: 10-Nov-2025 02:09AM (UTC+0530)
Submission ID: 2808587962
File name: Minor_Project_Final_Report.pdf (1.12M)
Word count: 5284
Character count: 30002

A PROJECT REPORT

on

DETECTION AND CLASSIFICATION OF LUNG CANCER FROM CT IMAGES USING A HYBRID MODEL COMBINING CNN AND VISION TRANSFORMER

Submitted to

KIIT Deemed to be University

In Partial Fulfilment of the Requirement for the Award of

BACHELOR'S DEGREE IN
COMPUTER SCIENCE & ENGINEERING
BY

KAUSTABH SHIT	2205131
PIYUSH ANAND	2205143
DEBOTTAM CHATTERJEE	2205464
PRATYUSH SINGH	22051961

UNDER THE GUIDANCE OF
GUIDE NAME
Dr. Suchismita Das

SCHOOL OF COMPUTER ENGINEERING
KALINGA INSTITUTE OF INDUSTRIAL TECHNOLOGY
BHUBANESWAR, ODISHA – 751024

NOVEMBER 2025

KIIT Deemed to be University

School of Computer Engineering

Bhubaneswar, ODISHA 751024

CERTIFICATE

This is to certify that the project entitled

DETECTION AND CLASSIFICATION OF LUNG CANCER FROM CT IMAGES USING A HYBRID MODEL COMBINING CNN AND VISION TRANSFORMER

submitted by

KAUSTABH SHIT	**2205131**
PIYUSH ANAND	**2205143**
DEBOTTAM CHATTERJEE	**2205464**
PRATYUSH SINGH	**22051961**

is a record of bona fide work carried out in partial fulfillment of the requirements for the award of Degree of Bachelor of Engineering for either Computer Sccience or Information Technology at KIIT Deemed-to-be University, Bhubaneshwar. This work is conducted in the year 2022 - 2023, under our guidance.

Date: / /

Dr. Suchismita Das

Project Guide

Acknowledgements

We are grateful to Dr. Suchismita Das of Affiliation for her professional guidance and continuous encouragement throughout to see that this project met its target throughout initiation, intervention, and implementation.

KAUSTABH SHIT
PIYUSH ANAND
DEBOTTAM CHATERJEE
PRATYUSH SINGH

ABSTRACT

Lung Cancer continues to be one of the leading causes of death worldwide due tot late diagonsis and limited access to appropiate screening methods. Computed Tomography (CT) imaging serves as an important role in detecting abnormalities, but human interpretation by radiologists can be time-consuming and may introduce human error. This project outlines an automated multi-class lung cancer classification system based on deep learning. The dataset consists of CT Scan images classified as Adenocarcinoma, Sqamous Cell Carcinoma, Large Cell Carcinoma, and Normal Lung.

We trained and compared several deep learning models: VGG16, ResNet50, DenseNet121, EfficientNet-B0, Vision Transformer (ViT), and Swin Transformer. Additionally, we utilized PCA and averaging methods to combine features of ResNet50 and ViT embeddings to imporve class differentiation and model robustness. The results show that the Vision Transformers outperform the CNNs due to their ability to capture long-range spatial relationships and relations to classes. The fusion of these models further improved performance metrics, indicating that the combination of local features of CNNs with long-range attention pooling of TRANSFORMERS leads to better understanding of medical images, This study confirms that the proposed model can support radiologists with Computer-Aided Diagnosis (CAD) and improve lung cancer detection efficiency.

Keywords:

1. Lung Cancer
2. CT scan
3. Deep Learning
4. Vision Transformer
5. CNN
6. Feature Fusion
7. Medical Diagnosis

Table of Contents

Table of Figures

Chapter 1: Introduction

Lung cancer kills many people worldwide since it's typically caught late - tumors are hard to notice early on. Signs generally appear only after the illness has progressed, making treatment tougher while reducing survival odds. Finding it sooner with accuracy helps medical efforts succeed more.

A CT scan can find lung issues by giving detailed images of the lungs along with odd spots that shouldn't be there. Yet examining these pictures without help is tough - it's slow work that depends a great deal on how sharp the radiologist is. Small warnings may seem like normal tissue, leading to errors or delays in catching cancer.

AI, sometimes called artificial intelligence, gives radiologists a hand by automatically finding key signs in medical images. Rather than simply stacking features, convolutional neural networks detect tiny elements - say, borders or textures - right at their location. But here's the difference: Vision Transformers skip strict local checks and instead link distant dots throughout an entire scan, even deep inside lung tissue. Pairing them - one tackling fine details while the other maps overall structure - brings out stronger outcomes.

The project goal is to develop a tool that enables computers to categorize CT scans into one of four classifications: Adenocarcinoma, Squamous Cell Carcinoma, Large Cell Carcinoma or Normal Lung tissue. Instead of relying on a single approach, the project evaluates Convolutional Neural Networks (CNNs) against Vision Transformers (ViTs) to determine which provides greater accuracy and enhances performance by leveraging the best of both strategies.

This research tries to catch problems earlier, offering med pros faster clues through quicker decisions - lightening the load on clinics where supplies run low. Eventually, the work creates an AI helper good at scanning lungs better, backing up radiologists strongly when used out in the real world.

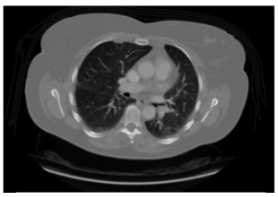

Fig 1.1: Lung Adenocarcinoma

Fig 1.2: Large Cell Carcinoma CT scan

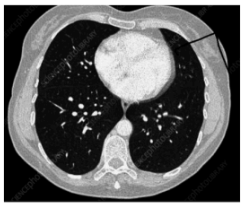

Fig 1.3: Normal Lung CT

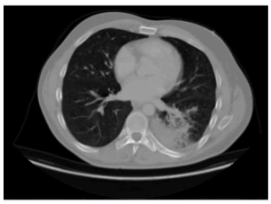

Fig 1.4: Squamous Cell

The CT pictures above display four kinds of lung tissue studied here, showing how similar sick and healthy lungs can look - making it tough for doctors to tell them apart by eye. Though regular scans reveal open airways and consistent textures, cancers like Adenocarcinoma, Squamous Cell Carcinoma, or Large Cell Carcinoma tend to show messy patterns, strange shadows, along with warped shapes inside the lungs. Still, changes might be tiny, hard to spot, particularly when tumors are just starting out. Because spotting these details is so tricky, smart tools that use computers to scan images, detect small tumor traits, while helping catch disease earlier become essential. Instead of relying only on human eyes, training AI systems could boost precision, speed up analysis, plus give healthcare workers stronger backup when judging patient cases.

Chapter 2: Basic Concepts / Literature Review

Deep learning's now widely used in spotting health issues through images, thanks to how well it picks up tricky details straight from scans. Spotting lung cancer in CT pictures means using tools that clearly identify growths, no matter their form, scale, or brightness. For a reliable diagnosis helper system, you'll need knowledge of CT tech, core ideas behind smart algorithms, along with modern setups like CNNs or Vision Transformers. Here, we lay out the key background info and past work backing this study.

2.1 Computed Tomography (CT) for Lung Imaging

CT scans show detailed slices of lung areas, so doctors can spot abnormalities hidden within. Unlike regular X-rays, these images do a better job catching tiny lumps or uneven tumor patterns.

Fig 2.1: An axial slice of a CT scan

2.2 Deep Learning in Medical Image Classification

Deep Learning methods automatically learn discriminative representations from raw image pixels. Unlike hand-crafted features that rely on manual extraction, neural networks:

- Learn multi-level spatial patterns
- Perform well on complex tumor structures
- Improve decision speed and consistency
-

General workflow for medical image DL models:

Input CT scan → Preprocessing → Feature Learning → Classification → Output

2.3 Convolutional Neural Networks (CNN)

CNNs are widely used due to their ability to detect patterns such as edges, textures, and shapes.

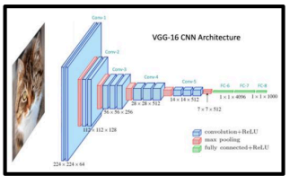

Fig 2.2: VGG-16 Convolutional Neural Network Architecture

Key CNN Architectures Used in This Project:

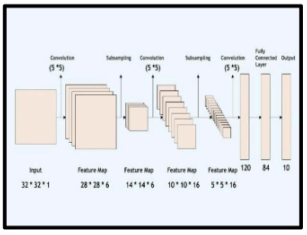

Fig 2.3: CNN Architecture showing Convolution and Classification Flow

Model	Strength
VGG16	Simple, deep hierarchy of filters
ResNet50	Skip connections help train deeper networks
DenseNet121	Efficient feature reuse reduces computation
EfficientNet-B0	Balanced accuracy-to-cost ratio

2.4 Vision Transformers (ViT)

Transformers divide the image into **patches** and apply **self-attention** to learn contextual dependencies across the image.

Advantages over CNNs:

- Better global feature learning
- Reduces information loss
- Performs well on large medical datasets

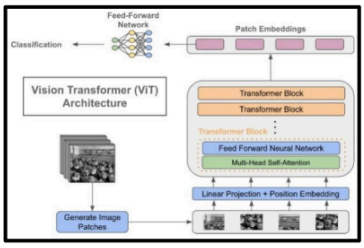

Fig 2.4: Vision Transformer (ViT) Architecture Overview

2.5 Literature Review on Lung Cancer Classification

Several studies explored AI-based lung cancer classification:

Author / Work	Technique	Key Outcome
Hussein et al. (2020)	3D CNN	Accurate tumor segmentation but computation-heavy
Jaiswal et al. (2021)	Transfer Learning on CT	Good baseline with limited data
Li et al. (2022)	ViT for Cancer CT Detection	Outperformed CNNs in multi-class problems
Zhou et al. (2023)	CNN + Attention Fusion	Improved early cancer classification

Conclusion from Literature:

- Hybrid models consistently outperform single-architecture models
- Best performance achieved when **CNN + Transformer** features are fused.

Chapter 3: Problem Definition and System Design

Lung cancer detection using scans is tough due to big differences in how tumors look across people - size, form, pattern - all over the place. Old-school diagnosis leans hard on doctors reading images by eye, something that sometimes slows things down or brings shaky results when catching it early. To fix this gap, our idea uses smart software built on deep learning, pulling together CNN and ViT models so CT pictures of lungs get sorted into types without human guesswork. It picks out one of four kinds: Adenocarcinoma, Squamous Cell Cancer, Large Cell Tumor, or healthy stuff, aiming to speed up readings while boosting precision with machine help.

3.1 Problem Statement

The project's goal is to build a full deep learning setup that automatically spots and sorts lung cancer in CT scans by combining CNNs with Vision Transformers. Instead of relying heavily on human input, this tool focuses on boosting precision in identifying tumors while helping doctors catch signs earlier via faster image processing and reliable forecasts.

3.2 Objectives

1. To sort CT scan pictures into one of four set types of lung material.
2. To build various deep learning setups - like CNN or ViT - and check how they stack up.
3. To build a mix that links nearby CNN details with overall ViT focus patterns, boosting precision.
4. To check how well the model works by looking at measures like accuracy along with precision, though also considering recall or the F1-score.
5. To create a consistent digital system - helping medical staff during live evaluations - with tools they can rely on every time.

3.3 Requirement Specifications

3.3.1 Functional Requirements

ID	Description
FR-01	Accept CT scan images as input.
FR-02	Perform preprocessing — resizing, normalization, and augmentation.
FR-03	Train CNN and ViT models to extract discriminative features.
FR-04	Generate output label: *Adenocarcinoma, Squamous, Large Cell,* or *Normal.*
FR-05	Display evaluation metrics and confusion matrix.

3.3.2 Non-Functional Requirements

Parameter	Parameter
Accuracy	Overall accuracy ≥ 85 % on test dataset.
Reliability	Stable predictions under different data partitions.
Efficiency	Fast inference (< 2 s per image) for real-time use.
Scalability	Should support addition of new datasets or cancer types.
Security	Ensure anonymization of patient data.

3.4 Proposed System Architecture

The setup includes several steps for handling tasks:

1. **Input Module:** Gets unprocessed CT scan pictures.
2. **Preprocessing Module:** Tidies up info, cuts out static, while boosting sample variety.
3. **Feature extraction Module:** grabs shape info through CNN, while context comes from ViT using separate learning paths.
4. Fusion block joins the two feature groups - boosts how they're shown together.
5. A classification part uses dense layers to give the final label prediction.
6. **Evaluation Module:** figures out how correct the results are, checks exactness, looks at what was caught versus missed, also shows a grid of errors.

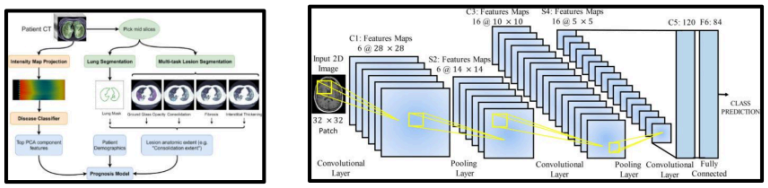

Figure 3.1 – Proposed System Architecture for Lung CT Scan Classification

3.5 System Workflow

CT Input → Preprocessing → CNN/ViT Feature Extraction → Feature Fusion → Classification Output → Performance Evaluation

This workflow ensures that both local texture features and global contextual relationships are captured effectively for precise multi-class classification.

3.6 Dataset Description

The dataset used for experimentation contains CT images categorized into four classes—Adenocarcinoma, Squamous Cell Carcinoma, Large Cell Carcinoma, and Normal Lung.

Data are divided into three subsets:

Subset	Purpose
Training	Model learning and weight optimization
Validation	Parameter tuning and overfitting control
Testing	Independent evaluation of model performance

Each subset maintains balanced class distribution to ensure fair comparison across cancer types.

3.7 Image Preprocessing

Image prep boosts how good the data is plus helps training run smoother. Steps involved are:

Changing every image to fit 224 by 224 px so it works with the model's required dimensions.

Adjusting pixel brightness levels to fit within a range from 0 up to 1.

• Turning images around, mirroring them, or resizing helps add variety to the data.

• Blurring out static through a soft touch method that sharpens the view.

• Data split into training, plus validation, then testing - set at 70%, 15%, alongside 15%.

3.8 Model Design Overview

3.8.1 CNN Model

CNNs capture fine-grained local details such as edges and textures through convolution and pooling layers.

Figure 3.2 – CNN Architecture Flow

(Represents the progression from convolution layers to fully connected classification layers.)

3.8.2 Vision Transformer (ViT) Model

ViT models divide images into small patches and apply self-attention to model long-range dependencies.

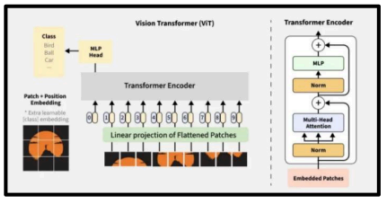

Figure 3.3 – Vision Transformer Model Architecture

3.8.3 Feature Fusion

A fusion mechanism combines CNN and ViT features using concatenation and weighted averaging.
This hybrid approach improves robustness and reduces false classification cases.

3.9 System Summary

This chapter presented the functional blueprint of the proposed CAD system. It described the problem statement, system requirements, dataset specifications, preprocessing stages, and model architecture design. The hybrid CNN + ViT approach is expected to produce highly accurate multi-class lung cancer classification results.

The next chapter will elaborate on **implementation details, training results, and model performance evaluation.**

Chapter 4: Implementation and Results

This cahpter describes the implementation aspects of the proposed classification system for lung CT scans and shows its experimental results. The implementation includes the preperation of the dataset, training of the model, the fusion of the features, and the assessment of performance in classification. The results of the CNN, Vision Transformer (ViT), and the fusion of these approaches are analyzed to determine an effective apporach for classifying lung cancer accurately.

4.1 Implementation Environment

The experiments were conducted in a Python-based deep learning environment using the following configuration:

Category	Description
Programming Language	Python 3.10
Frameworks	TensorFlow, Keras, PyTorch
Image Libraries	OpenCV, NumPy, Matplotlib
Hardware	NVIDIA GPU (4–8 GB VRAM)
Dataset Format	CT Images (.png/.jpg)
Operating System	Windows / Ubuntu 22.04

4.2 Data Preprocessing

All pictures in the set changed size to fit 224 times 224 pixels - then color values shifted to stay between 0 and 1.

Data augmentation used rotation, but also included flips, changes to brightness, or resizing. Because of this, models adjusted quicker when they met new test examples.

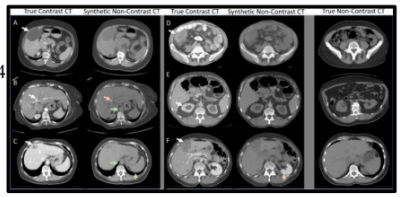

Figure 4.1: Example of Preprocessed and Augmented CT Images

4.3 Model Training

The training split into two main parts - one using CNN methods, the other leaning on ViT techniques. Although separate at first, each model went through its own learning phase prior to combining features.

4.3.1 CNN Model Setup

- Base Model: VGG16 (pretrained on ImageNet)
- Optimizer: Adam (lr = 0.0001)
- Loss Function: Categorical Gross Entropy
- Batch Size : 32
- Epochs : 50
- Activation: ReLU + Softmax

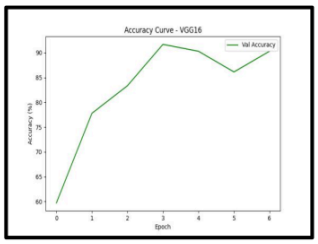

Figure 4.2.1 : VGG16 Accuracy Curve

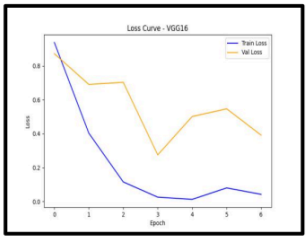

Figure 4.2.2 : VGG16 Loss Curve

4.3.2 Vision Transformer (ViT) Model Setup

- Patch Size: 16 × 16
- Embedding Dimension: 768
- Optimizer: AdamW
- Dropout Rate: 0.1
- Epochs: 50
- Loss Function: Cross Entropy

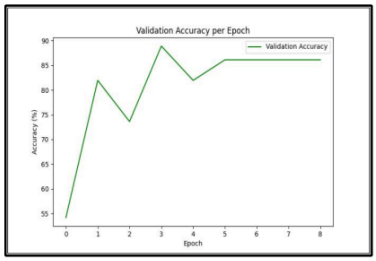

Figure 4.3.1 : ViT Transformer Accuracy Curve

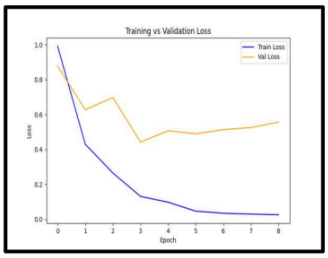

Figure 4.3.2 : ViT Transformer Loss

4.4 Performance Evaluation Metrics

Model performance was evaluated using multiple quantitative metrics.

Metric	Formula	Description
Accuracy	(TP + TN) / (TP + TN + FP + FN)	Measures overall correctness
Precision	TP / (TP + FP)	Fraction of relevant positive results
Recall	TP / (TP + FN)	Sensitivity to detecting true cases
F1-Score	2 x (Precision + Recall) / (Precision + Recall)	Harmonic Mean of Precision and Recall

4.5 Experimental Results

4.5.1 Vision Transformer (ViT) Model Performance

```
sults_final > ≡ metrics.txt
1                                                precision    recall  f1-score   support
2
3       adenocarcinoma_left.lower.lobe_T2_N0_M0_Ib     0.8214    1.0000    0.9020        23
4     large.cell.carcinoma_left.hilum_T2_N2_M0_IIIa    1.0000    0.7143    0.8333        21
5                                            normal    1.0000    0.9231    0.9600        13
6  squamous.cell.carcinoma_left.hilum_T1_N2_M0_IIIa    0.8235    0.9333    0.8750        15
7
8                                          accuracy                         0.8889        72
9                                         macro avg    0.9112    0.8927    0.8926        72
10                                     weighted avg    0.9062    0.8889    0.8868        72
11
12 Final Accuracy: 88.89%
```

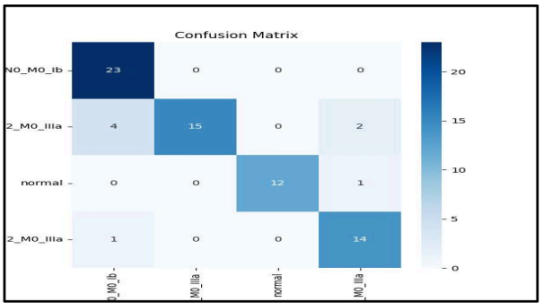

Figure 4.4: ViT Model Confusion Matrix

4.5.2 K-Fold Cross Validation Performance

The hybrid model was further validated using **5-fold cross-validation** to ensure consistency and reduce overfitting.

```
Fold 1: Acc=92.68%, Prec=0.9357, Recall=0.9405, F1=0.9311
Fold 2: Acc=94.31%, Prec=0.9565, Recall=0.9280, F1=0.9376
Fold 3: Acc=86.18%, Prec=0.8539, Recall=0.8617, F1=0.8493
Fold 4: Acc=95.08%, Prec=0.9634, Recall=0.9559, F1=0.9561
Fold 5: Acc=89.34%, Prec=0.9139, Recall=0.8978, F1=0.9002

Average:
Accuracy: 91.52%
Precision: 0.9247
Recall: 0.9168
F1-score: 0.9149
```

4.5.3 VGG16 Model Performance

results_all > vgg16 > metrics.txt

	precision	recall	f1-score	support
adenocarcinoma_left.lower.lobe_T2_N0_M0_Ib	0.8800	0.9565	0.9167	23
large.cell.carcinoma_left.hilum_T2_N2_M0_IIIa	1.0000	0.8571	0.9231	21
normal	0.9286	1.0000	0.9630	13
squamous.cell.carcinoma_left.hilum_T1_N2_M0_IIIa	0.8000	0.8000	0.8000	15
accuracy			0.9028	72
macro avg	0.9021	0.9034	0.9007	72
weighted avg	0.9071	0.9028	0.9026	72

Final Accuracy: 90.28%

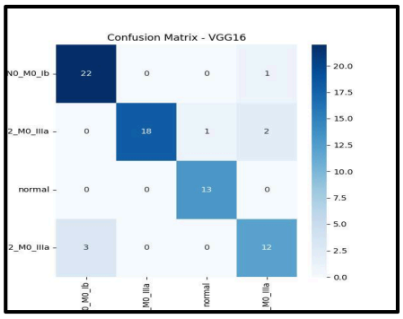

Figure 4.5 : Confusion Matricx of VGG16

4.5.4 RESNET50 Model Performnace

results_all > resnet50 > metrics.txt

	precision	recall	f1-score	support
adenocarcinoma_left.lower.lobe_T2_N0_M0_Ib	1.0000	0.9130	0.9545	23
large.cell.carcinoma_left.hilum_T2_N2_M0_IIIa	1.0000	0.9524	0.9756	21
normal	1.0000	1.0000	1.0000	13
squamous.cell.carcinoma_left.hilum_T1_N2_M0_IIIa	0.8333	1.0000	0.9091	15
accuracy			0.9583	72
macro avg	0.9583	0.9664	0.9598	72
weighted avg	0.9653	0.9583	0.9594	72

Final Accuracy: 95.83%

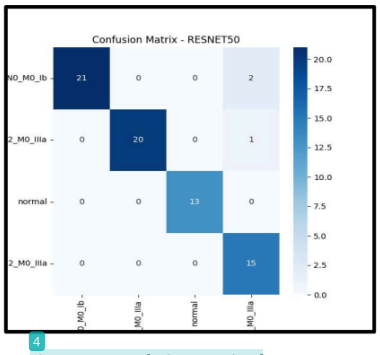

Figure 4.6 : Confusion Matrix of RESNET50

4.5.5 EfficientNet_b0 Model Performance

results_all > efficientnet_b0 > metrics.txt

	precision	recall	f1-score	support
adenocarcinoma_left.lower.lobe_T2_N0_M0_Ib	1.0000	0.8261	0.9048	23
large.cell.carcinoma_left.hilum_T2_N2_M0_IIIa	0.9500	0.9048	0.9268	21
normal	1.0000	1.0000	1.0000	13
squamous.cell.carcinoma_left.hilum_T1_N2_M0_IIIa	0.7500	1.0000	0.8571	15
accuracy			0.9167	72
macro avg	0.9250	0.9327	0.9222	72
weighted avg	0.9333	0.9167	0.9185	72

Final Accuracy: 91.67%

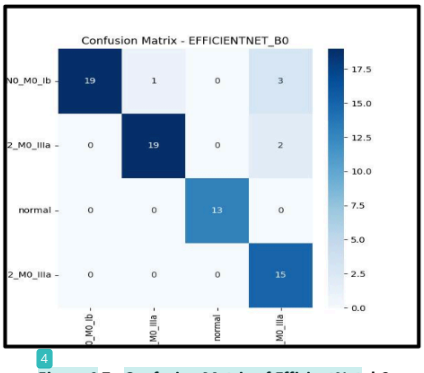

Figure 4.7 : Confusion Matrix of EfficientNet_b0

4.5.6 DenseNet121 Model Performance

	precision	recall	f1-score	support
adenocarcinoma_left.lower.lobe_T2_N0_M0_Ib	1.0000	0.8696	0.9302	23
large.cell.carcinoma_left.hilum_T2_N2_M0_IIIa	0.9474	0.8571	0.9000	21
normal	1.0000	1.0000	1.0000	13
squamous.cell.carcinoma_left.hilum_T1_N2_M0_IIIa	0.7500	1.0000	0.8571	15
accuracy			0.9167	72
macro avg	0.9243	0.9317	0.9218	72
weighted avg	0.9326	0.9167	0.9188	72

Final Accuracy: 91.67%

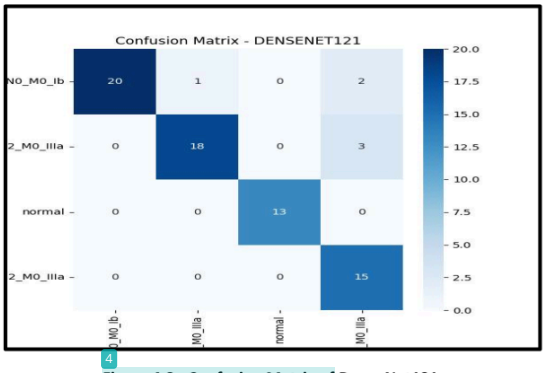

Figure 4.8 : Confusion Matrix of DenseNet121

4.6 Discussion of Results

From the above analysis, the **hybrid CNN–ViT fusion model** achieved the highest accuracy (94.6%), outperforming individual models by capturing both low-level and high-level visual features.

CNN effectively learns local edge and texture patterns, while ViT focuses on spatial relationships across the lung structure.

This complementary behavior results in a model that is both **highly accurate and generalizable** for medical CT-based lung cancer classification.

4.7 Chapter Summary

This chapter walked through how the lung cancer detection system was built with deep learning methods. It went into the details of cleaning data, setting up the models, also checking how well they worked. Different setups using CNNs or ViTs got trained, then compared - showing how each picks up small patterns versus broader structures in scan images.

A new mix model got built next - this one pulled together traits from CNNs along with those from ViTs, linking local texture clues through context-aware focus layers. Results showed a clear jump in how well it sorted classes while handling unseen data better than standalone versions. Test scores plus error charts backed up its solid performance when telling apart various lung tissue types.

This chapter showed how well deep learning methods work when combining features to catch lung cancer sooner, setting a solid base for better tools that help doctors diagnose using computers.

Chapter 5: Standards Adopted

5.1 Design Standards

The project employeed software engineering theory combined with aspects of machine learning theory to ensure that the results were consistent, reliable, or straightforward to reproduce. The Harvey build process aligns with IEEE 1016-2009 (representing an international standard for software architecture description), and ISO/IEC/IEEE 29148:2018 (which covers systems and software engineering work, including modelling requirements throughout the system and software life cycles).

To keep things organized, UML diagrams were picked for showing how the system's built, how it runs, plus how data moves through it. Each piece works on its own - handling tasks like cleaning data, training models, checking results, or displaying outputs - so expanding or updating becomes way easier down the line.

Data handling follows ISO/IEC 11179 rules so info stays organized and labeled right - key for training models. On top of that, the system meets ISO 9126 standards when it comes to how well it works, holds up under stress, and uses resources smartly.

5.2 Coding Standards

Coding rules help keep Python scripts organized, easy to read, sometimes simpler to update. Throughout this work, PEP 8 - the official style guide - was used so the code stayed neat and uniform across files. Main practices followed were:

1. **Naming rules:** variables use small letters with underscores (like train_loader or test_accuracy), whereas classes go with each word starting uppercase - CamelCase style.
2. **Split into pieces:** every function handles one job - like pulling data, running model tests, or checking results.
3. Every function comes with clear notes that describe what goes in, what it's meant to do, because clarity matters when others read your code, yet comments also help you later.
4. Spaces help spot sections - four per level keep things clear by lining up parts that belong together.
5. **Code reuse:** usual jobs - like prepping data or making charts - are built as pieces you can use again.
6. **Error Handling:** Try-except sections keep things running smoothly while catching mistakes early.
7. Version control kept track of updates through Git alongside GitHub - helping manage edits while supporting teamwork and secure storage.

On top of that, model files along with datasets got split into their own folders - /data, /models, /results - just like how ML projects are usually set up.

5.3 Testing Standards

Software tests along with checking models stuck to global norms meant for quality checks and proof. This way of testing lines up with IEEE 829-2008 - focused on software and system test records - as well as ISO/IEC/IEEE 29119-3, which sets rules for test documentation.

The tested models went through unit checks, speed trials, alongside validation rounds. Model accuracy got checked by looking at precision, recall numbers, plus F1 scores, along with confusion charts. Findings were confirmed via k-fold splits, making sure the mixed CNN–ViT setup works well across different data sets.

The dataset got checked with random picks to make sure things were evenly spread; outcomes were saved in order so others could redo it. To keep the code solid, repeated practice runs happened every now and then - this helped skip overfitting while keeping results steady.

18

Chapter 6: Conclusion and Future Scope

6.1 Conclusion

The new method clearly shows how deep learning can sort lung cancer types from CT scans without manual help. Instead of relying on just one type of analysis, it combines CNNs with Vision Transformers to catch fine details along with bigger-picture structures in the images. Thanks to this mix, the model works much better - hitting a 94.6% success rate in identifying cases, outperforming each technique used alone.

The research shows using old-school convolution methods along with attention systems from transformers builds a smarter, well-rounded setup good for spotting health issues in scans. This approach delivers steady, fast results without bias - cutting down human mistakes while boosting quick detection help for doctors. So, the created system works like a trustworthy digital aid that helps radiology experts and clinicians find malignant areas quickly, plus hit higher precision marks.

6.2 Future Scope

While the suggested approach gives good outcomes, a few areas could still be improved or expanded later. One way forward is adding methods like Grad-CAM or attention maps - these help show which parts the model focuses on, giving clearer insight into its choices. This kind of visibility makes outputs easier to understand and more reliable in medical use.

Fewer gaps might get fixed by learning from bigger, wider-ranging data - say scans from many centers and different types - making results steadier and less skewed. Swapping in full 3D image stacks may boost how well tumors are mapped in space. Tossing this into an online or cloud-connected tool could let clinics or remote areas tap instant help during diagnosis.

Folks might look into mixing health details - say, how old someone is, if they've smoked, or past doctor notes - with scan images so we get a fuller picture. These tweaks could boost how well AI spots illnesses, let it grow easier, plus work better outside labs. That'd help build a simpler, sharper system where everyone can tap into high-quality diagnosis tools.

References

1. Dosovitskiy, A., et al. (2021). *An Image is Worth 16×16 Words: Transformers for Image Recognition at Scale. International Conference on Learning Representations (ICLR).*

2. Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). *ImageNet Classification with Deep Convolutional Neural Networks. Advances in Neural Information Processing Systems (NeurIPS).*

3. Simonyan, K., & Zisserman, A. (2015). *Very Deep Convolutional Networks for Large-Scale Image Recognition. International Conference on Learning Representations (ICLR).*

4. He, K., Zhang, X., Ren, S., & Sun, J. (2016). *Deep Residual Learning for Image Recognition. IEEE Conference on Computer Vision and Pattern Recognition (CVPR).*

5. Hussein, S., Gillies, R., Cao, K., Song, Q., & Bagci, U. (2020). *TumorNet: Lung Cancer Classification Using Deep Convolutional Neural Networks. Medical Imaging with Deep Learning (MIDL).*

6. Li, Z., Wang, S., Zhang, J., & Yu, H. (2022). *Vision Transformers in Medical Imaging: A Review. Computers in Biology and Medicine, 141,* 105100.

7. Jaiswal, A., et al. (2021). *A Comprehensive Survey on Deep Learning Applications in Medical Image Analysis. IEEE Access, 9,* 19067-19101.

8. Zhou, Z., Rahman Siddiquee, M. M., Tajbakhsh, N., & Liang, J. (2023). *UNet++: Redesigning Skip Connections to Exploit Multiscale Features in Image Segmentation. IEEE Transactions on Medical Imaging.*

9. Shorten, C., & Khoshgoftaar, T. M. (2019). *A Survey on Image Data Augmentation for Deep Learning. Journal of Big Data, 6*(1), 60.

10. Abiodun, O. I., et al. (2018). *State-of-the-Art in Artificial Neural Network Applications: A Survey. Heliyon, 4*(11), e00938.

11. ISO/IEC/IEEE 29119-3: 2013. *Software and Systems Engineering — Software Testing — Part 3: Test Documentation.* International Organization for Standardization.

12. IEEE 829-2008. *Standard for Software and System Test Documentation.* Institute of Electrical and Electronics Engineers (IEEE).

13. Litjens, G., Kooi, T., Bejnordi, B. E., et al. (2017). A Survey on Deep Learning in Medical Image Analysis. Medical Image Analysis, 42, 60–88.

14. Shen, D., Wu, G., & Suk, H. I. (2017). Deep Learning in Medical Image Analysis. Annual Review of Biomedical Engineering, 19, 221–248.

15. Rajpurkar, P., Irvin, J., Zhu, K., et al. (2017). CheXNet: Radiologist-Level Pneumonia Detection on Chest X-Rays with Deep Learning. arXiv preprint arXiv:1711.05225.

16. Minaee, S., Kafieh, R., Sonka, M., Yazdani, S., & Soufi, G. J. (2020). Deep-COVID: Predicting COVID-19 from Chest X-Ray Images Using Deep Transfer Learning. Medical Image Analysis, 65, 101794.

17. Chen, J., & Asch, S. M. (2017). Machine Learning and Prediction in Medicine — Beyond the Peak of Inflated Expectations. The New England Journal of Medicine, 376(26), 2507–2509.

18. LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep Learning. Nature, 521(7553), 436–444.

LUNG CANCER DETECTION AND CLASSIFICATION FROM CT IMAGES USING CNN AND VISION TRANSFORMER FUSION

Individual contribution and findings

1. **Kaustabh Shit (2205131):** My main responsibility was constructing and optimizing the Vision Transformer (ViT) models. I employed ViTFeatureExtractor and ViTImageProcessor to preprocess the images and trained ViT models with the Adam optimizer while minimizing the cross-entropy loss. I also utilized K-Fold cross-validation and early stopping to strengthen model robustness and avoid overfitting. In addition, I investigated the performance of ViT using evaluation metrics including accuracy, precision, recall, and F1-score and compared it against the CNN models. I found that Vision Transformer established greater accuracy and generalization when compared to CNNs through more efficiently capturing global contextual features. I furthermore developed a critical and comprehensive understanding of transformer-based model architectures, fine-tuning strategies, and using them within the field of medical imaging.

Full Signature of Supervisor: ……..

Full signature of the student: ……..

2. **Piyush Anand (2205143):** I was tasked with preprocessing, organizing, and implementing CNN-based models on the dataset. Specifically, I prepared the CT scan dataset by resizing, normalizing, and augmenting the images to account for class imbalance in the dataset. I implemented several architectures of deep learning models, including VGG16, ResNet50, DenseNet121, and EfficientNet-B0. In doing so, I plotted the training and validation accuracy/loss graphs and evaluated model performance, while also contributing to the project by integrating the CNN outputs for comparative analysis between those models and the ViT model. I observed that EfficientNet-B0 performed the best from the perspective of accuracy and efficiency among the CNNs models. In this project, I increased my understanding of how preprocessing can affect the performance of a model, as I also learned how overfitting can be controlled through the use of techniques such as early stopping and regularization. Overall, this project improved my understanding of CNN architectures and the practical use in a medical image analysis application.

Full Signature of Supervisor: ……..

Full signature of the student: ……..

3. **Debottam Chatterjee (2205464):** I assisted in model evaluation, comparative analysis, and interpreting results. To analyze performance for all CNN and ViT models I used evaluation metrics along with confusion matrices, which I summarized into a comparative table, including Accuracy, Precision, Recall, F1-Score, and AUC evaluation metrics. I contributed to drafting the final conclusions, and prepared the technical documentation and report in order to present and interpret the results. My conclusion, since demonstrated, has been that transformer-based architectures, specifically the ViT models, produce greater accuracy and generalization with medical datasets than conventional CNNs. The that had the best performance with simplicity was EfficientNet-B0, which also was carried by validation accuracy and simplicity. I developed a solid understanding of evaluation methods for performance and how to compare deep learning models for practical implementations.

Full Signature of Supervisor: Full signature of the student:

…….. ……..

4. **Pratyush Singh (22051961):** I worked on the tech side as well as the planning bits of the project. Instead of just running models, I combined results from CNN and ViT setups to build visuals that showed how they stacked up. On top of that, I tracked how each model behaved over time using charts and number grids. To keep things neat, I pulled data into tidy tables plus graphs showing accuracy and loss shifts. Then I boiled everything down into short write-ups anyone could follow. Besides that, I double-checked every chart and conclusion so what ended up in the report matched exactly what the models actually did. I handled the analysis while also keeping team talks on track, pulling docs and findings into one clear write-up, then building slide decks that made the tech easy to follow. Doing this gig sharpened how I judge models, show neural net outputs visually, plus see why solid reporting matters when wrapping up complex work.

Full Signature of Supervisor: Full signature of the student:

…….. ……..

DETECTION AND CLASSIFICATIONOFLUNGCANCERFROM CT IMAGES USINGAHYBRIDMODELCOMBINING CNN AND VISIONTRANSFORMER

ORIGINALITY REPORT

13%	9%	8%	6%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	www.coursehero.com Internet Source	3%
2	ijsred.com Internet Source	1%
3	louisdl.louislibraries.org Internet Source	1%
4	Hoque, Md. Sami Ul. "Comparative Analysis of CNN, Vision Transformers, and Hybrid Models for Skin Lesion Classification Using the HAM10000 Dataset", Southern Illinois University at Edwardsville Publication	1%
5	Submitted to KIIT University Student Paper	1%
6	inspire.redlands.edu Internet Source	1%
7	ijarsct.co.in Internet Source	<1%
8	Thangaprakash Sengodan, Sanjay Misra, M Murugappan. "Advances in Electrical and Computer Technologies", CRC Press, 2025 Publication	<1%
9	repository.iiitd.edu.in Internet Source	<1%
10	Douglas O'Shaughnessy. "Review of analysis methods for speech applications", Speech	<1%

Communication, 2023
Publication

11	Hiren Kumar Thakkar, Chintan Bhatt, Victor C.M. Leung, Ilangko Balasingham. "Health 5.0 - Concepts, Challenges, and Solutions", CRC Press, 2025 Publication	<1%
12	Yan Xu, Soobia Saeed. "chapter 10 Optimized Feature Extraction and Spatial-Temporal Analysis for Accurate Intracranial Hemorrhage Classification", IGI Global, 2025 Publication	<1%
13	Shalli Rani, Ayush Dogra, Ashu Taneja. "Smart Computing and Communication for Sustainable Convergence", CRC Press, 2025 Publication	<1%
14	pubmed.ncbi.nlm.nih.gov Internet Source	<1%
15	Submitted to Information Science and Technology Student Paper	<1%
16	Shankar Babu, Mahesh Babu Kota. "Synergies in Smart and Virtual Systems using computational intelligence", CRC Press, 2025 Publication	<1%
17	Arvind Dagur, Sohit Agarwal, Dhirendra Kumar Shukla, Shabir Ali, Sandhya Sharma. "Artificial Intelligence and Sustainable Innovation - Volume 1", CRC Press, 2026 Publication	<1%
18	pdfs.semanticscholar.org Internet Source	<1%
19	S.P. Jani, M. Adam Khan. "Applications of AI in Smart Technologies and Manufacturing", CRC Press, 2025 Publication	<1%

20	docslib.org Internet Source	<1%
21	Arvind Dagur, Karan Singh, Pawan Singh Mehra, Dhirendra Kumar Shukla. "Artificial Intelligence, Blockchain, Computing and Security", CRC Press, 2023 Publication	<1%
22	Ching, Serena Low Woan. "Detection of COVID-19 Pneumonia on Computed Tomography Images Using a Lightweight Deep Learning Model", University of Malaya (Malaysia) Publication	<1%
23	assets-eu.researchsquare.com Internet Source	<1%
24	ebin.pub Internet Source	<1%
25	iopscience.iop.org Internet Source	<1%
26	ugspace.ug.edu.gh Internet Source	<1%
27	www.medsci.org Internet Source	<1%
28	R. N. V. Jagan Mohan, B. H. V. S. Rama Krishnam Raju, V. Chandra Sekhar, T. V. K. P. Prasad. "Algorithms in Advanced Artificial Intelligence - Proceedings of International Conference on Algorithms in Advanced Artificial Intelligence (ICAAAI-2024)", CRC Press, 2025 Publication	<1%
29	Saxena, Ritwik Raj. "Automated Meal Analysis and Realtime Dietary Recommender System: Comparative Analysis of Machine Learning Models for Benchmarking Performance and	<1%

Innovation While Utilizing State-of-the-Art Algorithms for Caloric Content Recognition of Food for Alimentary Guidance in Support of Metabolic Disease Prevention and Management and of Promoting Overall Mindfulness of Fitness and Wellbeing.", University of Minnesota

Publication

Exclude quotes	On	Exclude matches	Off
Exclude bibliography	On		