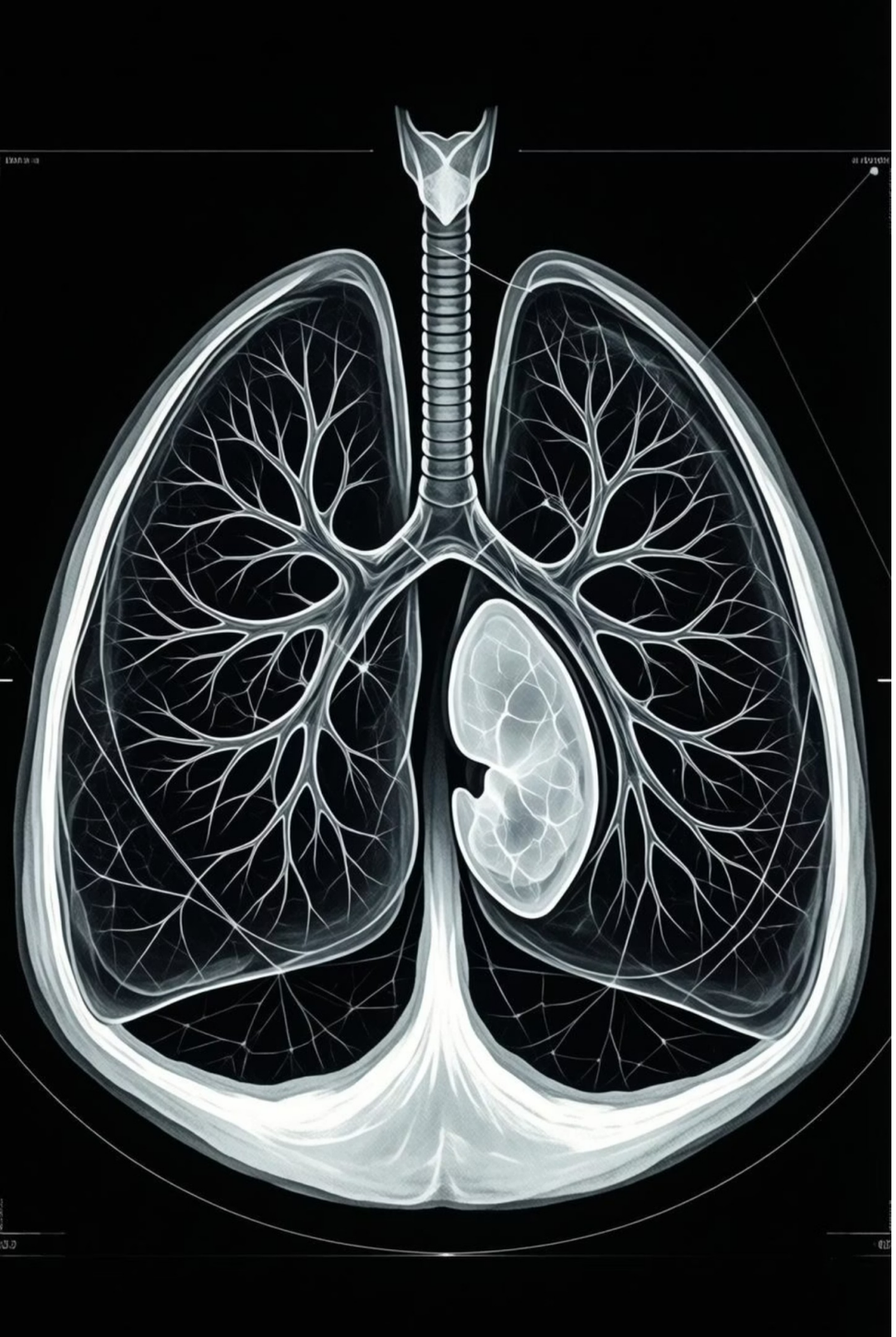


Lung Cancer Disease Prediction Using CNN

1. Basic Data

1

Domain & Technique

Leveraging Artificial Intelligence in Medical Imaging, specifically Convolutional Neural Networks (CNN) for image analysis.

2

Input & Dataset

Utilizes CT Scan Images from a publicly available dataset, comprising approximately 800 labeled images.

3

Classification

Identifies four key classes: Adenocarcinoma, Large Cell Carcinoma, Squamous Cell Carcinoma, and Normal lung tissue.

4

Deployment

The model is deployed via a user-friendly Flask-based Web Application for accessibility and interaction.

2. Introduction

Lung cancer remains a leading cause of cancer-related mortality globally. Early and accurate detection is critical for improving patient outcomes and survival rates.

Traditional methods of CT scan analysis are time-consuming and heavily reliant on expert radiologists, often leading to potential human error, especially in early disease stages.

- Time-consuming manual analysis
- Dependency on expert interpretation
- Risk of human error in diagnosis

Deep learning, particularly CNNs, offers a robust solution for enhancing medical image analysis and overcoming these challenges.

3. Abstract

This project develops a CNN-based system for lung cancer prediction, designed to identify visual patterns within CT scan images indicative of cancerous and non-cancerous conditions.

1

Image Preprocessing

CT scan images are prepared for optimal model input.

2

Multi-Class Classification

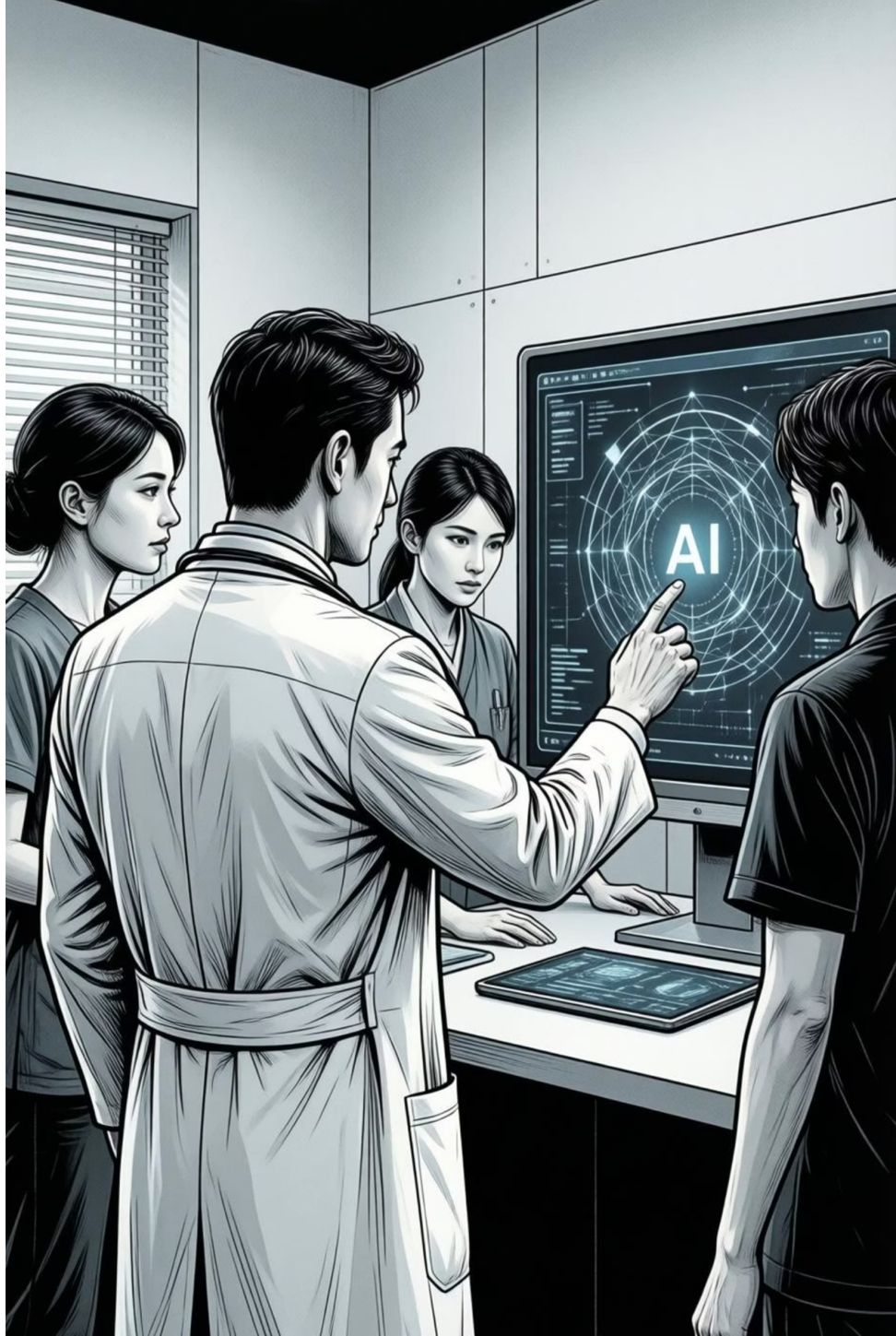
Images are categorized into specific lung cancer types or deemed normal.

3

Simplified Output

The final prediction is presented as either "Cancer Detected" or "No Cancer Detected."

The deployed web application offers an interactive interface. It functions purely as a decision-support tool, augmenting clinical judgment rather than replacing it.



4. Requirements

Hardware Requirements

- Standard PC / Laptop
- Minimum 8 GB RAM (recommended)

Software Requirements

- Python Environment
- TensorFlow & Keras
- NumPy, Matplotlib, Pillow
- Scikit-learn, OpenCV
- Flask Framework
- Web Browser

5. Literature Review

Traditional Diagnosis

Historically, lung cancer diagnosis relies on meticulous manual examination of CT scans by radiologists, a process prone to subjectivity and error.

Recent Advancements (IEEE)

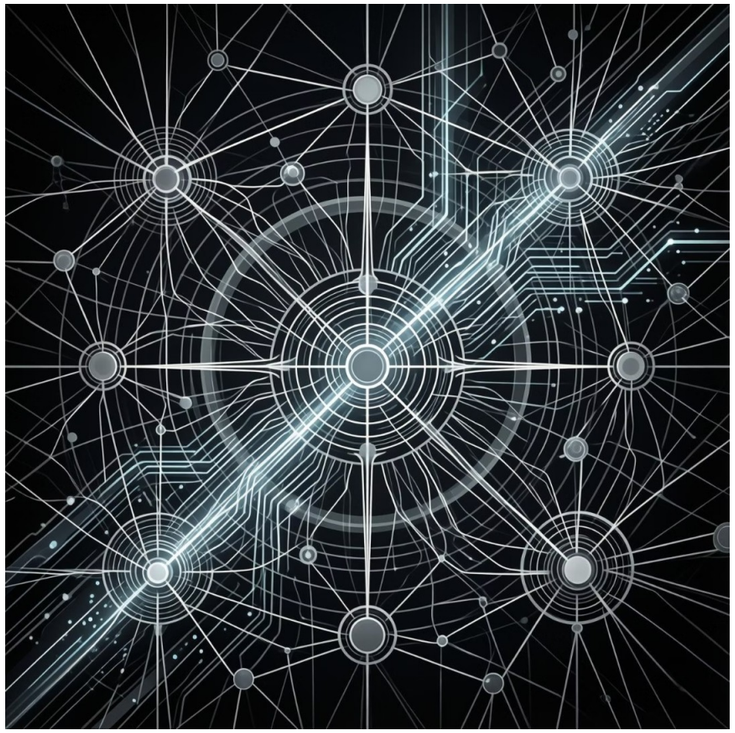
- Studies often employ advanced models like YOLOv8 for detection and staging.
- These methods typically demand extensive datasets and significant computational resources.

CNN-Based Classification

Convolutional Neural Networks offer a compelling alternative:

- Simpler architectural design
- Reduced training time and computational overhead
- Ideal for academic research and rapid prototyping

This project adopts a lightweight CNN approach, drawing inspiration from existing research to balance performance with computational efficiency.



6. Present Scope

Automated Prediction

Provides automated lung cancer prediction directly from CT scan images, streamlining the initial screening process.

Multi-Class Output

Performs multi-class classification, simplifying the results into a clear "Cancer Detected" or "No Cancer Detected" binary output.

User-Friendly Interface

Features an intuitive web interface for easy interaction and result visualization.

High Accuracy

Achieves approximately 93% accuracy on sample test images, demonstrating robust performance.

7. Future Scope

01

Input Validation

Implement robust validation to differentiate between lung and non-lung image inputs, preventing irrelevant analyses.

02

Tumor Localization

Integrate Grad-CAM to provide visual explanations of tumor locations within the CT scans, enhancing interpretability.

03

Subtype Reporting

Develop detailed, subtype-wise cancer reporting for more nuanced diagnostic information.

04

System Integration

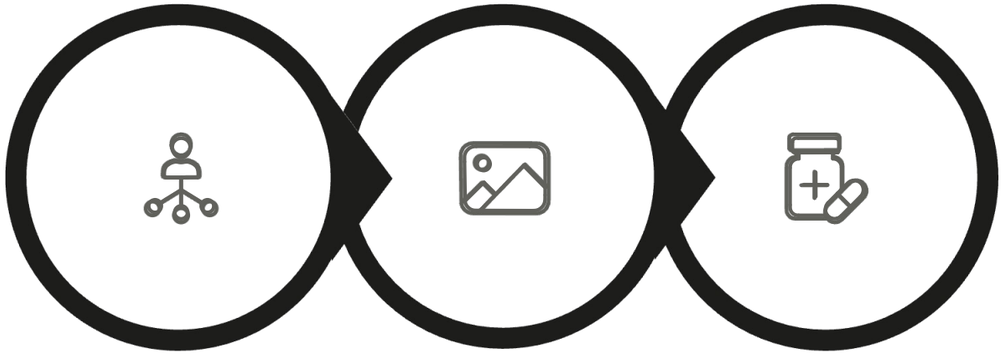
Facilitate seamless integration with existing hospital information systems for improved clinical workflows.

05

Clinical Dataset Training

Expand training to larger, more diverse clinical datasets for enhanced model generalization and real-world applicability.

CNN Workflow



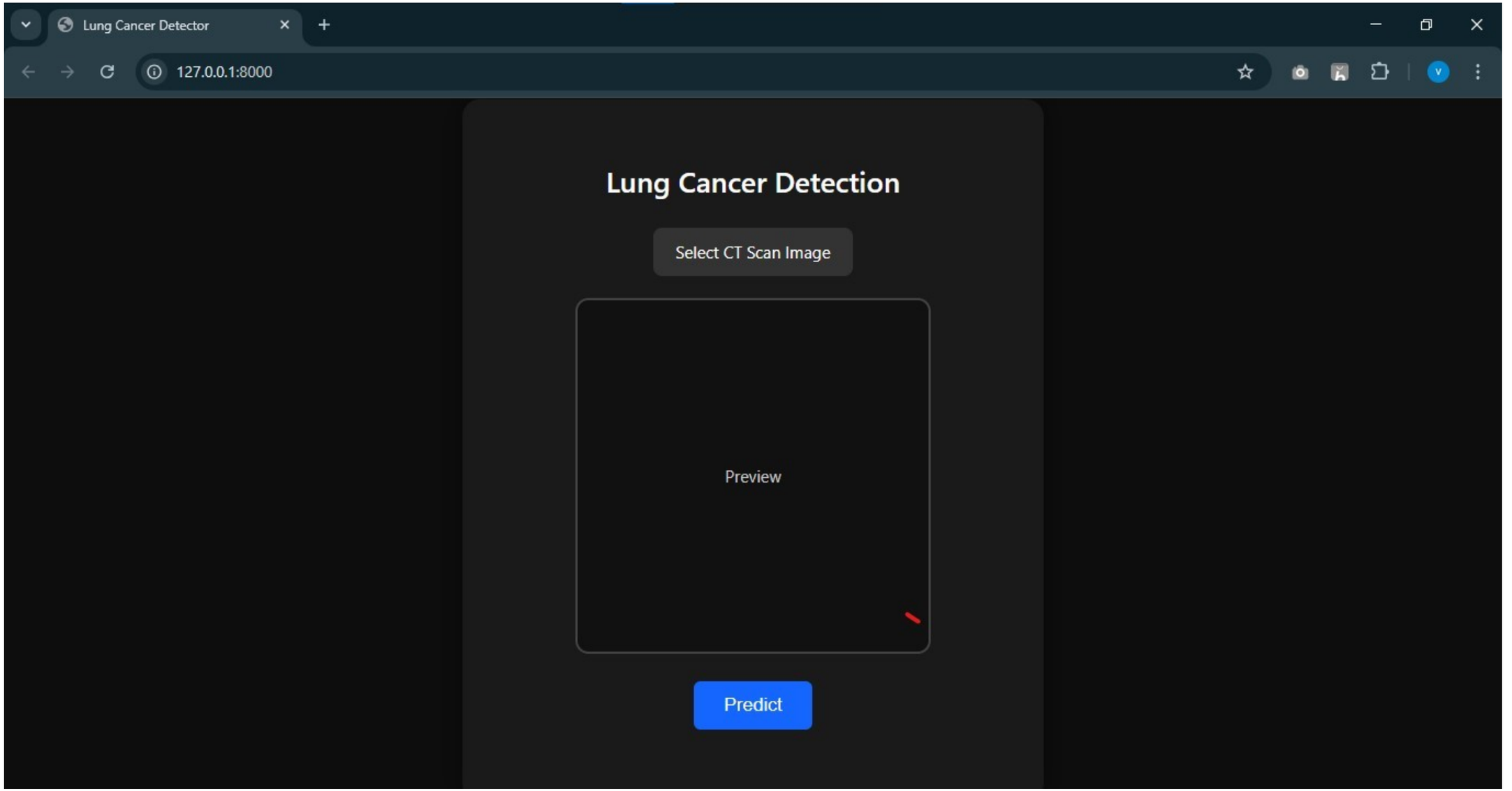
Training

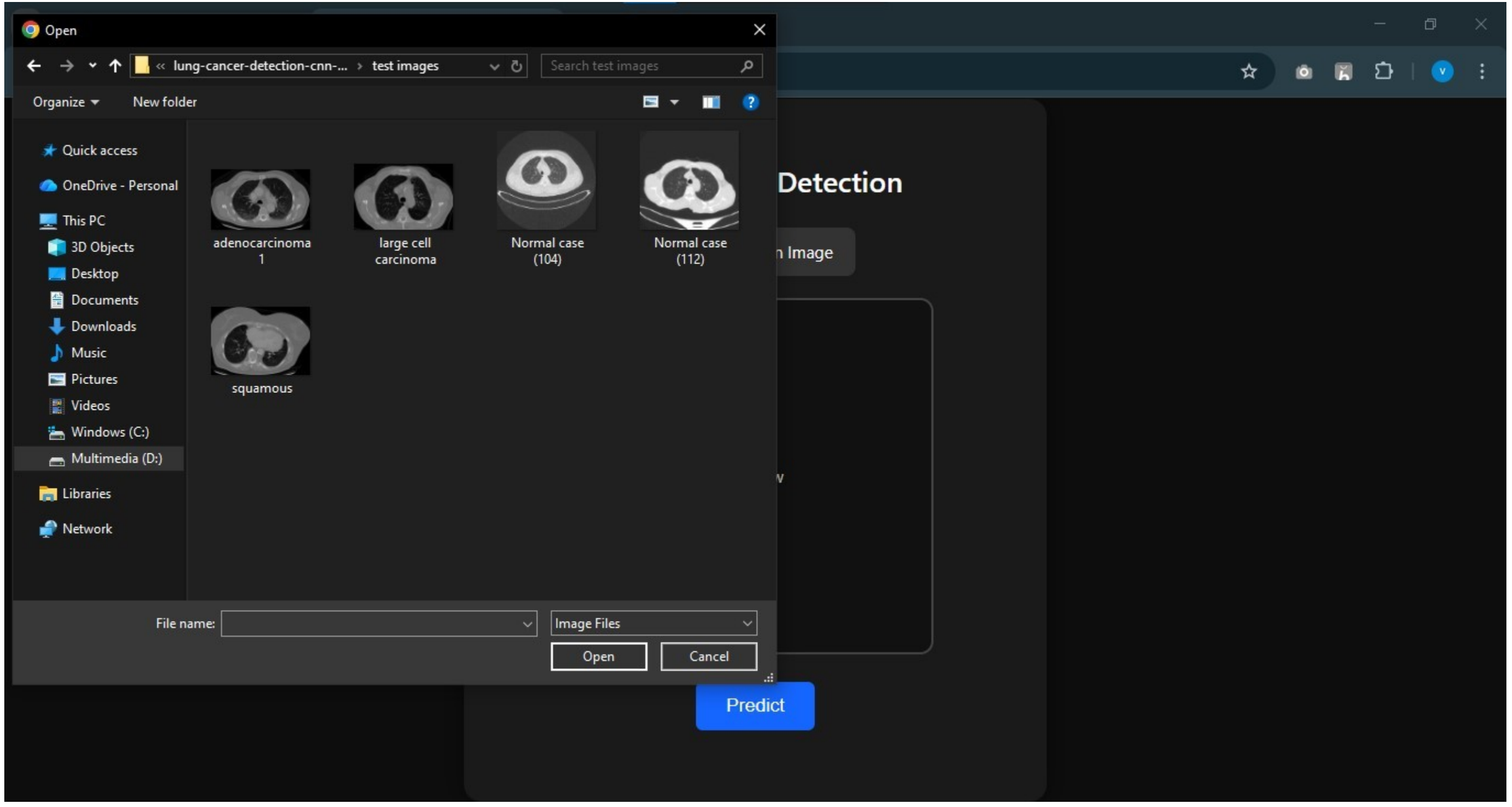
Preprocessing

Prediction

The CNN workflow begins with the Training Phase, where the model learns from labeled CT scan images. This is followed by the Preprocessing Phase for new input images, and finally the Prediction Phase, where the trained model classifies the prepared images.

PROJECT EXECUTION




Open
lung-cancer-detection-cnn-...
test images
Search test images
Organize
New folder
Quick access
OneDrive - Personal
This PC
3D Objects
Desktop
Documents
Downloads
Music
Pictures
Videos
Windows (C:)
Multimedia (D:)
Libraries
Network
adenocarcinoma 1
large cell carcinoma
Normal case (104)
Normal case (112)
squamous
File name:
Image Files
Open
Cancel
Detection
Image
Predict

Lung Cancer Detector

New Tab

127.0.0.1:8000

Lung Cancer Detection

Select CT Scan Image

Predict

Lung Cancer Detector

New Tab

127.0.0.1:8000

Lung Cancer Detection

Select CT Scan Image

Predict

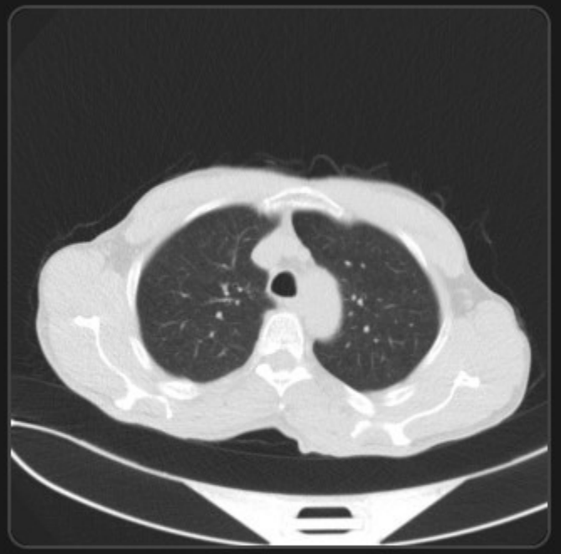
Prediction: Cancer
Confidence: 84.65%

Lung Cancer Detector

127.0.0.1:8000

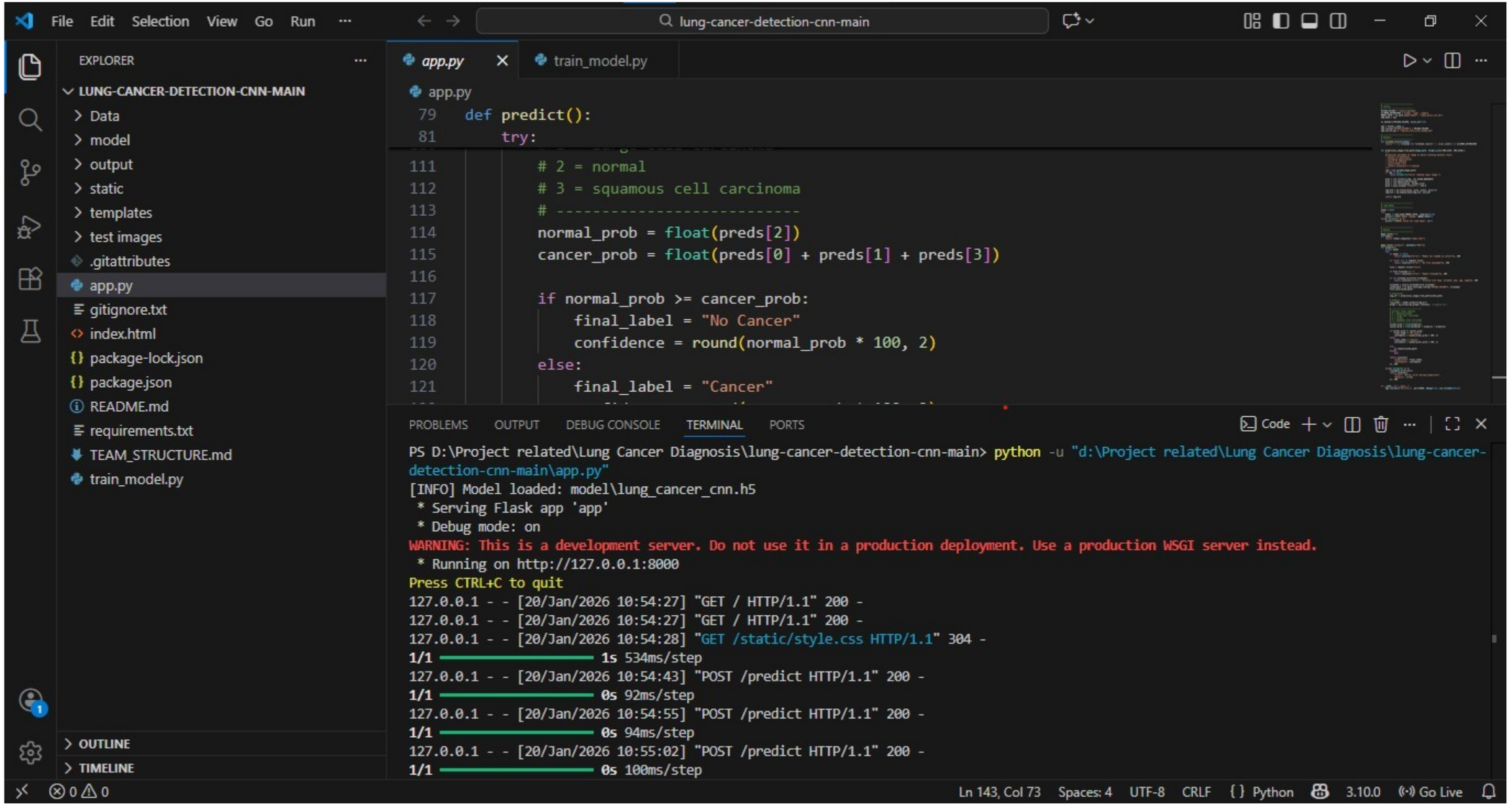
Lung Cancer Detection

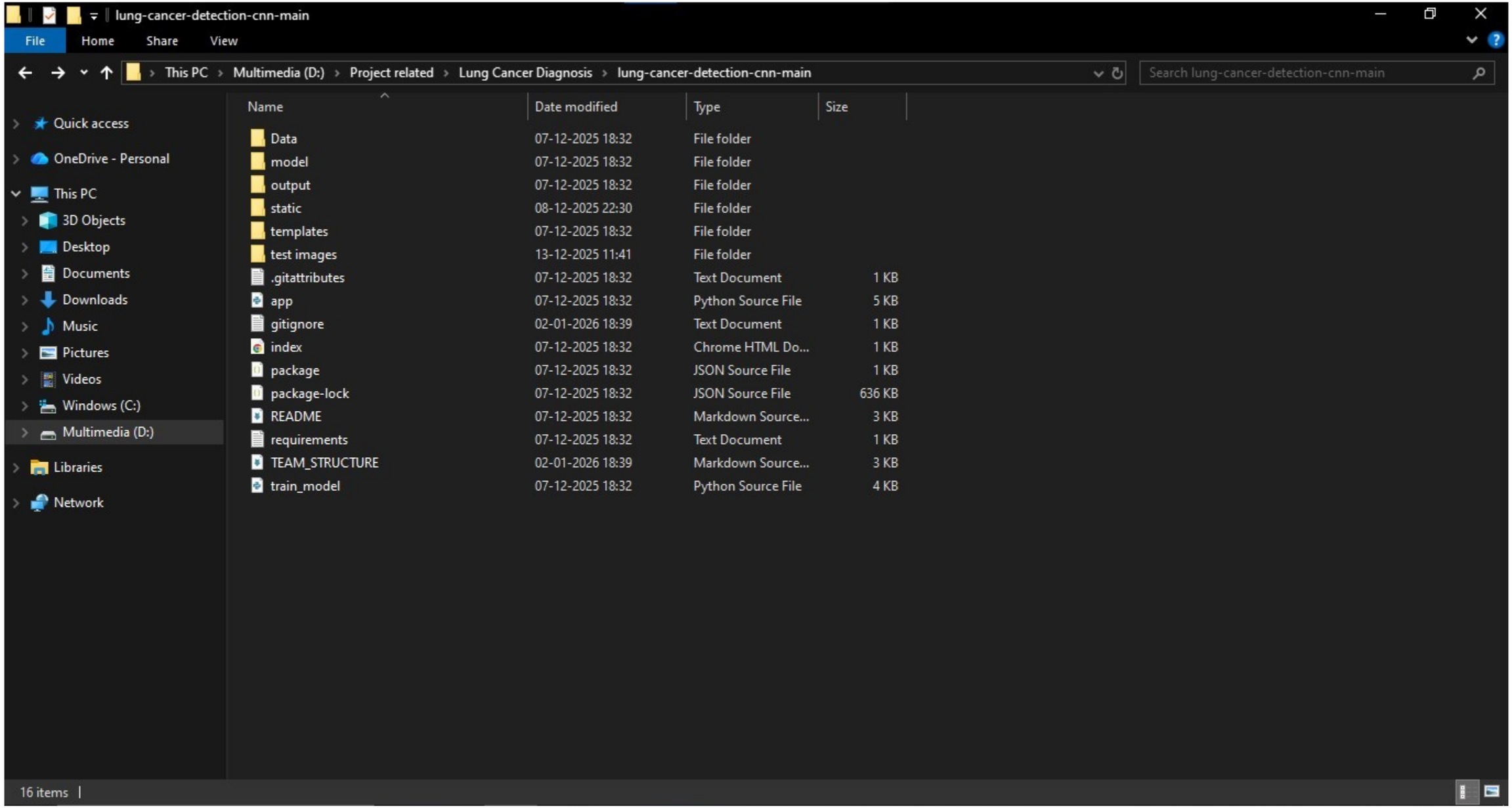
Select CT Scan Image

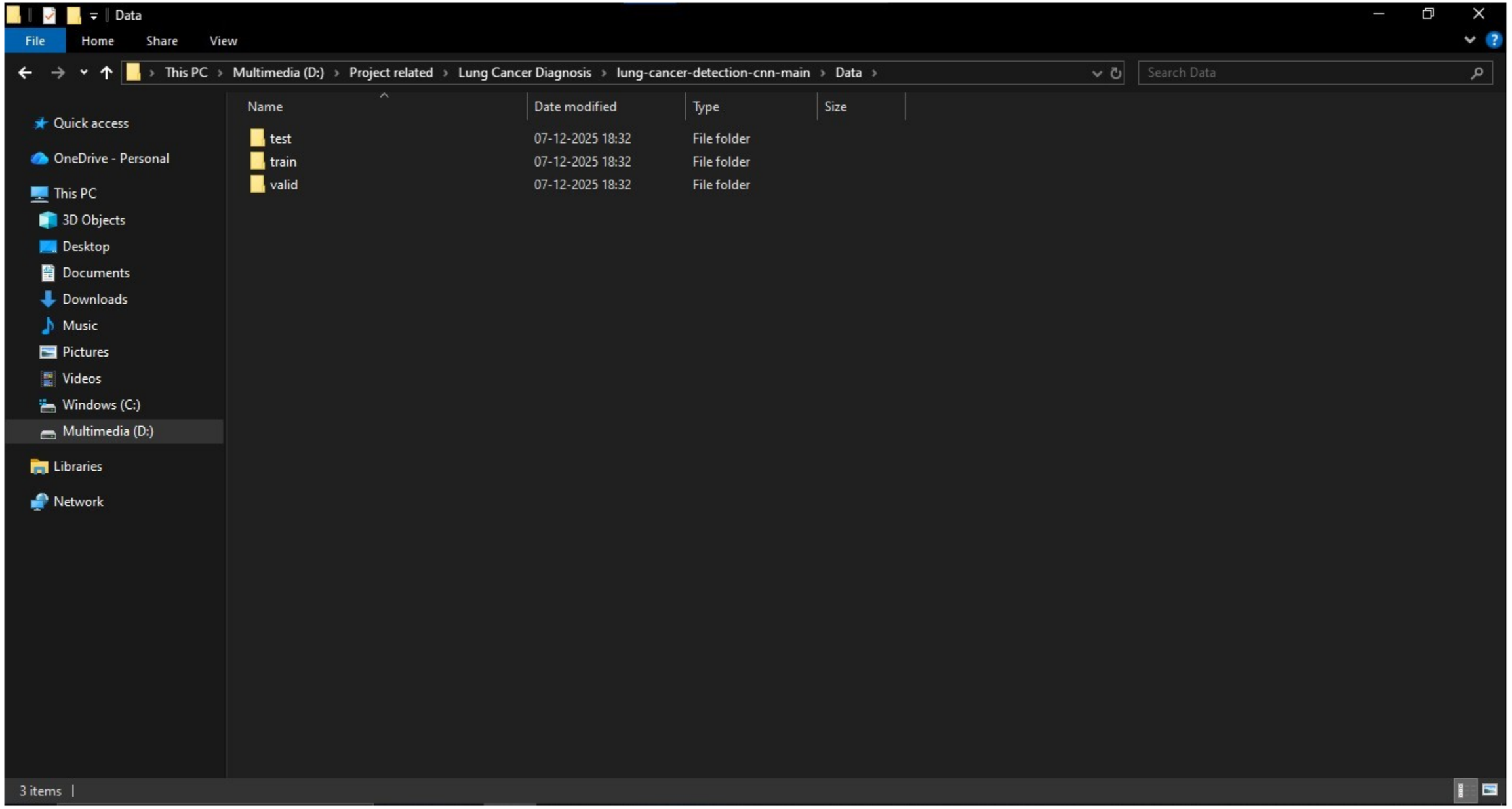


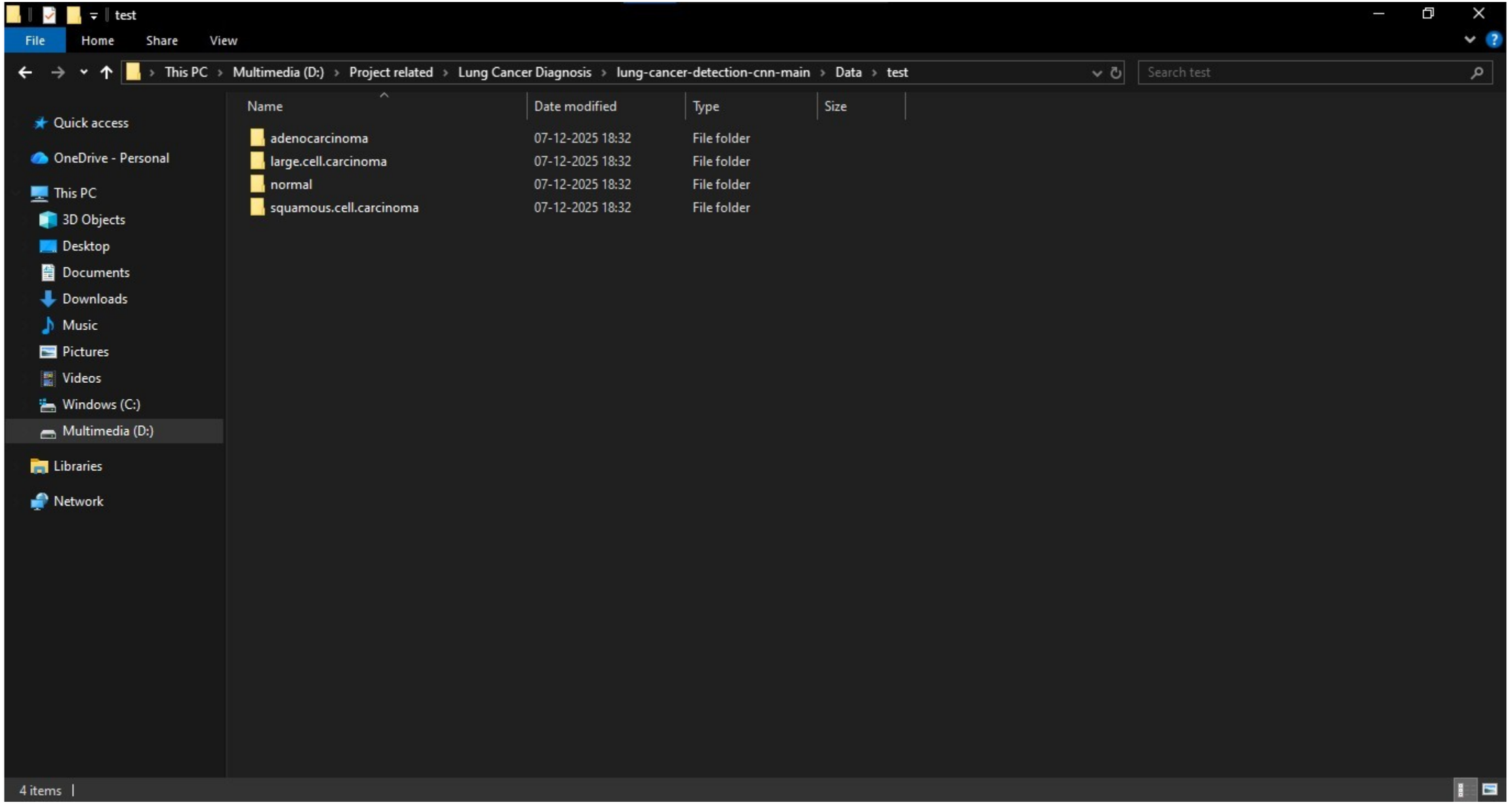
Predict

Prediction: No Cancer
Confidence: 100%

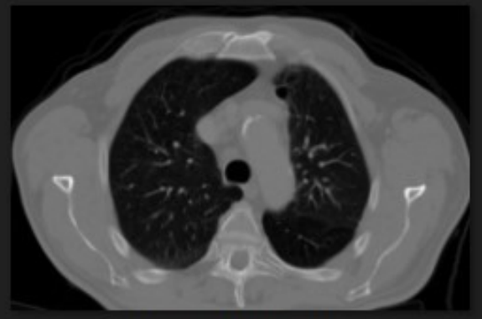

File Edit Selection View Go Run ···
lung-cancer-detection-cnn-main
EXPLORER
LUNG-CANCER-DETECTION-CNN-MAIN
Data
model
output
static
templates
test images
.gitattributes
app.py
gitignore.txt
index.html
package-lock.json
package.json
README.md
requirements.txt
TEAM_STRUCTURE.md
train_model.py
OUTLINE
TIMELINE
app.py
train_model.py
app.py
79 def predict():
81 try:
111 # 2 = normal
112 # 3 = squamous cell carcinoma
113 # ----------
114 normal_prob = float(preds[2])
115 cancer_prob = float(preds[0] + preds[1] + preds[3])
116
117 if normal_prob >= cancer_prob:
118 final_label = "No Cancer"
119 confidence = round(normal_prob * 100, 2)
120 else:
121 final_label = "Cancer"
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
Code
PS D:\Project related\Lung Cancer Diagnosis\lung-cancer-detection-cnn-main> python -u "d:\Project related\Lung Cancer Diagnosis\lung-cancer-detection-cnn-main\app.py"
[INFO] Model loaded: model\lung_cancer_cnn.h5
* Serving Flask app 'app'
* Debug mode: on
WARNING: This is a development server. Do not use it in a production deployment. Use a production WSGI server instead.
* Running on http://127.0.0.1:8000
Press CTRL+C to quit
127.0.0.1 - - [20/Jan/2026 10:54:27] "GET / HTTP/1.1" 200 -
127.0.0.1 - - [20/Jan/2026 10:54:27] "GET / HTTP/1.1" 200 -
127.0.0.1 - - [20/Jan/2026 10:54:28] "GET /static/style.css HTTP/1.1" 304 -
1/1 1s 534ms/step
127.0.0.1 - - [20/Jan/2026 10:54:43] "POST /predict HTTP/1.1" 200 -
1/1 0s 92ms/step
127.0.0.1 - - [20/Jan/2026 10:54:55] "POST /predict HTTP/1.1" 200 -
1/1 0s 94ms/step
127.0.0.1 - - [20/Jan/2026 10:55:02] "POST /predict HTTP/1.1" 200 -
1/1 0s 100ms/step
Ln 143, Col 73 Spaces: 4 UTF-8 CRLF Python 3.10.0 Go Live


lung-cancer-detection-cnn-main
File
Home
Share
View
This PC
Multimedia (D:)
Project related
Lung Cancer Diagnosis
lung-cancer-detection-cnn-main
Search lung-cancer-detection-cnn-main
Quick access
OneDrive - Personal
This PC
3D Objects
Desktop
Documents
Downloads
Music
Pictures
Videos
Windows (C:)
Multimedia (D:)
Libraries
Network
Name
Date modified
Type
Size
Data 07-12-2025 18:32 File folder
model 07-12-2025 18:32 File folder
output 07-12-2025 18:32 File folder
static 08-12-2025 22:30 File folder
templates 07-12-2025 18:32 File folder
test images 13-12-2025 11:41 File folder
.gitattributes 07-12-2025 18:32 Text Document 1 KB
app 07-12-2025 18:32 Python Source File 5 KB
gitignore 02-01-2026 18:39 Text Document 1 KB
index 07-12-2025 18:32 Chrome HTML Do... 1 KB
package 07-12-2025 18:32 JSON Source File 1 KB
package-lock 07-12-2025 18:32 JSON Source File 636 KB
README 07-12-2025 18:32 Markdown Source... 3 KB
requirements 07-12-2025 18:32 Text Document 1 KB
TEAM_STRUCTURE 02-01-2026 18:39 Markdown Source... 3 KB
train_model 07-12-2025 18:32 Python Source File 4 KB
16 items


Data
File
Home
Share
View
This PC
Multimedia (D:)
Project related
Lung Cancer Diagnosis
lung-cancer-detection-cnn-main
Data
Search Data
Quick access
OneDrive - Personal
This PC
3D Objects
Desktop
Documents
Downloads
Music
Pictures
Videos
Windows (C:)
Multimedia (D:)
Libraries
Network
Name
Date modified
Type
Size
test
07-12-2025 18:32
File folder
train
07-12-2025 18:32
File folder
valid
07-12-2025 18:32
File folder
3 items


test
File
Home
Share
View
This PC
Multimedia (D:)
Project related
Lung Cancer Diagnosis
lung-cancer-detection-cnn-main
Data
test
Search test
Quick access
OneDrive - Personal
This PC
3D Objects
Desktop
Documents
Downloads
Music
Pictures
Videos
Windows (C:)
Multimedia (D:)
Libraries
Network
Name
Date modified
Type
Size
adenocarcinoma 07-12-2025 18:32 File folder
large.cell.carcinoma 07-12-2025 18:32 File folder
normal 07-12-2025 18:32 File folder
squamous.cell.carcinoma 07-12-2025 18:32 File folder
4 items

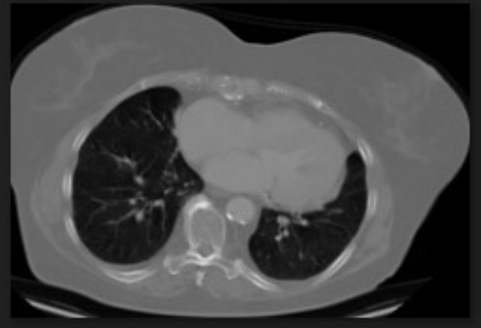
adenocarcinoma1



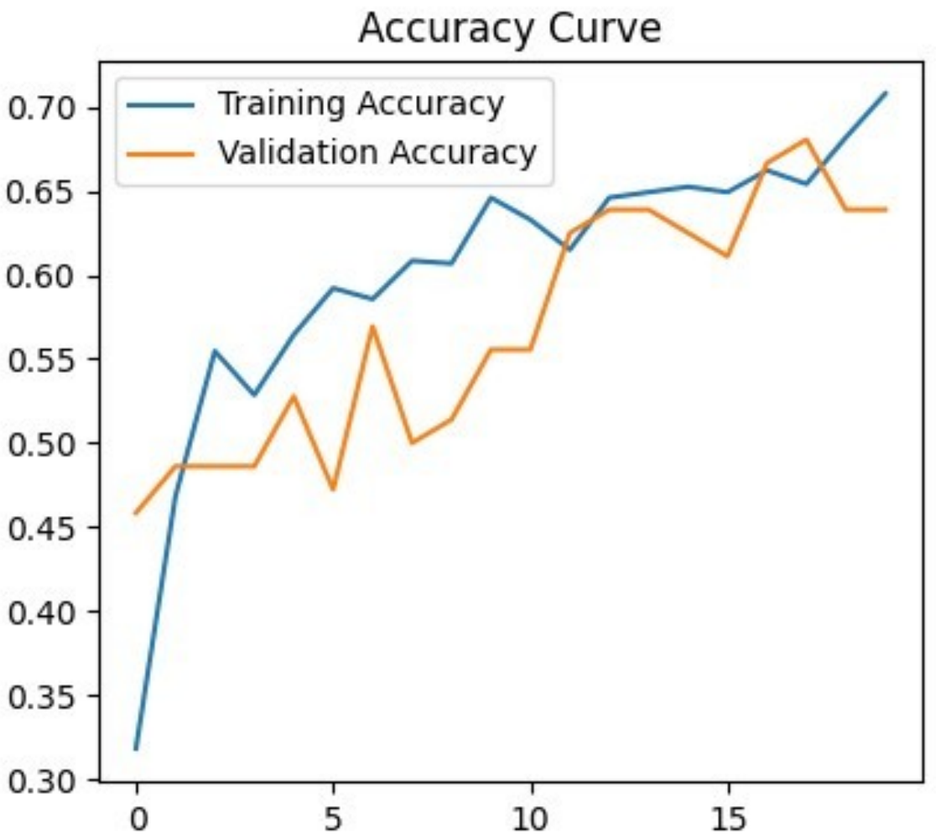
large cell carcinoma

Normal case (104)

Normal case (112)



squamous


Accuracy Curve
Training Accuracy
Validation Accuracy
0.70
0.65
0.60
0.55
0.50
0.45
0.40
0.35
0.30
0
5
10
15


Loss Curve
Training Loss
Validation Loss
1.4
1.3
1.2
1.1
1.0
0.9
0.8
0.7
0.6
0
5
10
15

8. Conclusion

- This project successfully demonstrates the efficacy of CNNs in lung cancer prediction, offering an automated solution for initial screening.
- Automated analysis significantly reduces manual workload and accelerates the diagnostic process.
- The CNN architecture provides reliable feature extraction from complex CT images, contributing to high prediction accuracy.

- While serving as a robust decision-support tool, this system is not intended to replace professional medical diagnosis.
- The project establishes a strong foundation for the continued development of AI-driven solutions in healthcare.