

Lung Cancer Detection From CT Images With Clinical Feature Fusion

Neelambika¹, Dr Laxmi Math²

¹M.Tech Student, Department of Artificial Intelligence and Data Science, Faculty of Engineering & Technology (Co-Ed), Sharanbasva University, KALABURAGI. neelambikaskolar5167@gmail.com

²Guide, Department of Artificial Intelligence and Data Science, Faculty of Engineering & Technology (Co-Ed), Sharanbasva University, KALABURAGI.

ABSTRACT

Lung cancer diagnosis from computed tomography (CT) images is challenging because early abnormalities may be subtle and manual review of large image volumes is time-consuming. The supplied thesis develops a web-based computer-aided diagnosis system that integrates CT image preprocessing, lung segmentation, image-feature extraction, and Convolutional Neural Network (CNN) classification. Python, Flask, TensorFlow, Keras, OpenCV, HTML, CSS, JavaScript, and SQLite are used to support image upload, preprocessing, inference, result visualization, and prediction-history storage. The system classifies CT images into Normal and Lung Cancer categories and reports confidence scores. All 12 functional test cases in the thesis passed, including image upload, preprocessing, segmentation, feature extraction, CNN prediction, report generation, and end-to-end workflow execution. The epoch-wise accuracy graph reaches 95.9% after 20 epochs. A separate model-performance dashboard reports 94.8% overall accuracy, 96.5% sensitivity, 93.2% specificity, and an AUC-ROC score of 0.975; a five-fold cross-validation summary reports 94.8% average validation accuracy. The developed framework is presented as a decision-support tool intended to assist, rather than replace, qualified healthcare professionals.

Keywords: Lung cancer detection; CT images; Convolutional Neural Network; Image processing; Deep learning; Computer-aided diagnosis

I. INTRODUCTION

Lung cancer remains a serious diagnostic challenge because delayed identification reduces the opportunity for timely intervention. CT imaging provides detailed cross-sectional views of lung tissue and pulmonary nodules, but a single examination can contain many image slices and therefore imposes a substantial review burden on radiologists. The thesis identifies noise, low contrast, imaging artifacts, overlapping structures, and subtle early-stage lesions as factors that can complicate manual interpretation.

Artificial Intelligence, image processing, and deep learning offer a route to automated support. The proposed system enhances CT images, isolates lung regions, extracts meaningful image characteristics, and applies a CNN to classify images as Normal or Lung Cancer. A Flask web interface is used to make the diagnostic workflow accessible, while SQLite stores prediction history.

Related work cited in the thesis includes the LIDC-IDRI reference database [1], multi-view CNN nodule detection [2], multi-crop CNN malignancy classification [3], multi-stream pulmonary nodule classification [4], CNN-based lung-pattern analysis [5], and broader demonstrations of deep learning in medical imaging [6]-[8].

II. OBJECTIVES

- To develop an automated lung cancer detection system using image processing techniques.
- To detect lung cancer from CT scan images with high accuracy.
- To preprocess CT scan images for noise removal and image enhancement.
- To perform lung segmentation for isolating the region of interest.
- To extract significant image features for accurate disease classification.
- To implement a Convolutional Neural Network (CNN) for lung cancer classification.
- To classify CT scan images into normal and cancerous categories.
- To develop a Flask-based web application for image upload and diagnosis.
- To reduce diagnosis time, assist radiologists, improve diagnostic accuracy, and maintain prediction records.

III. LITERATURE SURVEY

Armato et al. introduced the LIDC-IDRI database, a widely used reference resource for pulmonary nodule research [1]. Setio et al. used multi-view convolutional networks to reduce false positives in CT nodule detection [2], while Shen et al. used multi-crop CNNs for lung nodule malignancy suspiciousness classification [3]. Ciompi et al. further investigated deep learning for pulmonary nodule management in lung cancer screening [4].

Anthimopoulos et al. demonstrated CNN-based lung-pattern classification [5], and Esteva et al. showed that deep convolutional models can achieve strong performance in medical image classification [6]. Litjens et al. surveyed deep learning across medical image analysis tasks [7], and Ardila et al. developed an end-to-end deep learning system for low-dose CT lung-cancer screening [8]. These studies provide the technical background for the thesis focus on preprocessing, segmentation, CNN-based classification, and web-based result delivery.

IV. MATERIALS AND METHODOLOGY

System architecture

The diagnostic workflow begins when the user uploads a CT scan through the web interface. Flask forwards the image to preprocessing, where resizing, grayscale conversion, normalization, Gaussian filtering, contrast enhancement, and noise removal are applied. The lung segmentation module isolates the region of interest, after which texture, shape, intensity, edge, boundary, and structural characteristics are extracted. The processed image is then analyzed using a trained CNN implemented with TensorFlow and Keras. The predicted class and confidence information are displayed through the dashboard and stored in SQLite.

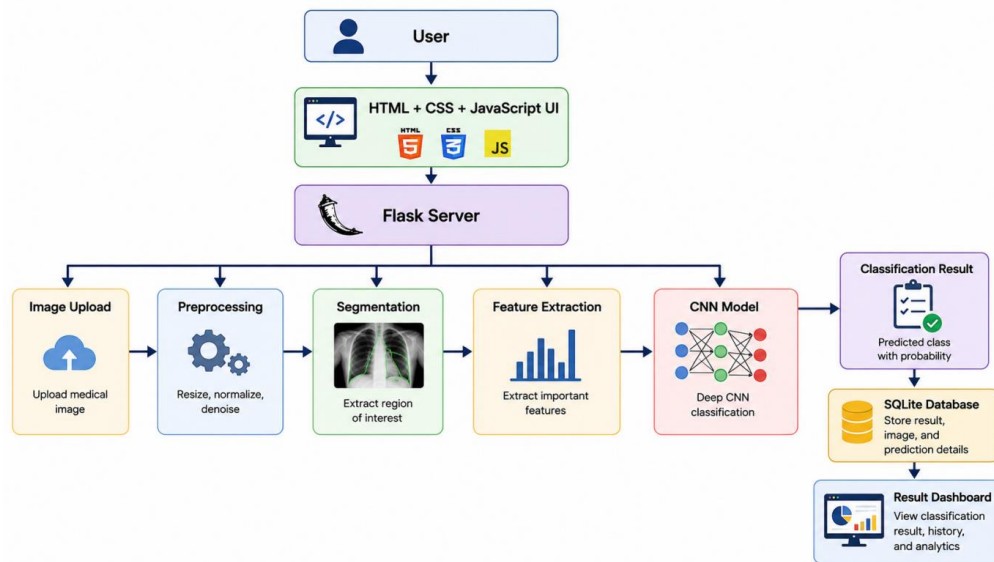


Fig. 1. System architecture retained from the thesis.

Implementation environment

Table 1. Software environment reported in the thesis

Software component	Specification
Operating System	Windows 10 / Windows 11
Programming Language	Python 3.10 or above
Backend Framework	Flask
Frontend Technologies	HTML5, CSS3, JavaScript
Deep Learning Framework	TensorFlow, Keras
Image Processing Library	OpenCV
Database	SQLite
IDE	Visual Studio Code
Web Browser	Google Chrome / Microsoft Edge

Source-data scope

The client-provided article title is preserved exactly. However, the supplied thesis documents CT-image preprocessing, segmentation, image-feature extraction, CNN classification, and web-based diagnosis. It does not document a clinical-variable dataset, a clinical-feature encoding pipeline, a CT/clinical fusion architecture, or fusion-specific experiments. Accordingly, this conversion does not invent or add unsupported clinical-feature-fusion methods or results.

Testing protocol

The thesis reports unit, integration, system, functional, performance, usability, and validation testing. Unseen CT scan images were used for validation testing, and classification results were compared with diagnostic labels. The functional test matrix covered image upload, preprocessing, segmentation, feature extraction, CNN

prediction, database storage, report generation, dashboard visualization, retrieval of previous predictions, and the complete diagnostic workflow.

V. RESULTS AND DISCUSSION

Functional testing

Table 2. Functional test cases reported in the thesis

ID	Test description	Expected result	Actual result	Status
TC01	Launch web application	Application opens successfully	Application launched	Pass
TC02	Upload CT scan image	Image uploaded successfully	Upload successful	Pass
TC03	Perform image preprocessing	Image enhanced correctly	Successful	Pass
TC04	Perform lung segmentation	Lung region extracted	Successful	Pass
TC05	Execute feature extraction	Image features extracted	Successful	Pass
TC06	CNN model prediction	Correct classification generated	Successful	Pass
TC07	Display prediction result	Prediction displayed	Successful	Pass
TC08	Store prediction history	Database updated	Successful	Pass
TC09	Generate diagnostic report	Report generated	Successful	Pass
TC10	Retrieve previous predictions	Prediction history displayed	Successful	Pass
TC11	Dashboard visualization	Dashboard updated	Successful	Pass
TC12	Complete diagnostic workflow	Entire process completed successfully	Successful	Pass

Table 3. Module-level test summary

Module	Result
Image Upload	Pass
Image Preprocessing	Pass
Lung Segmentation	Pass
Feature Extraction	Pass
CNN Classification	Pass
Prediction Visualization	Pass
Database Management	Pass
Report Generation	Pass
Overall System Performance	Pass

All 12 functional test cases were reported as Pass. The module-level summary likewise reports successful operation for image upload, preprocessing, lung segmentation, feature extraction, CNN classification, prediction visualization, database management, report generation, and overall system performance.

CNN accuracy over training epochs

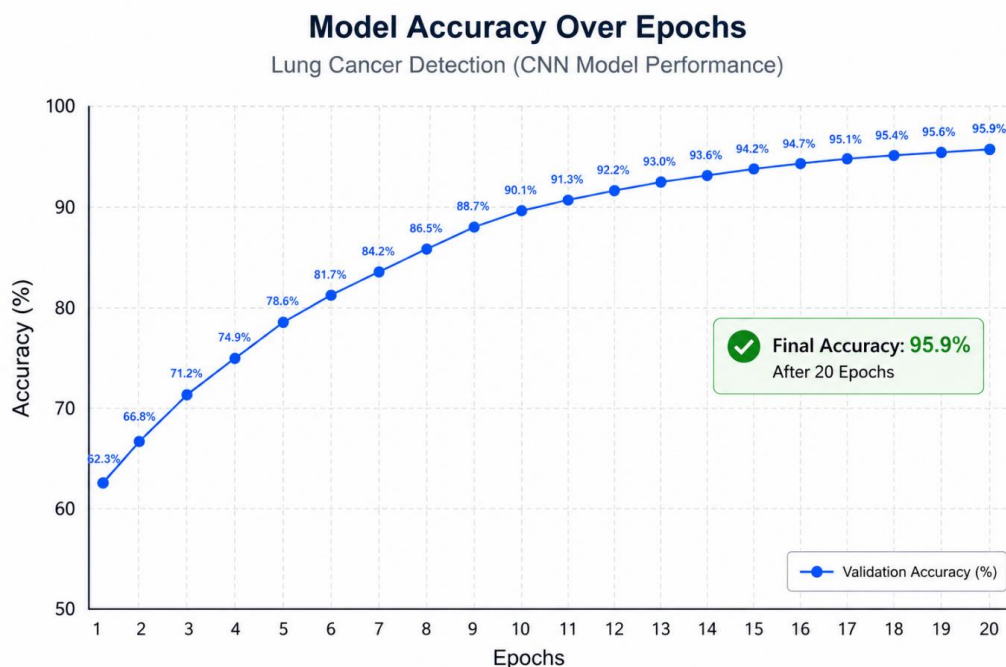


Fig. 2. Model accuracy over 20 epochs retained from the thesis.

The thesis accuracy graph shows validation accuracy increasing from 62.3% at epoch 1 to 95.9% at epoch 20. The curve rises steadily throughout training, with the graph explicitly reporting a final accuracy of 95.9% after 20 epochs.

Model performance metrics

Table 4. Model-performance values shown in the thesis dashboard

Metric	Reported value
Overall accuracy	94.8%
Sensitivity	96.5%
Specificity	93.2%
AUC-ROC score	0.975

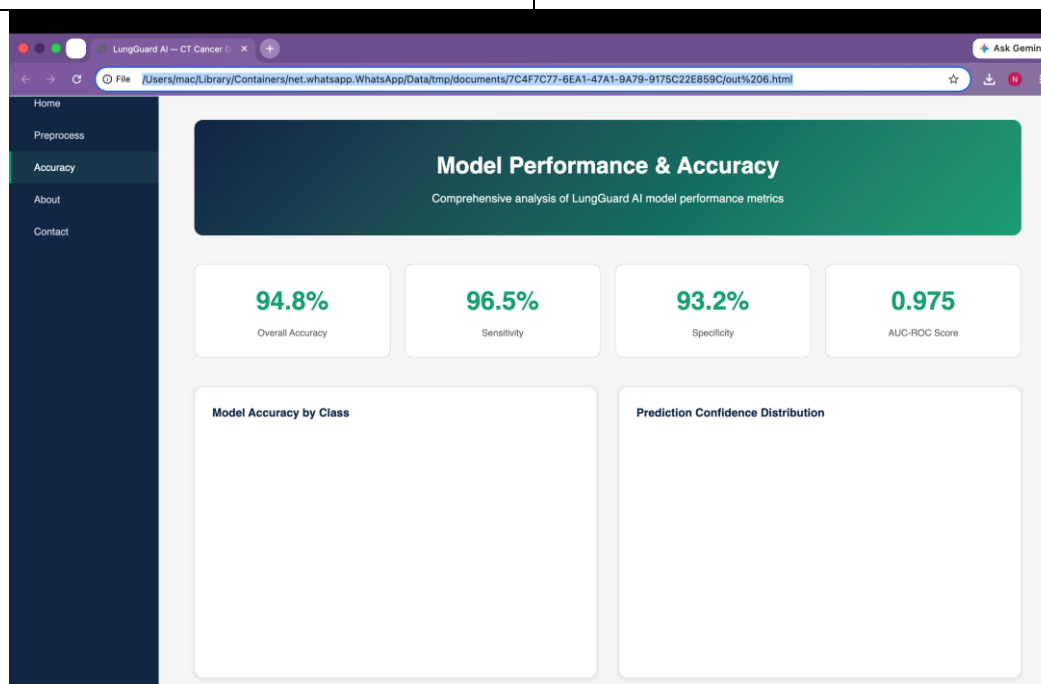


Fig. 3. Model performance and accuracy dashboard retained from the thesis.

Five-fold cross-validation

Table 5. Five-fold cross-validation results shown in the thesis

Fold	Training accuracy	Validation accuracy	Validation loss	Time
Fold 1	96.2%	94.5%	0.142	2.34 s
Fold 2	96.8%	94.9%	0.128	2.41 s
Fold 3	95.9%	94.2%	0.151	2.38 s
Fold 4	96.5%	95.3%	0.135	2.35 s
Fold 5	96.1%	94.8%	0.139	2.37 s
Average	96.3%	94.8%	0.139	2.37 s

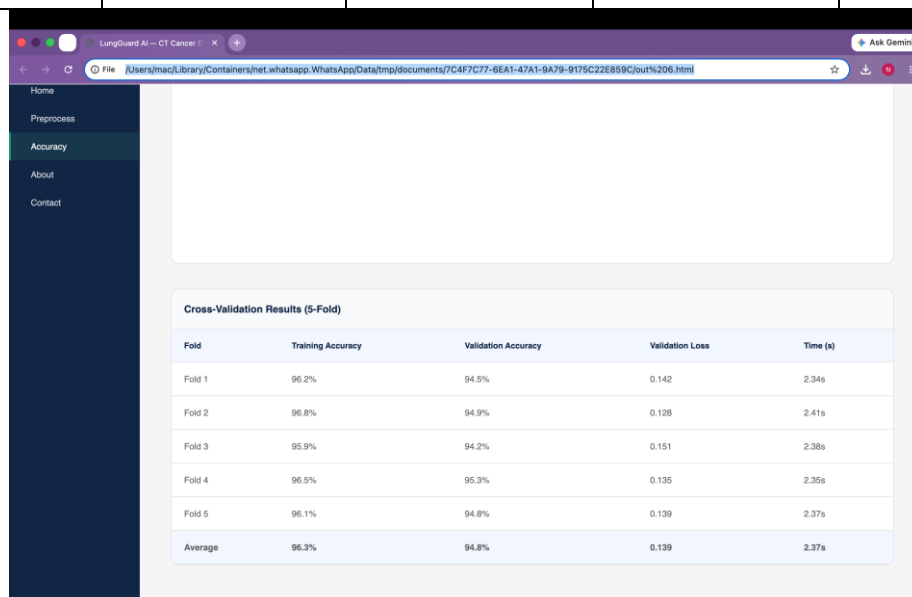


Fig. 4. Five-fold cross-validation results retained from the thesis.

The five-fold summary reports an average training accuracy of 96.3%, average validation accuracy of 94.8%, average validation loss of 0.139, and average time of 2.37 s. The thesis also presents a 95.9% final epoch-wise accuracy in Fig. 2. These two performance views are reproduced as reported; the thesis does not state that they use the same evaluation partition, so they are not merged into a single value.

System output

Fig. 5. CT image upload and patient-information interface retained from the thesis.

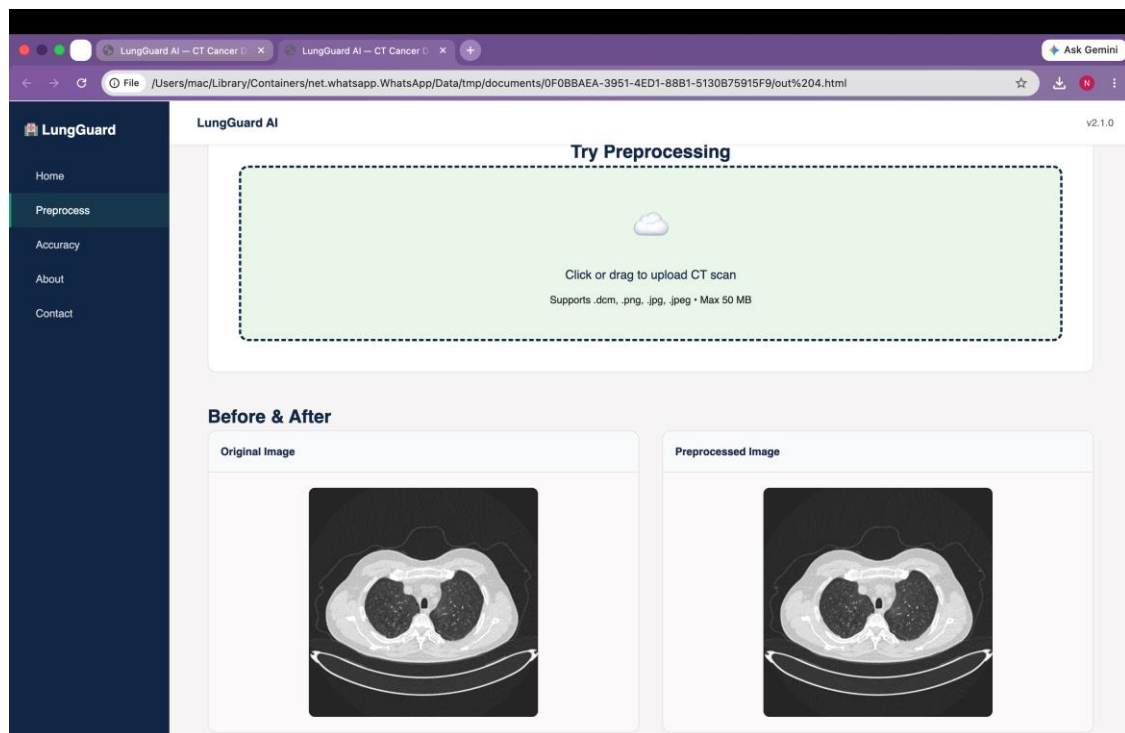


Fig. 6. Before-and-after preprocessing display retained from the thesis.

The implementation screenshots demonstrate the web-based diagnostic workflow, including CT image upload, preprocessing visualization, prediction display, and model-performance reporting. The interface also contains patient-information fields; however, the thesis does not describe these fields as inputs to the CNN or as a formal clinical-feature-fusion branch.

Discussion

The retained evidence supports the thesis claim that the application successfully integrates image preprocessing, segmentation, CNN classification, data storage, and web visualization. The reported test matrix demonstrates end-to-end functional execution, while the training and cross-validation outputs provide quantitative evidence of classification performance. The system is positioned as a computer-aided decision-support tool that can reduce manual screening burden and provide rapid preliminary analysis, while final diagnosis and treatment decisions remain the responsibility of qualified medical professionals.

VI. CONCLUSION

- The supplied thesis successfully implements an automated CT-image lung cancer detection workflow using image processing and CNN classification.
- Preprocessing includes grayscale conversion, resizing, normalization, noise reduction, Gaussian filtering, and contrast enhancement, followed by lung segmentation and feature extraction.
- All 12 reported functional test cases passed, and all listed system modules received Pass outcomes.
- The epoch-wise accuracy graph reports 95.9% final accuracy after 20 epochs.
- The model-performance dashboard reports 94.8% overall accuracy, 96.5% sensitivity, 93.2% specificity, and an AUC-ROC score of 0.975.
- Five-fold cross-validation reports 94.8% average validation accuracy and 0.139 average validation loss.
- The thesis supports CT-image processing and CNN-based classification; no unsupported clinical-feature-fusion experiment has been introduced in this article conversion.

SUPPLEMENTARY SYSTEM DESIGN

The following design diagrams are retained as supporting implementation evidence because they document data flow, database organization, and user-system interaction.

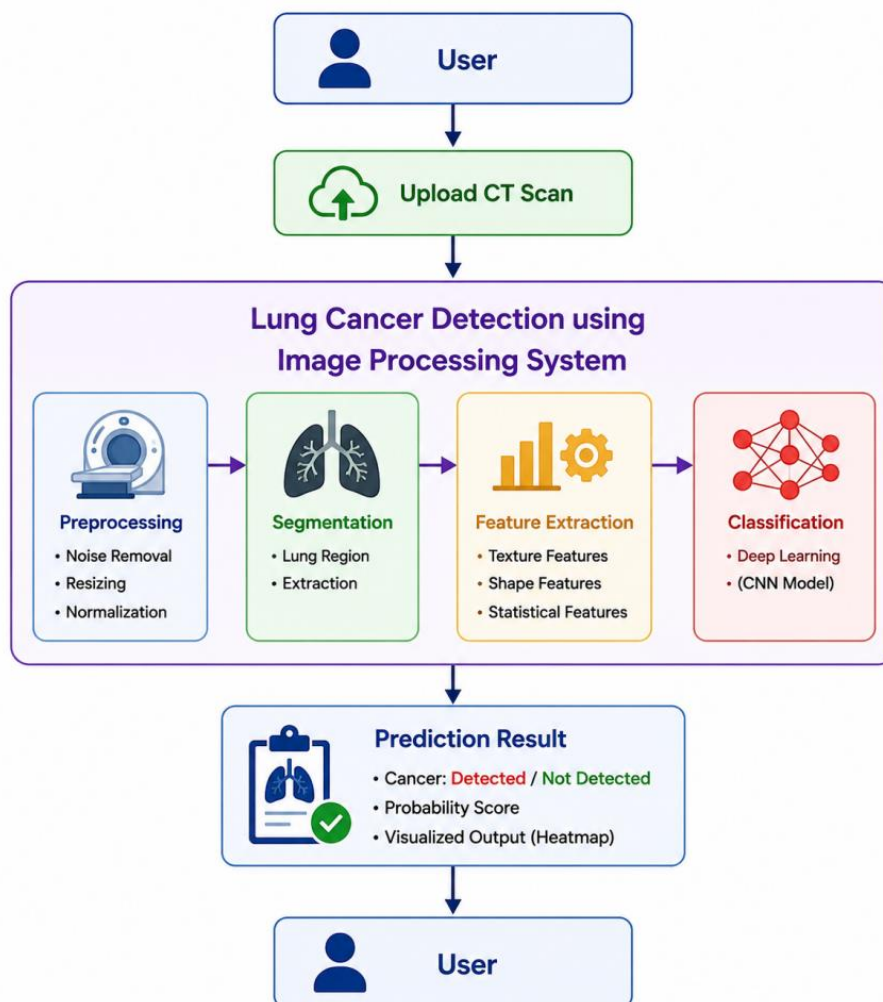


Fig. S1. Data Flow Diagram retained from the thesis.

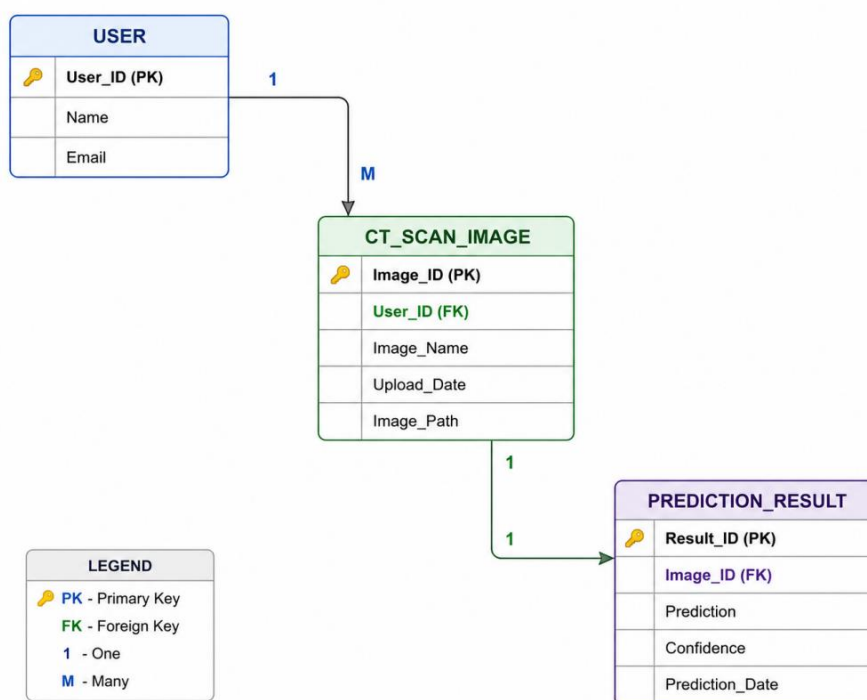


Fig. S2. Entity Relationship Diagram retained from the thesis.

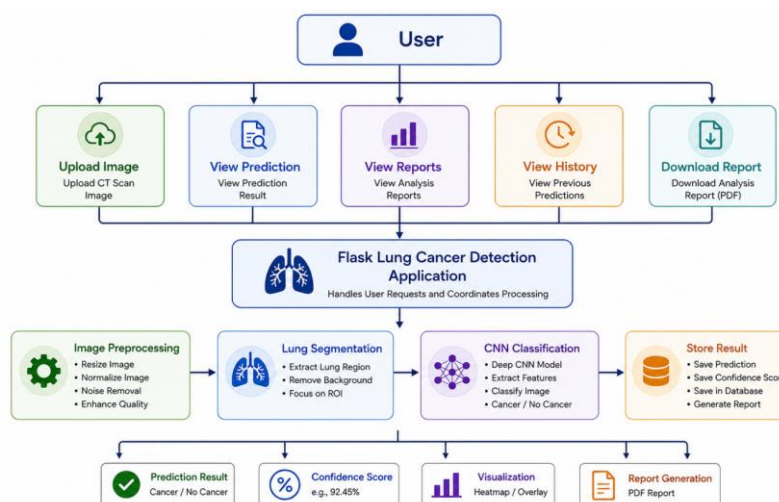


Fig. S3. Use Case Diagram retained from the thesis.

REFERENCES

- [1] S. G. Armato III, G. McLennan, L. Bidaut, M. F. McNitt-Gray, C. R. Meyer, A. P. Reeves, et al., "The Lung Image Database Consortium (LIDC) and Image Database Resource Initiative (IDRI): A Completed Reference Database of Lung Nodules on CT Scans," *Medical Physics*, vol. 38, no. 2, pp. 915–931, 2011.
- [2] A. A. A. Setio, F. Ciompi, G. Litjens, P. Gerke, C. Jacobs, S. J. van Riel, et al., "Pulmonary Nodule Detection in CT Images: False Positive Reduction Using Multi-View Convolutional Networks," *IEEE Transactions on Medical Imaging*, vol. 35, no. 5, pp. 1160–1169, 2016.
- [3] W. Shen, M. Zhou, F. Yang, C. Yang, and J. Tian, "Multi-Crop Convolutional Neural Networks for Lung Nodule Malignancy Suspiciousness Classification," *Pattern Recognition*, vol. 61, pp. 663–673, 2017.
- [4] F. Ciompi, K. Chung, S. J. van Riel, A. A. A. Setio, C. Gerke, C. Jacobs, et al., "Towards Automatic Pulmonary Nodule Management in Lung Cancer Screening with Deep Learning," *Scientific Reports*, vol. 7, Article 46479, 2017.
- [5] M. Anthimopoulos, S. Christodoulidis, L. Ebner, A. Christe, and S. Mougiakakou, "Lung Pattern Classification for Interstitial Lung Diseases Using a Deep Convolutional Neural Network," *IEEE Transactions on Medical Imaging*, vol. 35, no. 5, pp. 1207–1216, 2016.
- [6] A. Esteva, B. Kuprel, R. A. Novoa, J. Ko, S. M. Swetter, H. M. Blau, and S. Thrun, "Dermatologist-Level Classification of Skin Cancer with Deep Neural Networks," *Nature*, vol. 542, no. 7639, pp. 115–118, 2017.
- [7] G. Litjens, T. Kooi, B. E. Bejnordi, A. A. A. Setio, F. Ciompi, M. Ghafoorian, et al., "A Survey on Deep Learning in Medical Image Analysis," *Medical Image Analysis*, vol. 42, pp. 60–88, 2017.
- [8] D. Ardila, A. P. Kiraly, S. Bharadwaj, B. Choi, J. Reicher, L. Peng, et al., "End-to-End Lung Cancer Screening with Three-Dimensional Deep Learning on Low-Dose Chest Computed Tomography," *Nature Medicine*, vol. 25, no. 6, pp. 954–961, 2019.
- [9] O. Ronneberger, P. Fischer, and T. Brox, "U-Net: Convolutional Networks for Biomedical Image Segmentation," *Proceedings of the International Conference on Medical Image Computing and Computer-Assisted Intervention (MICCAI)*, pp. 234–241, 2015.
- [10] K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition," *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 770–778, 2016.
- [11] M. Abadi, P. Barham, J. Chen, Z. Chen, A. Davis, J. Dean, et al., "TensorFlow: A System for Large-Scale Machine Learning," *Proceedings of the 12th USENIX Symposium on Operating Systems Design and Implementation (OSDI)*, pp. 265–283, 2016.
- [12] D. P. Kingma and J. Ba, "Adam: A Method for Stochastic Optimization," *International Conference on Learning Representations (ICLR)*, 2015.
- [13] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. Cambridge, MA, USA: MIT Press, 2016.
- [14] M. Grinberg, *Flask Web Development: Developing Web Applications with Python*, 2nd ed. O'Reilly Media, 2018.
- [15] E. Matthes, *Python Crash Course*, 3rd ed. No Starch Press, 2023.
- [16] J. C. Russ, *The Image Processing Handbook*, 7th ed. CRC Press, 2016.
- [17] R. C. Gonzalez and R. E. Woods, *Digital Image Processing*, 4th ed. Pearson Education, 2018.
- [18] R. Elmasri and S. B. Navathe, *Fundamentals of Database Systems*, 7th ed. Pearson Education, 2016.
- [19] J. Duckett, *HTML and CSS: Design and Build Websites*. John Wiley & Sons, 2011.
- [20] D. Flanagan, *JavaScript: The Definitive Guide*, 7th ed. O'Reilly Media, 2020.