Lung Cancer Detection using CT Scan Images by Machine Learning

Mohamed Mostafa, Romaisaa Shrief, Youssef Kadry, Zeyad Amr
Systems and Biomedical Engineering
Faculty of Engineering Cairo University

Cairo, Egypt

Mohamed.Ahmed019@eng-st.cu.edu.eg,
romaisaa.elsaidy01@eng-st.cu.edu.eg,
yousefkadry01@gmail.com,
zeyadamr020@gmail.com

***Abstract*—Lung cancer is one of the leading causes of cancer-related deaths worldwide. Early detection plays a crucial role in improving patient outcomes and survival rates. This paper presents a machine learning-based approach for lung cancer detection using computed tomography (CT) scans. The methodology employed in this research is image-processing/computer-vision techniques for preprocessing involving smoothing, thresholding, edge detection and dilation. After applying these techniques, the preprocessed data is used for extracting GLCM features that are used in training different models with different hyperparameters to obtain the best results. Random forest is used as a final model as it provides real-time feedback to users for identifying the cases on the slides as one of three classes: normal, benign, or malignant with overall classification accuracy of 84.0%.**

I. Introduction

Lung cancer, one of the most lethal diseases, is responsible for causing the death of approximately one million people each year. With the increasing prevalence of lung nodules, it has become critically important to perform lung nodule identification on chest CT scans. To achieve the objective of early lung cancer detection, the deployment of computer-aided detection (CAD) systems has become essential.

Cancer detection is a complex task that requires accurate and efficient methods to identify and classify cancerous cells or tumors. Traditional diagnostic approaches often rely on manual examination and subjective interpretation, which can be time-consuming and prone to human error.

Machine learning, with its ability to learn patterns and make predictions from large datasets, has emerged as a promising tool for cancer detection.

The Iraq-Oncology Teaching Hospital/National Center Lung Cancer Dataset provides a valuable resource for researchers to develop and evaluate machine learning models for lung cancer detection.

This dataset contains a diverse collection of lung CT scan images, along with corresponding labels indicating the presence or absence of lung cancer. By leveraging this dataset, machine learning algorithms can be trained and evaluated to accurately classify lung cancer cases.

II. Related Work

A. Approaches

There are two main approaches to lung cancer detection using CT scan:

1) Image-based approaches: extract features from CT images and use machine learning algorithms to classify them as either normal or cancerous. These approaches have been shown to be effective in detecting lung cancer, but they can be computationally expensive and require a large amount of training data.
2) Deep learning approaches: use deep neural networks to learn the features that are predictive of lung cancer. These approaches have been shown to be more accurate than image-based approaches, but they require even more training data.

B. Studies

There have been many studies that have investigated the use of machine learning and deep learning for lung cancer detection using CT scan:

1) Lung Cancer Classification and Prediction Using Machine Learning and Image Processing [1]

This study focuses on the early detection of lung cancer through the use of CAD systems and image processing techniques applied to chest CT scans. It emphasizes the importance of accurate lung cancer classification and prediction using machine learning and image processing technologies. The preprocessing involves the application of a geometric mean filter for noise reduction and the use of the K-means approach for image segmentation. Machine learning algorithms, including Artificial Neural Networks (ANN), are employed for classification, with ANN showing higher accuracy in predicting lung cancer.

Advantages of this study include the potential for improved accuracy in lung cancer detection and the utilization of cutting-edge machine learning techniques. However, it is important to note the limitations of the study, such as the sample size and the lack of detailed performance metrics or comparative analysis of the classification models.

Our study, on the other hand, focuses on lung cancer detection using a machine learning approach and preprocessing techniques that involve blurring images and performing thresholding to binarize the data. It utilizes a weighted random forest machine learning model for training.

The similarities between this and our study include their common objective of lung cancer detection using machine learning algorithms. Both projects also employ image preprocessing techniques to enhance the quality and segment the images. However, the specific preprocessing techniques differ, as our study emphasizes blurring and thresholding, while this study focuses on noise reduction and segmentation using the K-means approach. Additionally, our study utilizes a weighted random forest machine learning model, whereas this study explores the use of ANN, KNN, and RF models for classification.

2) A Neural Network and Optimization Based Lung Cancer Detection System in CT Images [2]

This study proposes an effective lung cancer screening procedure using image processing and machine learning techniques. The procedure involves contrast enhancement and noise reduction through pre-processing, Otsu thresholding segmentation using the cuckoo search algorithm, retrieval of regions of concern, retrieval of descriptors using local binary patterns (LBP), and classification using a convolutional neural network (CNN). The proposed framework achieves a high accuracy of 96.97%.

Advantages of this study include the use of advanced image processing techniques and the application of deep learning through CNN for accurate lung cancer classification. The study demonstrates improved accuracy and presents results obtained using particle swarm optimization and genetic algorithms. However, potential disadvantages include the lack of detailed information on the dataset used and the absence of comparative analysis with other existing methods.

Our study, on the other hand, involves a machine learning approach with preprocessing techniques of blurring images and thresholding to binarize the data, followed by training using a weighted random forest machine learning model.

The similarities between this study and our study lie in their common goal of lung cancer detection using machine learning algorithms. Both projects also involve preprocessing steps to enhance the quality and segment the images. However, the specific preprocessing techniques differ, as our study emphasizes blurring and thresholding, while this study utilizes contrast enhancement, noise reduction, and Otsu thresholding segmentation. Additionally, our study employs a weighted random forest model, while this study utilizes a CNN model.

3) A Hybrid Deep Learning Model for Lung Nodule Detection and Classification in CT Images [3]

This study focuses on the detection of malignant lung nodules from CT images using a hybrid model called VCNet, which combines the features of VGG-16 and capsule network (CapsNet).

The advantages of VCNet include improved accuracy and sensitivity on large datasets, addressing variations in image characteristics, and considering spatial information of image features. The proposed framework achieves a high testing accuracy of 99.49%, outperforming other models such as MobileNet, Xception, and VGG-16. However, a potential disadvantage is the lack of detailed information on the dataset used and the specific limitations and challenges faced during implementation.

Our study, on the other hand, adopts a Machine Learning approach with preprocessing techniques of blurring images and thresholding to binarize the data, followed by training using a weighted random forest machine learning model.

The similarities between this study and our study lie in their common goal of lung nodule detection using machine learning algorithms. However, the specific approaches and techniques differ. This study utilizes a hybrid model combining VGG-16 and CapsNet, while our study employs a random forest model. The preprocessing steps in our study involve blurring and thresholding, whereas this study focuses on addressing image variations and spatial relationships of features.

G. State-of-the-art

The state-of-the-art for image-based machine learning and deep learning is constantly evolving. However, deep learning approaches are currently the leading approach for many image-related tasks. This is because deep learning approaches can learn complex features from images, which allows them to achieve state-of-the-art performance.

Some examples of state-of-the-art deep learning approaches for image-related tasks:

1) Convolutional neural networks (CNNs) are used for image classification, object detection, and segmentation.
2) Recurrent neural networks (RNNs) are used for image captioning and video understanding.
3) Generative adversarial networks (GANs) are used for image generation and style transfer.

III. DATASET AND FEATURES

A. The IQ-OTH/NCCD lung cancer dataset [4]

This dataset of lung cancer was collected over a period of 3 months in fall 2019 by the Iraq-Oncology teaching hospital/national center for cancer. It consists of CT scans of either patients who are diagnosed with lung cancer in 2 different stages (benign, and malignant) or healthy ones. Those CT scans are originally saved in DICOM format (Digital Imaging and Communication). Each scan contains several slices that show an image of human chest from different sides and angles. All cases vary in gender, age, area of residence and living status.

Here are some random samples of the 3 different classes in the data:

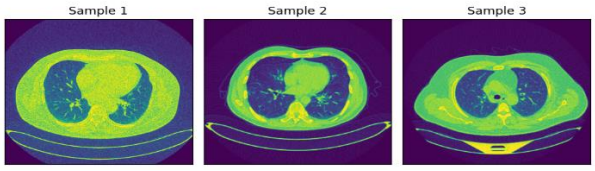

Fig 1. Benign Cases

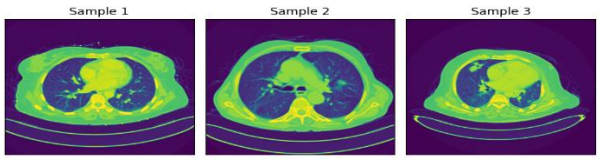

Fig 2. Malignant Cases

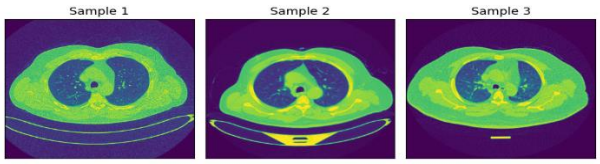

Fig 2. Normal Cases

B. *Data Size and Distribution*

The dataset contains 1097 samples labeled with one of 3 classes (benign, malignant, normal) and distributed as follows: 120 benign, 561 malignant and 416 normal.

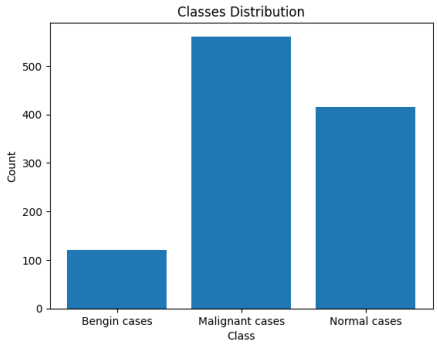

Fig 4. Classes Distribution

We will split the data later into 90% train with 5 folds cross validation and 10% test while keeping the same classes distribution in all (stratify), meaning we will have 987 samples for train and cross validation and 110 for test.

C. *Data Preprocessing*

Preprocessing is essential for our work to ensure the best representation of the key points in the data and to get the best possible accuracy. This is achieved by applying several processes on the original images data. We start by applying Gaussian blur on images to smooth them and eliminate noise. Next, Thresholding is applied, which in our case is keeping only pixel values between 120 and 255 to remove unnecessary objects and isolate the lungs from the background. Then, Edge Detection filters are applied to extract lungs borders and its infra structure outlines. Then, Dilation is applied to enlarge and thicken the boundaries obtained from the previous process. and finally, we resize all images to be in 256*256 resolution as a standard.

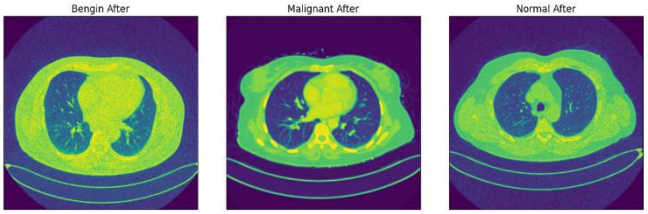

Fig 5. After Gaussian filter

Fig 6. After Thresholding

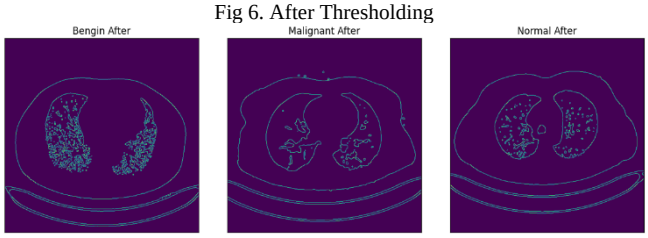

Fig 7. After Edge Detection

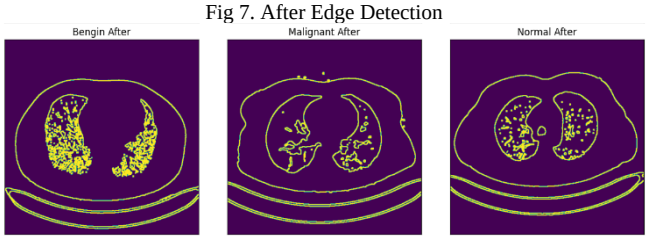

Fig 8. After Dilation

IV. METHODS

We apply 4 steps to the input images to analyze them as shown in the next block diagram.

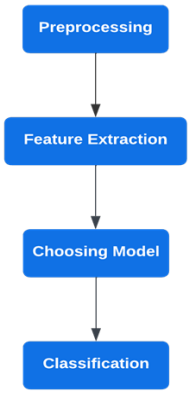

Fig 9. Analysis Steps

A. Preprocessing

As explained previously, we apply the necessary preprocessing actions (smoothing, thresholding, edge detection, dilation, resizing) on the images in order to prepare them for further steps.

B. Features Extraction

Here we get the most important features of the images to train our model on. We start by normalizing the images pixel values to be from (0 to 255), Then we get the GLCM (Gray Level Co-occurrence Matrix) of each image. The GLCM explains the spatial relationship between two adjacent pixels in an image as a function of the gray level. It computed by counting the frequency of occurrence of pairs of pixel values with a certain distance and orientation as follows [5]:

$$G(i,j,d,\theta) = \sum_i \sum_j \delta[I(x,y) = i]\, \delta[I(x + dx, y + dy) = j]$$

For preprocessed lungs CT Images distance was chosen to be 5 with 4 orientations (0, π/4,π/2,3* π/4,), where i and j are the gray levels of pixels in interest, d is the distance between the 2 pixels, θ is the angle between the pixels, (x, y) is the spatial position of the first pixel, and δ[] is a function that equals 1 if the condition inside the brackets is true, and 0 otherwise.

The GLCM is used to extract various texture features from images by applying various statistical measures to the matrix. It is widely used in image analysis, especially in the biomedical field.

The next step is to extract image features from the matrix. The features extracted are as given below:

1) Contrast: Measurement of grey-level variations between the interest pixel and its neighbors.

$$Contrast = \sum_i \sum_j (i-j)^2\, G(i,j)$$

2) Dissimilarity: Measures the number of differences or variations between the gray levels in the matrix.

$$Dissimilarity = \sum_i \sum_j |i-j|\, G(i,j)$$

3) Homogeneity: Measures the closeness of the distribution of elements in the matrix to the diagonal.

$$Homogeneity = \sum_i \sum_j \frac{G(i,j)}{1+(i-j)^2}$$

4) Energy: Gives the sum of square elements in the matrix, value of energy will be large if pixels are very similar and will be small otherwise.

$$Energy = \sum_i \sum_j G(i,j)^2$$

5) Correlation: Measures the linear dependency of the grey levels in the matrix.

$$Correlation = \sum_i \sum_j \frac{(i - \mu x)(j - \mu y) G(i,j)}{\partial x \partial y}$$

Where G is the GLCM Matrix, and (i, j) is the element of the GLCM.

Now we take the 5 chosen features for 4 orientations of each image to represent the data and train our model on them in the next steps.

C. Choosing Model

After extraction the most important features of the data, we fit those features into different classification models and tested its prediction accuracy and precision on them. We choose the SVM, KNN and Random Forest algorithms to test and fit our data into them. Then the best model algorithm was then chosen based on the best accuracy and fastest prediction time. The results of each model will be shown and explained more in the Experiments, Results and Discussion section.

D. Classification

As our final model, Random Forest was chosen. It is an ensemble learning algorithm that operates by constructing multiple decision trees and combining their outputs to make predictions. The algorithm begins by randomly selecting subsets of the training data, known as bootstrap samples, and building individual decision trees on each subset. These trees are trained using a random subset of features at each split, introducing randomness, and reducing the risk of overfitting. During prediction, each decision tree in the forest independently classifies or predicts the target variable. The final prediction is determined through a voting or averaging mechanism.

For this stage, Weighted Random Forest [6] was more suitable for the imbalance in our data as the Random Forest classifier tends normally to the majority class. with Gini impurity as criterion. Gini impurity splits as in this equation:

$$\text{Gini} = 1 - \text{n} \sum_{i=1}^{n} {p_i}^2$$

Where 'p_i' is the probability of particular class.

Weighted Random Forest is to add a cost if the classifier was biased towards majority class, which are the classes malignant and normal in our used dataset, and penalize misclassifying the minority class more severely. We give each class a weight, with the minority class receiving a heavier weight (i.e., a higher cost of misclassification) as shown in this equation:

$$\text{Class} = \max\left(\sum_{i=0}^{\#classes} w_i * count_i\right)$$

Where 'w_i' is the weight of class. '$count_i$' is class count.

Class weights are used to weigh the Gini criterion for finding splits and in the terminal nodes of each tree. The "weighted majority vote"—i.e., the weighted vote of a class is

the weight for that class times the number of cases for that class at the terminal node—determines the class prediction of each terminal node. The weighted vote from each individual tree, where the weights are average weights in the terminal nodes, is then aggregated to produce the final class prediction for the Random Forest.

V. RESULTS

For this classification problem, three models were evaluated: Random Forest, Support Vector Machine (SVM), and K-Nearest Neighbors (KNN). The dataset was split into two parts, 90% training set (5% Cross Validation) and 10% testing set. The hyperparameters for each model were tuned using grid search, The chosen hyperparameters for each model are as follows:

Table 1. Best Hyperparameters table

Random Forest	SVM	KNN
Class Weight: 0:2.5, 1:1, 2:1	C: 10.0	Number of Neighbors: 9
Max Depth: 15	Kernel: Linear	Weights: distance

The resulted parameters are used to evaluate models' performance, the accuracies are obtained as follows:

Table 2. Train Accuracy table

	Random Forest	SVM	KNN
Accuracy	83.64%	60.00%	79.09%
Mean Fit Time	0.5674 sec	18.2946 sec	0.0024 sec
Mean Score Time	0.0213 sec	0.0062 sec	0.0047 sec

As SVM accuracy is too low, it is excluded from other performance metrics used to evaluate the models. Other performance metrics are used to estimate model performance involving precision, recall and f1-score as follows:

Table 3. Metric Scores of Random Forest table

Metric	Benign	Malignant	Normal
Precision	0.58	0.89	0.83
Recall	0.87	0.86	0.81
f1-score	0.67	0.88	0.82
Test Accuracy	0.84		

Table 4. Metric Scores of KNN table

Metric	Benign	Malignant	Normal
Precision	0.58	0.89	0.71
Recall	0.70	0.82	0.77
f1-score	0.64	0.85	0.74
Test Accuracy	0.79		

The confusion matrix provides a detailed breakdown of the model's performance for each class as shown in figure:

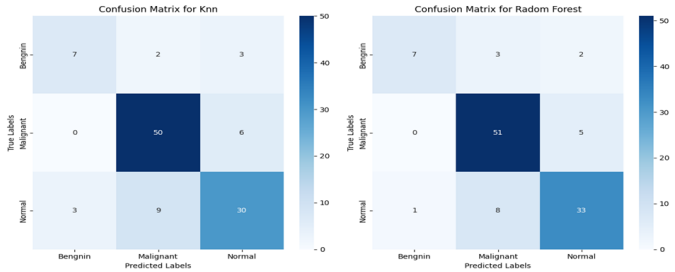

Fig 10. Confusion Matrix

The Random Forest model with the chosen hyperparameters demonstrated the highest accuracy on the test set. The difference in scores between KNN and Random Forest is not huge but Random Forest provides real-time prediction for big data trained models which makes it more applicable in real-life products such as web and mobile applications.

VI. CONCLUSION

Our study focused on using machine learning techniques to detect lung cancer from CT scan images. We compared the performance of K-nearest neighbors (KNN) and Random Forest algorithms, particularly considering the convolution matrix. Our results showed that KNN and Random Forest performed similarly, even in terms of the convolution matrix. However, it is worth noting that KNN consumed a larger amount of memory compared to Random Forest, although it exhibited shorter training times. We also found that Random Forest without weights struggled to predict the Benign class accurately, but as we trained the model with more weights, the accuracy improved.

Moving forward, we recommend two potential areas for future work. Firstly, it would be beneficial to optimize the imbalance between classes in our dataset. Imbalanced class distribution can negatively affect the performance of machine learning models, and exploring techniques to address this issue could enhance the accuracy of our lung cancer detection system. Secondly, we suggest considering the use of generative models to generate additional images of the Benign class. This approach could help balance the classes and potentially improve the overall performance of the model.

VII. References

[1] S. Nageswaran et al., “Lung Cancer Classification and Prediction Using Machine Learning and Image Processing,” BioMed Research International, vol. 2022, pp. 1–8, Aug. 2022, doi: 10.1155/2022/1755460.

[2] C. Venkatesh, K. Ramana, S. Y. Lakkisetty, S. Shamshirband, S. Agarwal, and A. Mosavi, “A Neural Network and Optimization Based Lung Cancer Detection System in CT Images,” Frontiers in Public Health, vol. 10, Jun. 2022, doi: 10.3389/fpubh.2022.769692.

[3] R. Tandon, S. Agrawal, A. Chang, and S. Shamshirband, “VCNet: Hybrid Deep Learning Model for Detection and Classification of Lung Carcinoma Using Chest Radiographs,” Frontiers in Public Health, vol. 10, Jun. 2022, doi: 10.3389/fpubh.2022.894920.

[4] H. Alyasriy and M. S. Al-Huseiny, “The IQ-OTHNCCD lung cancer dataset,” IQ-OTHNCCD, vol. 2, Jan. 2021, doi: 10.17632/bhmdr45bh2.2.

[5] S. K. PS and D. VS, “Extraction of Texture Features using GLCM and Shape Features using Connected Regions,” International Journal of Engineering and Technology, vol. 8, no. 6, pp. 2926–2930, Dec. 2016, doi: 10.21817/ijet/2016/v8i6/160806254.

[6] Using Random Forest to Learn Imbalanced Data. Chao Chen, Andy Liaw, Leo Breiman. [Online]. Available: https://statistics.berkeley.edu/sites/default/files/tech-reports/666.pdf

[7] McKinney, W. (2010). Pandas (Version 1.3.0) [software]. PyPI. https://pypi.org/project/pandas/

[8] Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., & Duchesnay, É. (2011). Scikit-learn (Version 0.24.2) [software]. PyPI. https://pypi.org/project/scikit-learn/

[9] Hunter, J. D. (2007). Matplotlib (Version 3.4.2) [software]. PyPI. https://pypi.org/project/matplotlib/

[10] Bradski, G. (2000). OpenCV (Version 4.5.2) [software]. OpenCV. https://opencv.org/

VIII. Contribution

Name	Research	Coding
Mohamed Mostafa	Abstract, Methods	Models training, Data Visualization
Romaissa Shrief	Results, Conclusion	Preprocessing
Youssef Kadry	Dataset and Features, Introduction	Results and Model Selection
Zeyad Amr	Related Work	Feature extraction