

Lung Cancer Detection Using Image Processing and Machine Learning Algorithms

Final Year Project



Advisor: **Mam Maria Tamoor**
Secondary Advisor: **Dr. Aasia Khanum**

Presented by:

Muneeb Amer	21-10579
Mohid Ali Gill	21-10842
Maham Jamil	21-10053

Department of Computer Science

Forman Christian College (A Chartered University)

Forman Christian College (A Chartered University)

Lung Cancer Detection Using Image Processing and Machine Learning Algorithms

By

NAME(S) OF PARTICIPANT(S)

Muneeb Amer

Mohid Ali Gill

Maham Jamil

Project submitted to

Department of Computer Science,

Forman Christian College (A Chartered University),

Lahore, Pakistan.

in partial fulfillment of the requirements for the degree of

**BACHELOR OF SCIENCE
IN
COMPUTER SCIENCE (Honors)**

Primary Project Advisor

Secondary Project Advisor

Senior Project Management
Committee Representative

Abstract

In this research, an artificially trained system is presented that aims to help reduce biopsies and the time it takes to detect lung cancer through normal procedures followed by doctors. Early detection of cancer can help save lives which can be achieved by the automation of the classification process. This project makes use of CT Scans as they provide very detailed images and are more likely to show lung tumors than regular chest x-rays. They can also show the size, shape and position of any lung tumors. In the first step, all the dicom images are read and pre-processed by the system. After this, the region of interest is extracted. Features such as mean, entropy, energy, contrast and homogeneity are extracted from each image in the training set. For this research, a model is trained using the k-mean clustering algorithm based on the five extracted features. One of the advantages of using k-means is that it scales to large data and easily adapts to new examples. Many challenges were faced such as reading and applying filters on dicom images. The patients' data are kept confidential by the hospitals hence it was very difficult to gather huge dataset. Evaluation is done on the basis of tumors which are correctly classified. It was observed that all the five features combined yield the highest accuracy of 94%.

Keywords: Mean, Entropy, Energy, Contrast, Homogeneity, KNN

Acknowledgement

We would like to extend our deepest gratitude to our advisors Ma'am Maria Tamoor and Dr. Aasia Khanum for their support and guidance in this project. We would also like to mention all the teachers of the Computer Science Department at FCC who have taught us over the course of the last four years and allowed us to learn, understand and enjoy the subject of Computer Science.

List of Figures

Figure 1.11: Types of lung cancer	2
Figure 2.20: Class Diagram	9
Figure 2.60: Use Case Diagram	19
Figure 3.1.1: Context Diagram.....	22
Figure 3.1.2: Activity Diagram.....	23
Figure 3.1.3: State Diagram	24
Figure 3.2.1: Sequence Diagram.....	25
Figure 3.2.2: Dataflow Diagram	26
Figure 3.3.2: Dilation of a grayscale image	27
Figure 3.3.5.1: Elements in clusters.....	30
Figure 3.3.5.2: Plot of clusters.....	30
Figure 3.4.2: Accuracy of features	31
Figure 3.4.2.1: Accuracy of model trained on mean.....	32
Figure 3.4.2.2: Accuracy of model trained on entropy.....	33
Figure 3.4.2.3: Accuracy of model trained on energy.....	34
Figure 3.4.2.4: Accuracy of model trained on contrast.....	35
Figure 3.4.2.5: Accuracy of model trained on homogeneity	36
Figure 3.4.3.1: Accuracy of final trained model	37
Figure 3.4.3.2: Screenshot of accuracy of final model.....	38
Figure 3.6.1: Swim lane diagram.....	39
Figure 3.6.2: Main screen	40
Figure 3.6.3.1: Load image screen	41
Figure 3.6.3.2 Pre-processing image screen.....	41
Figure 3.6.3.3.: Segmenting image screen	42
Figure 3.6.3.4: Feature extraction screen	42
Figure 3.6.3.5: Classifying image screen	43

List of Tables

Table 1: Comparison of different works on Object Detection.....	7
Table 2.5.1: CT scan loading use case table	12
Table 2.5.2: Pre-processing use case table	13
Table 2.5.3: Segmentation use case table	14
Table 2.5.4 Feature extraction use case table	15
Table 2.5.5: Classification use case table	17
Table 2.5.6 Clear use case table	18
Table 2.5.7: Exit use case table	18
Table 4.1: Test Case 1	44
Table 4.2: Test Case 2	45
Table 4.3: Test Case 3	46
Table 4.4: Test Case 4	47
Table 4.5: Test Case 5	48
Table 4.6: Test Case 6	49
Table 4.7: Test Case 7	50
Table 4.8: Test Case 8	51
Table 4.9: Test Case 9	52
Table 4.10: Test Case 10	53
Table 4.11: Test Case 11	54
Table 4.12: Test Case 12	55
Table 4.13: Test Case 13	56
Table 4.14: Summary of Test Cases	57

Table of Contents

ABSTRACT	I
ACKNOWLEDGEMENT	II
LIST OF FIGURES	III
LIST OF TABLES	IV
CHAPTER 1. INTRODUCTION.....	1
1.1 INTRODUCTION	1
1.2 OBJECTIVES	3
1.3 PROBLEM STATEMENT.....	4
1.4 SCOPE	4
CHAPTER 2. REQUIREMENTS ANALYSIS	5
2.1 LITERATURE REVIEW	5
2.2 USER CLASSES AND CHARACTERISTICS	8
2.3 DESIGN AND IMPLEMENTATION CONSTRAINTS.....	10
2.4 ASSUMPTIONS AND DEPENDENCIES.....	10
2.5 FUNCTIONAL REQUIREMENTS.....	10
2.5.1 CT scan loading use case.....	12
2.5.2 Pre-processing use case	13
2.5.3 Segmentation use case.....	14
2.5.4 Feature extraction use case	15
2.5.5 Classification use case.....	17
2.5.6 Clear use case	18
2.5.7 Exit use case	18
2.6 USE CASE DIAGRAM.....	19
2.7 NONFUNCTIONAL REQUIREMENTS	20
2.7.1 Performance Requirements	20
2.7.2 Safety Requirements	21
2.7.3 Security Requirements.....	21
2.7.4 Additional Software Quality Attributes.....	21
CHAPTER 3. SYSTEM DESIGN.....	22
3.1 APPLICATION AND DATA ARCHITECTURE	22
3.1.1 Context diagram	22
3.1.2 Activity diagram	23
3.1.3 State diagram.....	24
3.2 COMPONENT INTERACTIONS AND COLLABORATIONS.....	25
3.2.1 Sequence diagram.....	25
3.2.2 Data flow diagram	26
3.3 SYSTEM ARCHITECTURE.....	27
3.3.1 Pre processing	27
3.3.2 Segmentation	27
3.3.3 Feature Extraction	28
3.3.4 Classification using k-means clustering	29
3.3.5 Training Model.....	29
3.4 SYSTEM EVALUATION	31
3.4.1 K-fold cross validation	31
3.4.2 Comparison with other features.....	31
3.4.3 Accuracy of final trained model.....	37

3.5	COMPONENT-EXTERNAL ENTITIES INTERFACE	38
3.6	SCREENSHOTS/PROTOTYPE	39
3.6.1	<i>Workflow</i>	39
3.6.2	<i>Main screen</i>	40
3.6.3	<i>Software Execution</i>	41
CHAPTER 4.	TEST SPECIFICATION AND RESULTS.....	44
4.1	TEST CASE SPECIFICATION	44
4.2	SUMMARY OF TEST RESULTS	57
CHAPTER 5.	CONCLUSION AND FUTURE WORK.....	58
5.1	PROJECT SUMMARY.....	58
5.2	PROBLEMS FACED AND LESSONS LEARNED	58
5.2.1	<i>Dataset</i>	58
5.2.2	<i>Dicom Images</i>	58
5.2.3	<i>Segmentation</i>	59
5.2.4	<i>Choosing Features</i>	59
5.2.5	<i>Covid-19</i>	59
5.3	FUTURE WORK	59
5.3.1	<i>Bounding box tumor area</i>	59
5.3.2	<i>Area and depth of tumor</i>	59
5.3.3	<i>Location of tumor</i>	60
REFERENCES		61
APPENDIX A GLOSSARY		63
APPENDIX B DEPLOYMENT/INSTALLATION GUIDE		65
APPENDIX C USER MANUAL		67
APPENDIX D STUDENT INFORMATION SHEET		72
APPENDIX E PLAGIARISM FREE CERTIFICATE		73
APPENDIX F PLAGIARISM REPORT		74

Revision History

Name	Date	Reason For Changes	Version
Maham Jamil	15-3-21	Initial Documentation	1.0
Muneeb Amer	17-3-21	Changes in Introduction	1.0
Mohid Ali Gill	28-3-21	Addition in literature review	1.0
Muneeb Amer	20-4-21	Addition in Tools & Libraries Used	1.1
Maham Jamil	24-4-21	Insertion of Images	1.1
Mohid Ali Gill	01-5-21	Changes in project methodology	1.1
Maham Jamil	05-5-21	Insertion of formulas	1.2
Mohid Ali Gill	19-5-21	Addition of tables in literature review	1.2
Muneeb Amer	20-5-21	Addition to Chapter 4	1.3
Maham Jamil	20-5-21	Insertion of Captions	1.3
Muneeb Amer	22-5-21	Addition in Discussion	1.4.1
Mohid Ali Gill	23-5-21	Conclusion	1.4.2
Muneeb Amer	24-5-21	Addition in Chapter 3	1.4.3
Mohid Ali Gill	24-5-21	Diagrams added in chapter 3	1.4.4
Maham Jamil	24-5-21	Swim-lane diagram added, additions in chapter 4	1.4.5
Muneeb Amer	24-5-21	Final Changes	1.4.6
Mohid Ali Gill	28-5-21	Formatting issues fixed	1.4.7
Muneeb Amer	01-6-21	Correction in report	1.4.8
Muneeb Amer	17-06-21	Plagiarism report added	1.4.9

Chapter 1. Introduction

1.1 Introduction

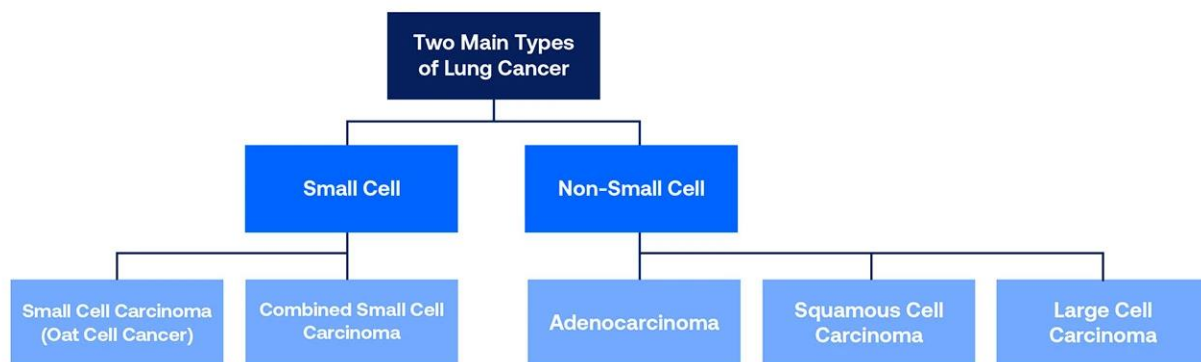
Lung cancer is one of the most common types of cancer. The cancer cells which are the abnormal cells in a human lung start to cluster together and form a tumor. The cancer cells spread without control and have no visible symptoms such as smell or pain in the initial stages. There are two basic types of tumors, **benign** tumor which is harmless as they do not spread and damage other tissues and **malignant** tumor, which is the more dangerous of the two as it spreads rapidly and eventually targets other organs as well. The spread of this tumor affects the proper functionality of the organ and these also have a chance of growing back once surgically removed which is another reason why it is more dangerous than benign tumors.

Smoking is one the most common cause of lung cancer and the risk increases is directly proportional to the number of cigarettes one smokes daily. Quitting smoking at any time can reduce the risk of tumors. The tumor, however, can also develop in people who do not smoke, as it has many other causes, such as passive smoking. Radiation plays a vital role in tumor generation. If one is constantly exposed to radiation from CT scans and X-Rays, it may damage and weaken one's organs leading to cancer. Radioactive materials like uranium when broken down in water or soil, release radon gas which is also highly poisonous for one's lungs and may cause cancer. Finally, people with a family history of lung cancer also have a risk of developing this type of cancer.

There are two main types of lung cancer and a third, very rare type of lung cancer called carcinoid. The first type of lung cancer is small cell lung cancer (SCLC), and is directly related to cigarette smoking and can be cured by chemotherapy. This type of cancer is further divided into two categories which are small cell carcinoma and combined small/large cell cancer. They get their names by their shape and size which can be seen under the

microscope. The second main type of lung cancer is non-small lung cancer (NSCLC). This is the most common lung cancer and accounts for almost 80% of all lung cancer cases [10]. The spread of this cancer is much weaker and slower than the small cell lung cancer. NSCLC is further divided into three categories which are Adenocarcinoma, Squamous cell carcinoma and large cell carcinoma. The types of lung cancer are described in **fig 1.11** below

Fig 1.11 – Types of Lung Cancer



To prevent cancer from spreading, it is vital to have the cancer cells removed in the initial stages. The process of detecting cancer is quite time-consuming as the patient has to go through many stages before the final detection of cancer. First, the patient has to go for a physical exam by a local physician where the doctor may feel areas of the body and look for abnormalities such as lumps or any organ enlargements. If the cancer is in its initial stages, it may not be detected by this method therefore some patients may get tested by laboratories. The basic tests conducted here are urine and blood tests which show the increase in white blood cells. Advanced tests like advanced genomic are far more expensive but at the same time, are more effective as they examine the body at a genetic level to detect any changes or alterations in one's DNA caused by the tumor. Once these tests are conducted and some irregularities are found, the patient is prescribed to get imaging tests such as computer tomography (CT scan), magnetic resonance imaging (MRI), and positron emission tomography (PET). These tests help in pinpointing the area where the biopsy is to be performed. The final step of detecting cancer is the biopsy which is the only procedure in

which confirmation regarding the presence of a tumor can be attained. A sample such as a tissue or a lobe is collected from the affected organ and is tested. The complete testing process from a physical exam to a biopsy is very time-consuming. Patients have to wait for appointments and test results which takes quite a bit of time; time which a cancer-positive patient does not have as the tumor is continuously spreading. On average this can take about 60-69 days i.e. 2 months [11].

Early detection of lung cancer can reduce the mortality rate by 14-20% in high-risk populations [12]. To automate medical imaging, very high accuracy is required so that no false positive or false negative is classified which may prove to be fatal for a patient. There are no projects with high accuracy available for lung cancer detection and the available ones, are not using evolutionary machine learning and artificial intelligence algorithms.

1.2 Objectives

- The project aims to automatically detect and classify tumors in lung CT scans to reduce biopsies and time taken otherwise.
- The first objective is to apply image processing techniques on the CT scan images for extracting different features and segmenting the region of interest.
- The second objective is to continuously improve pre-processing algorithms for increasing the accuracy with which the images are segmented and features are extracted.
- The third objective of this study is to train a model using the K-means clustering algorithm based on the five extracted features for the classification of tumors (whether the tumor is benign or malignant).
- The fourth objective is to compare different unsupervised learning algorithms and analyze the effectiveness of the K-means clustering algorithm.
- The fifth objective is to test the system thoroughly to improve performance and remove bugs.

1.3 Problem Statement

Biopsies are invasive, costly, time-consuming, and may lead to multiple complications such as blood loss, pain, and problems from general anesthesia. Our project aims to help avoid these biopsies and the complications that occur in 4-60% of the patients [1].

According to International Cancer Benchmarking Partnership (ICBP), delay in treatment increases the chances of cancer spreading [8]. One of the reasons for this delay is the time taken between CT scans, biopsy appointments, and lab results being released. Our project aims to reduce this time taken and in turn, increase the survival rate by detecting lung cancer with higher accuracy than other projects.

1.4 Scope

The research project aims to automate the process of lung cancer detection. To this end, an artificially trained system is developed on Matlab that takes the patients' CT scan images as an input and applies image pre-processing algorithms on it to suppress unwilling distortions and enhance some of the image features important for further processing. Next, image segmentation is applied based on the threshold values and irregularities detected to locate tumors and other boundaries in the CT scan. Five features (mean, entropy, energy, contrast, and homogeneity) are extracted from the segmented image which will be used to train the model for classifying the tumor. K-means clustering algorithm is then applied on the training set to form two clusters based on the five features. Finally, labels are generated and the tumors are classified after which the final result is displayed.

Chapter 2. Requirements Analysis

2.1 Literature Review

One project uses algorithms such as “k-means clustering, k-median clustering, particle swarm optimization, inertia-weighted particle swarm optimization, and guaranteed convergence particle swarm optimization (GCPSO)” [1] to help detect lung cancers. In this process, first, the noise from the CT scan images is removed and the image quality and visual appearance are enhanced by adaptive histogram equalization. After getting a clear image, they apply different AI techniques such as the ones mentioned above for segmentation after which detection algorithms of cancer cells are run. In this project, AI techniques were used for machine learning and we intend to use evolutionary machine learning algorithms as it will increase the accuracy and speed of cancer detection. Furthermore, we will not only be detecting the tumor but also classifying it as there are two types of tumors as discussed previously.

Another project has developed an Artificial Neural Network (ANN) to detect the absence or presence of lung cancer in the human body [2]. They, however, are taking symptoms as input variables for their Artificial Neural Network (ANN). Symptoms include fatigue, chronic disease, coughing, allergy, shortness of breath, yellow fingers, wheezing, chest pain, and difficulty swallowing. We, on the other hand, are taking only one input variable which is the CT scan, thus reducing time and unnecessary inputs.

Also, in Bandyopadhyay [3] a system is used which makes use of using Computer-Aided Diagnosis (CAD) to detect diseases from CT scan images of lungs using edge detection. Gaussian smoothing model was used to smooth the CT scan image so edge detection can become easier and the bell-shaped lung is clearer. The image in digital form is stored as an array of pixel values. The dimension of the image and its midpoint is calculated first to apply maximum differential thresholding which is done based on the change in intensity levels of the image. Homogeneity is calculated in two scans both the vertical and horizontal scans are required. The horizontal scan iterates the array row-wise from the first to the last, the

vertical scanning iterates the array column-wise. This method of calculation is much slower than the new evolutionary algorithms such as the canny edge detection algorithm.

There is also another system that is used to detect lung cancer at an early stage [4]. The system comprises of several steps such as image acquisition, pre-processing, segmentation, feature extraction, binarization, thresholding, and neural network detection. Once the CT scan image is given as an input, it is passed to the image pre-processing stage. In the first stage, the Binarization technique is used to convert the scanned image to a binary image and later, a comparison is made with the threshold value to detect cancer. In the second stage, image segmentation is used and essential features of segmented images are extracted which is further used to train the neural network. Finally, the system can test any cancerous or noncancerous image with a 92.27% accuracy rate.

Another system is being developed which uses an automatic CAD system to detect lung cancer at an early stage through analysis of the CT scan image [5]. In this project, the Hopfield Artificial Neural Network Classifier model is used for the segmentation of lung CT scans. The pictures are gathered utilizing Computed Tomography imaging techniques from a normal subject as well as of the people who have a possibility for lung cancer. In the beginning, the lung region is extracted through the scans by image processing techniques. In order to enhance the edges' detection of the lung region lobes, a combination of bit-planes of every pixel is used. Three filters are then applied to extract the true lung cancer region. The system can generate promising results and accurately differentiates between malignant and benign tumors.

This project uses CAD software to classify the tumor in lungs with unmarked nodules, a dataset from the Kaggle Data Science Bowl, 2017 [6]. For the initial segmentation of the lung tissue from the scan, thresholding is used. Initially, the segmented CT scans were to be given as input into 3D CNNs for classification, but the results were not accurate. Therefore, a modified U-Net trained on LUNA16 data was used to first detect nodule candidates in the Kaggle CT scans. This produced many false-negative results, hence the regions of interest that were segmented by the U-Net were forwarded to a 3D CNN network. The CNN classified

the data into cancer positive or negative. The 3D CNNs were able to generate an accuracy of 86.6%.

Table 1: Comparison of different works on Lung Cancer Detection

Author	Dataset	Classification Method	Accuracy
Rachid Sammouda (2016)	Data World website (user: sta427ceyin)	ANN	85.36%
Md. Badrul Alam Miah (2015)	-	CNN	92.27%
Wafaa Alakwaa (2017)	Kaggle Data Science Bowl	3D CNN	86.6%

The evolutionary algorithms are much more accurate and give a better, precise result in a fraction of the time. We will be using such algorithms in our project for the best and most accurate results of edge detection.

The main purpose of our research paper is to detect the tumor from a given CT scan image and if found, classify it as either a benign or malignant tumor all with more accuracy and time efficiency as compared to other available projects.

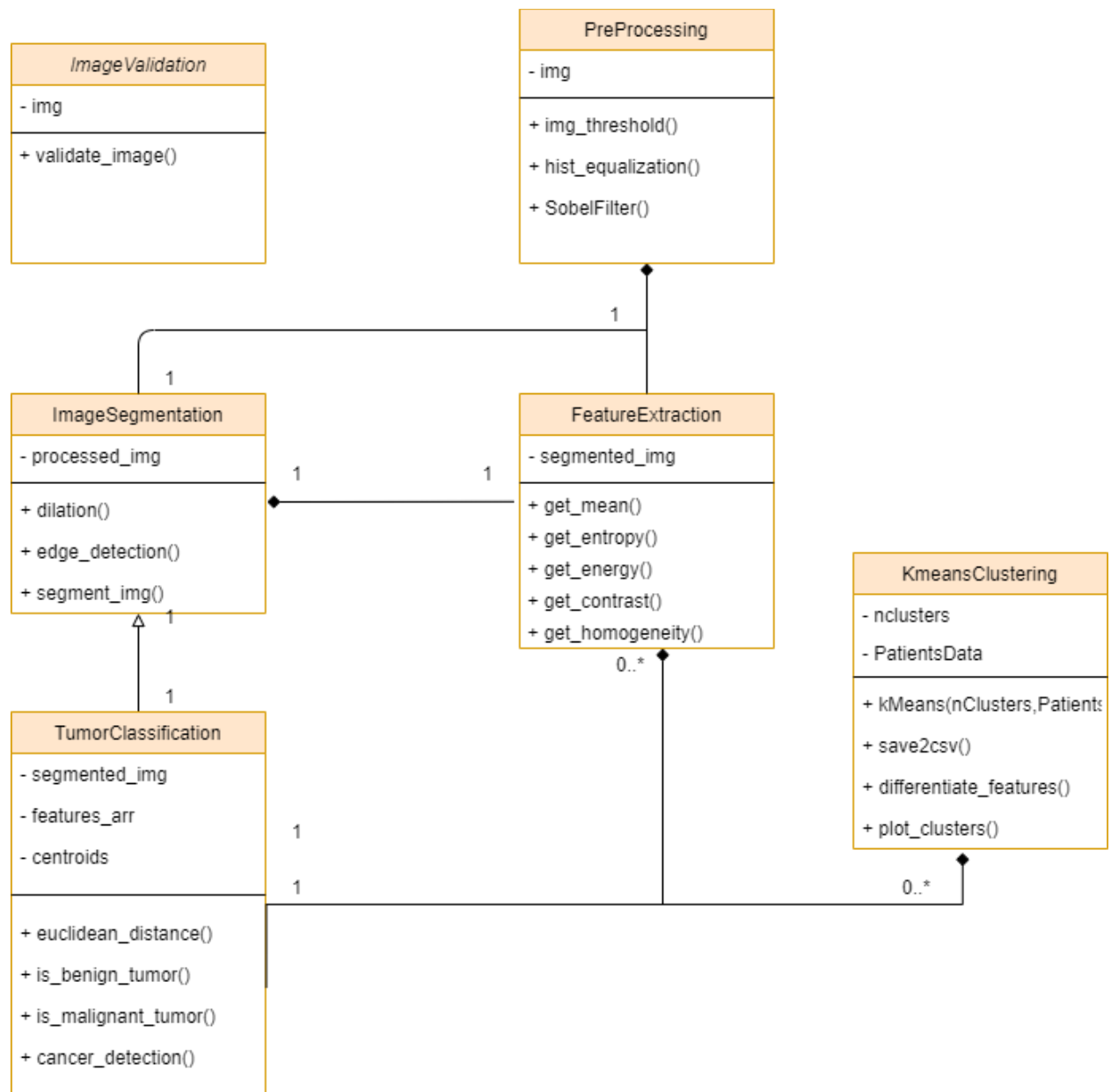
2.2 User Classes and Characteristics

The list of classes identified are as follows:

1. Image Validation – takes an image as an input and validates it if the correct image is provided. Only CT scan Dicom images are allowed.
2. Pre-processing – takes the image as an input and loops through the dicom images applying thresholding, histogram equalization, and noise removal.
3. Image segmentation – the pre-processed image is dilated first. Edge detection algorithms are then applied to detect boundaries. Finally, the image is segmented based on threshold values and irregularities found.
4. Feature Extraction – Five features; mean, entropy, energy, contrast, and homogeneity are extracted from the segmented images.
5. K-means Clustering – K-means clustering algorithm is applied on the training set to form two clusters based on the five features
6. Tumor Classification – This class checks if the tumor detected is benign or malignant based on the trained model.

The class diagram for the system is shown in **fig 2.20**

Fig 2.20 – Class Diagram



2.3 Design and Implementation Constraints

When a patient's data is stored, it raises worries about Patient Privacy and Data Security. To comply with regulations set by HIPAA, The system should provide a guarantee of keeping the patients' data confidential. This system adheres to the HIPAA Standard and the Data Protection Acts of 1998 and 2003. However, ensuring full confidentiality will cause hurdles in the doctor's work. To train the model, developers need datasets and CT scan images of both healthy lungs and cancer-positive lungs. The biggest issue is that most of the data of cancer-positive patients is kept confidential and not released by the hospitals due to which the compilation of a balanced dataset might cause a delay in development.

2.4 Assumptions and Dependencies

The program is developed on Matlab version **9.4** (R2018a). It is preferred that the program is run on version **9.4** or higher. The image processing and machine learning toolbox need to be installed on Matlab for the program to run correctly. The program also only supports the detection of cancer in lung CT scan dicom images.

2.5 Functional Requirements

- The first stage is **Image acquisition** and we will be using CT Scans as they have low noise.
- The software supports dicom images only.
- Only authorized doctors should have access to the patient's dataset.
- MATLAB Software will automatically validate if a correct image is provided before moving on to the next stage. If an incorrect image is provided, an error will be displayed preventing further processing.
- The second stage is the **Pre-Processing Stage**: The software will apply thresholding, histogram equalization, and noise filtering algorithms on the original image to enhance certain details.

- The third Stage is **Image Segmentation**: Segmentation will be done in the software to make the image more meaningful and easier to analyze. This is a crucial stage that will lead to the early diagnosis of lung cancer.
- Images will be segmented based on irregularities and threshold values.
- Thresholding should be used for segmenting the image and different edge detection algorithms experimented, to get the best results.
- The system will make use of different filters such as Sobel filter for edge detection.
- The fourth Stage is **Feature extraction**: Extraction techniques should be applied to the pre-processed and segmented image for the detection of tumors in the lungs. Five features need to be extracted from the segmented image which are mean, entropy, energy, contrast, and homogeneity.
- Unsupervised machine learning must be conducted to train the model. The algorithm which yields the best result will be chosen.
- The distribution of clusters will be as follows:
 1. **Benign or No Tumor**
 2. **Malignant Tumor**
- After Tumor classification the system should provide the following data: Different Output Images after preprocessing, segmented image, features extracted, and finally the classification result: whether cancer is detected or not.

2.5.1 CT scan loading use case

Identifier		UC-1
Purpose		Load the CT scan of the patient
Priority		Medium
Pre-conditions		CT scan images in DICOM format
Post-conditions		Displays the CT scan in the project
Typical Course of Action		
S#	Actor Action	System Response
1	The user presses the browse button	System opens the directory window
2	User navigates to the patient's folder containing the CT scan images and selects it	System opens the folder System validates the content of folder System loads the CT scan and display it
Alternate Course of Action		
S#	Actor Action	System Response
2a	User selects a folder with DICOM images not present in it	System displays an error message System prompts user to re select the folder

Table 2.5.1: CT scan loading use case table

2.5.2 Pre-processing use case

Identifier		UC-2
Purpose		Applies pre-processing techniques on the loaded CT scan
Priority		Medium
Pre-conditions		CT scan is loaded in the software (UC-1 successfully passed)
Post-conditions		Displays and returns processed CT scans
Typical Course of Action		
S#	Actor Action	System Response
1	User presses the pre-processing button	System converts the CT scan to binary System displays the binarized CT scan System applies histogram equalization on the binary CT scan System displays the equalized CT scan System smooths the equalized CT scan by applying Laplacian or Gaussian filters. System displays and returns filtered CT scan
Alternate Course of Action		
S#	Actor Action	System Response
1a	User presses the pre-processing button without loading the CT scan	System displays an error message System prompts user to load a CT scan first

Table 2.5.2: Pre-processing use case table

2.5.3 Segmentation use case

Identifier		UC-3
Purpose		Segments the pre-processed CT scan
Priority		High
Pre-conditions		A pre-processed CT scan
Post-conditions		Displays and returns segmented CT scan
Typical Course of Action		
S#	Actor Action	System Response
1	User presses the segmentation button	System applies dilation on the pre-processed CT scan System displays the dilated CT scan System applies filters for edge detection System returns and displays the segmented CT scan
Alternate Course of Action		
S#	Actor Action	System Response
1a	User presses the segmentation button without completing the previous steps	System displays an error message System prompts user to complete the previous steps

Table 2.5.3: Segmentation use case table

2.5.4 Feature extraction use case

Identifier		UC-4
Purpose		Evaluates the features from the segmented CT scan
Priority		Medium
Pre-conditions		A segmented CT scan
Post-conditions		Displays and returns the extracted features
Typical Course of Action		
S#	Actor Action	System Response
1	User presses the feature extraction button	<p>System calculates mean of the segmented CT scan</p> <p>System displays the mean of CT scan</p> <p>System calculates entropy of the segmented CT scan</p> <p>System displays the entropy of CT scan</p> <p>System calculates energy of the segmented CT scan</p> <p>System displays the energy of CT scan</p> <p>System calculates contrast of the segmented CT scan</p> <p>System displays the contrast of CT scan</p> <p>System calculates homogeneity of the segmented CT scan</p> <p>System displays the homogeneity of CT scan</p>

		System returns an array of features extracted
Alternate Course of Action		
S#	Actor Action	System Response
1a	User presses the feature extraction button without completing the previous steps	System displays an error message System prompts user to complete the previous steps

Table 2.5.4: Feature extraction use case table

2.5.5 Classification use case

Identifier	UC-5	
Purpose	Classifies the CT scan on the basis of the features extracted by applying Artificial Intelligence algorithm k means clustering. The classes are Malignant tumor (Cancer Detected) and Benign or No Tumor (Cancer Not Detected)	
Priority	High	
Pre-conditions	All features extracted from the patient’s CT scan	
Post-conditions	Displays the class of patient’s CT scan	
Typical Course of Action		
S#	Actor Action	System Response
1	User presses the classification button	System passes the features to a trained model System retrieves a label of the provided features System displays the label (Cancer or Nor Cancer)
Alternate Course of Action		
S#	Actor Action	System Response
1a	User presses the classification button without completing the previous steps	System displays an error message System prompts user to complete the previous steps

Table 2.5.5: Classification use case table

2.5.6 Clear use case

Identifier		UC-6
Purpose		Resets the software. Clears all completed steps
Priority		Low
Pre-conditions		Software running
Post-conditions		Clears the display and wash the set variables
Typical Course of Action		
S#	Actor Action	System Response
1	User presses the clear button	System clears all the figures in GUI System resets all the changed variables

Table 2.5.6: Clear use case table

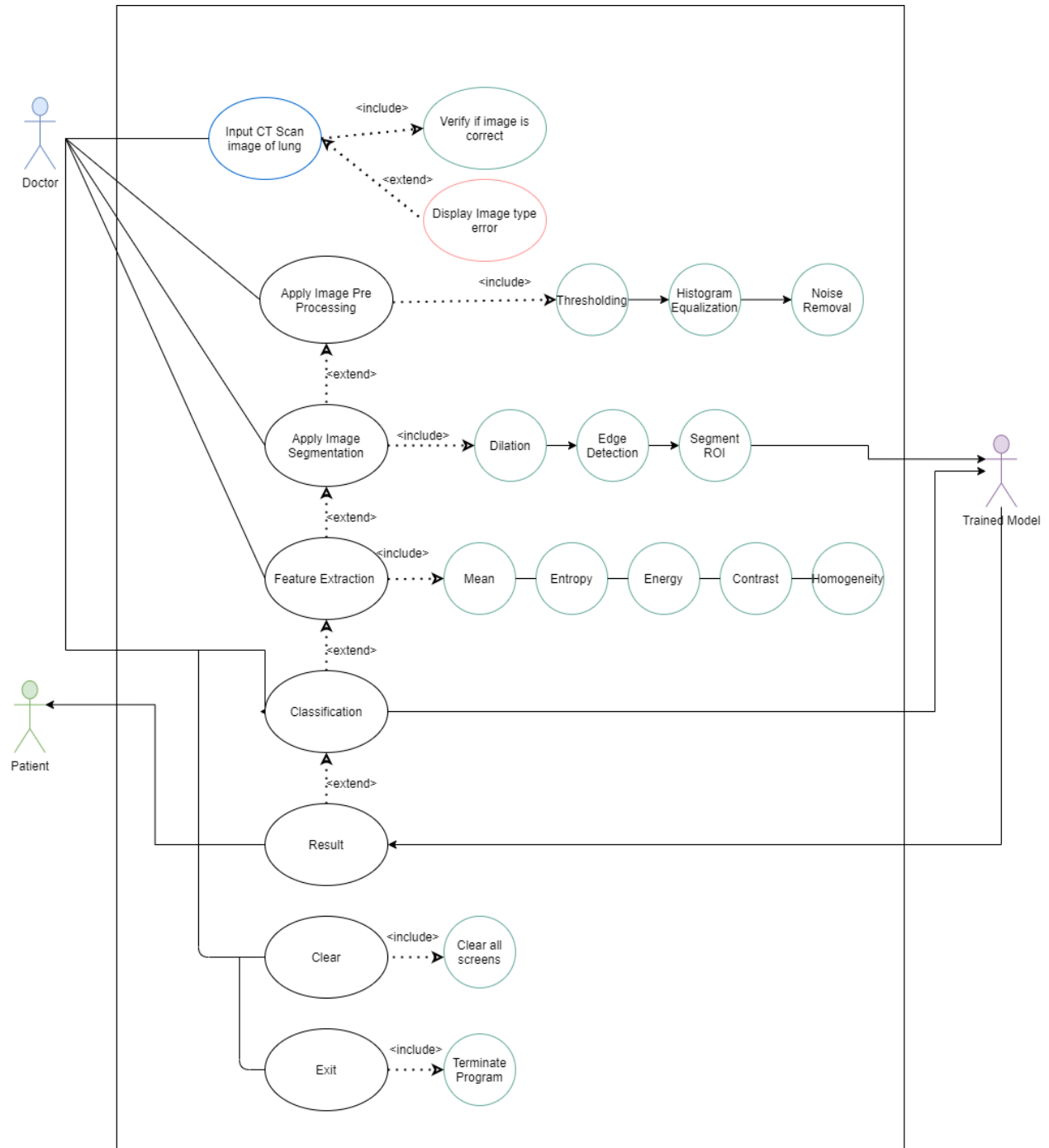
2.5.7 Exit use case

Identifier		UC-7
Purpose		Exits the software
Priority		Low
Pre-conditions		Software running
Post-conditions		Closes the software window
Typical Course of Action		
S#	Actor Action	System Response
1	User presses the exit button	System exits and closes the GUI window

Table 2.5.7: Exit use case table

2.6 Use Case Diagram

Fig 2.60 – Use Case Diagram



2.7 Nonfunctional Requirements

1. **Completeness:** All features should be working
2. **Time behavior:** The system should give the desired output in minimum time.
3. The system shall be lightweight so that it can also run on computers with low specifications.
4. **Accessibility:** The system should run without the need for an internet connection.
5. **User error protection:** The system should be very accurate leaving a very low margin for errors.
6. **Operability:** The system should be easy to learn, manageable, and user-friendly. A user manual should be available with the software.
7. The system should be fault-tolerant i.e. it should not crash if any incorrect input is given
8. The system should be recoverable in case of any crash.
9. **24/7** support should be available for the system in case anything goes wrong.
10. **Integrity:** The system should keep the information of the patient confidential.
11. The functions of a system should be written such that they can be reused for adding more features in future
12. The system should give at least **90%** accuracy

2.7.1 Performance Requirements

Our system accuracy should be at least 90%. The accuracy will be calculated using the k fold cross-validation method which evaluates machine learning models on a limited data sample.

The dataset is shuffled randomly and split into k groups.

For each unique group:

1. The testing dataset is taken
2. The remaining groups are training datasets
3. The model is fit on the training set and evaluated on the testing set
4. The evaluation scores are retained

We have taken **k = 10** which gives the most unbiased results.

2.7.2 Safety Requirements

As false positives and false negatives are possible, the doctor should perform some other tests to confirm the condition as it can be fatal for the patient.

2.7.3 Security Requirements

The network connection to the system should be private and restricted such that only the doctor and the patient should have access to the CT scans datasets and the result so the data cannot be misused.

The software can be vulnerable to viruses and hacking so internet connectivity should be avoided to minimize the risk.

2.7.4 Additional Software Quality Attributes

Although a workshop will be conducted to educate the users about the functionality of the system and how to operate it, yet we will try our best to make the system as user-friendly and easily adaptable as possible.

Moreover, 24/7 online assistance will be available to assist the users. Furthermore, the system will be regularly updated incorporating the latest features to our system making the user experience even better. The system will be thoroughly tested by our test team before handing over and validation checks will be applied. The test team will also test the system's robustness by the method of fault injection.

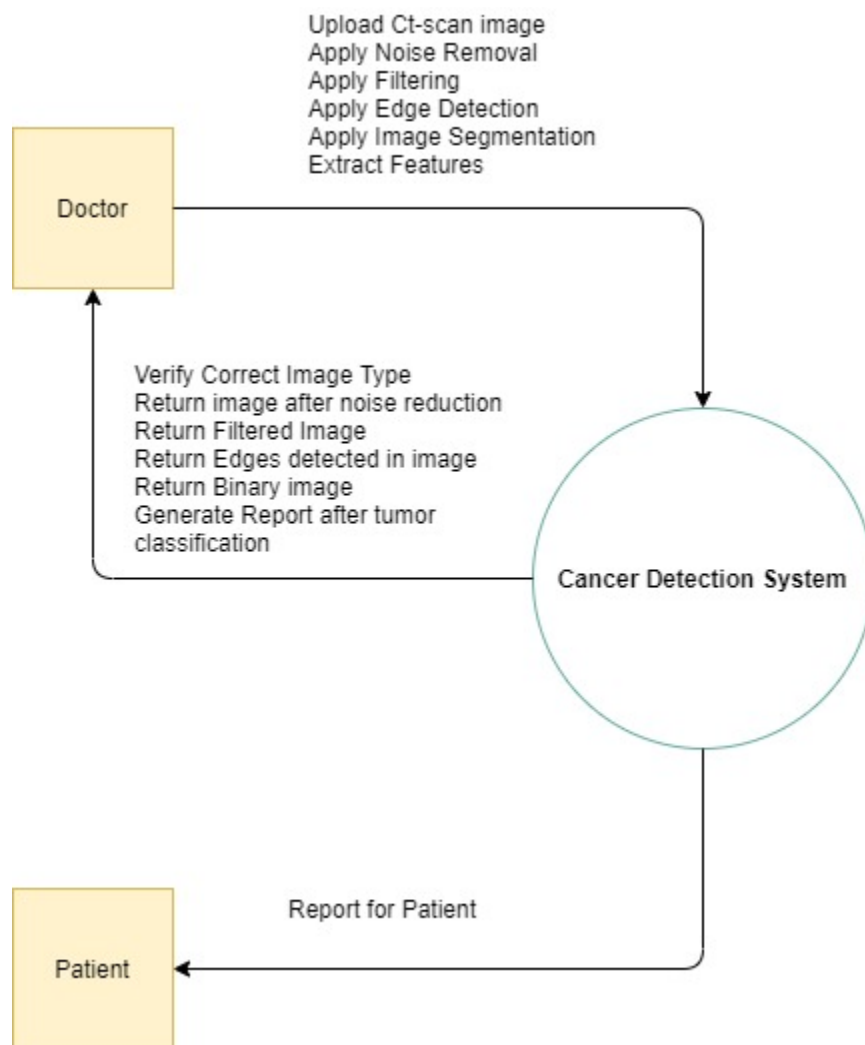
Chapter 3. System Design

3.1 Application and Data Architecture

3.1.1 Context diagram

In the context diagram **fig 3.1.1**, the entire lung cancer detection software is shown as a single process.

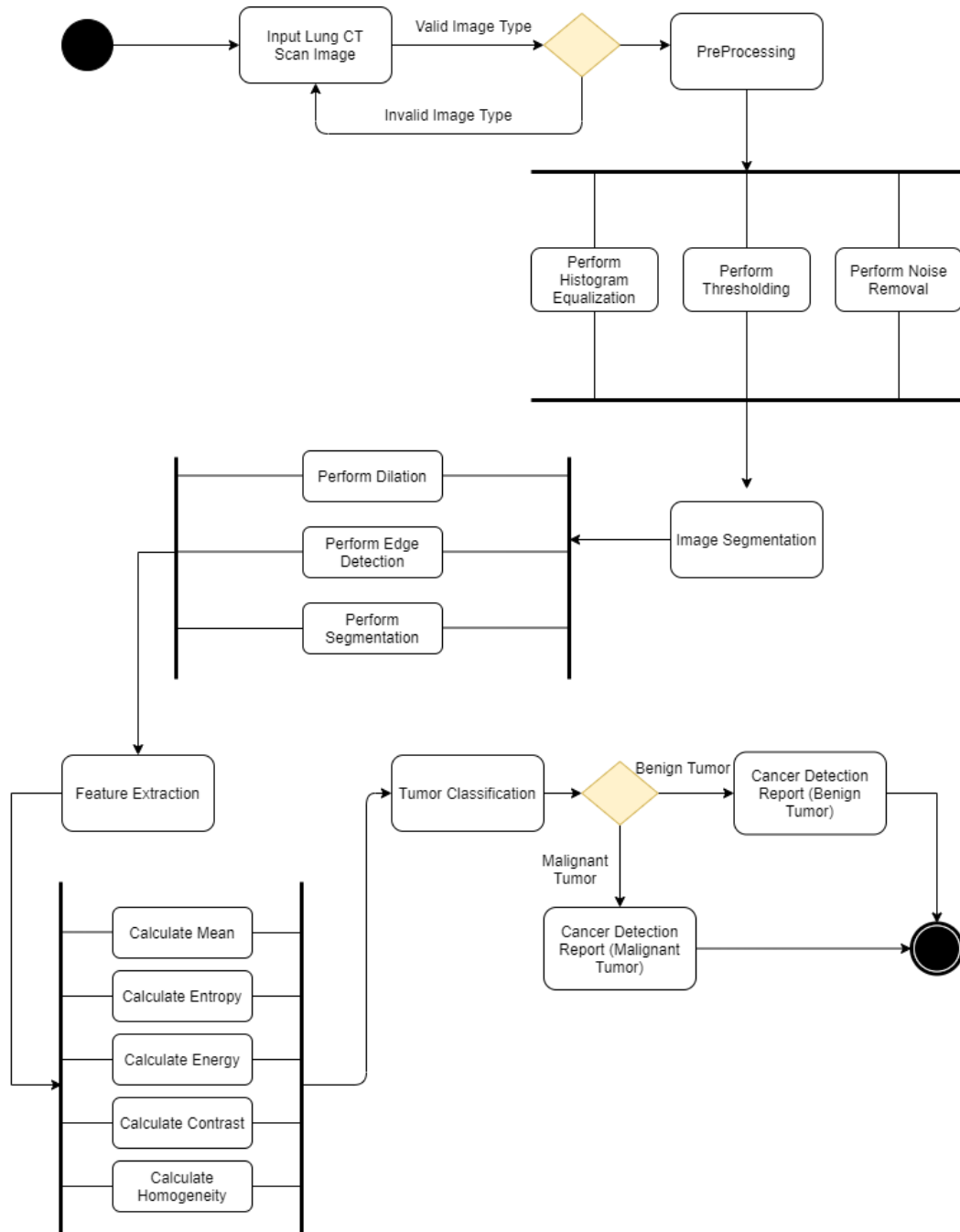
Fig 3.1.1 – Context Diagram



3.1.2 Activity diagram

The activity diagram shown in **fig 3.1.2** shows the dynamic aspects of the lung cancer detection system. All the activities performed by the system at different stages can be seen clearly in the figure.

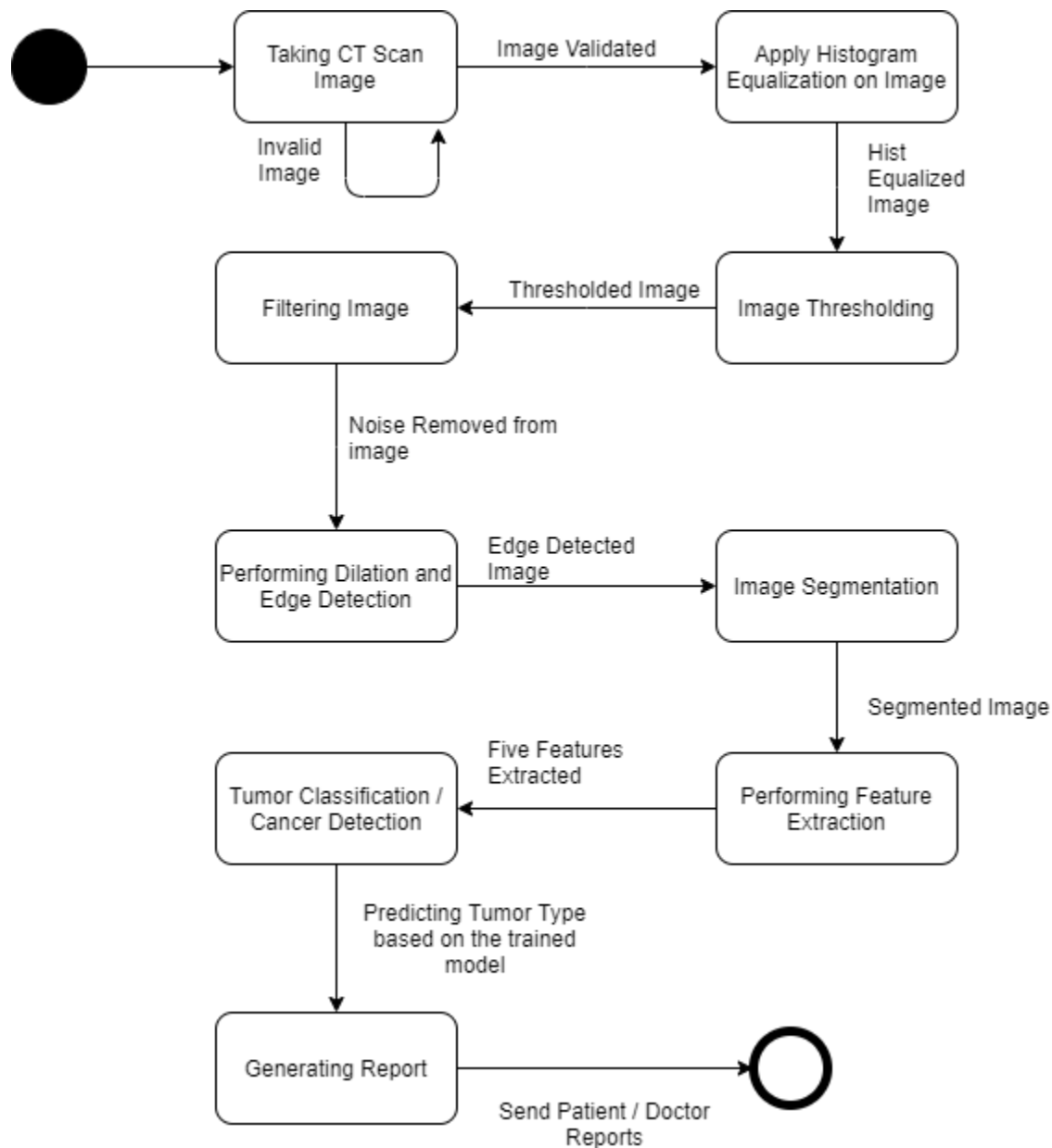
Fig 3.1.2 – Activity Diagram



3.1.3 State diagram

Each state of the system is clearly shown in the state diagram in **fig 3.1.3**

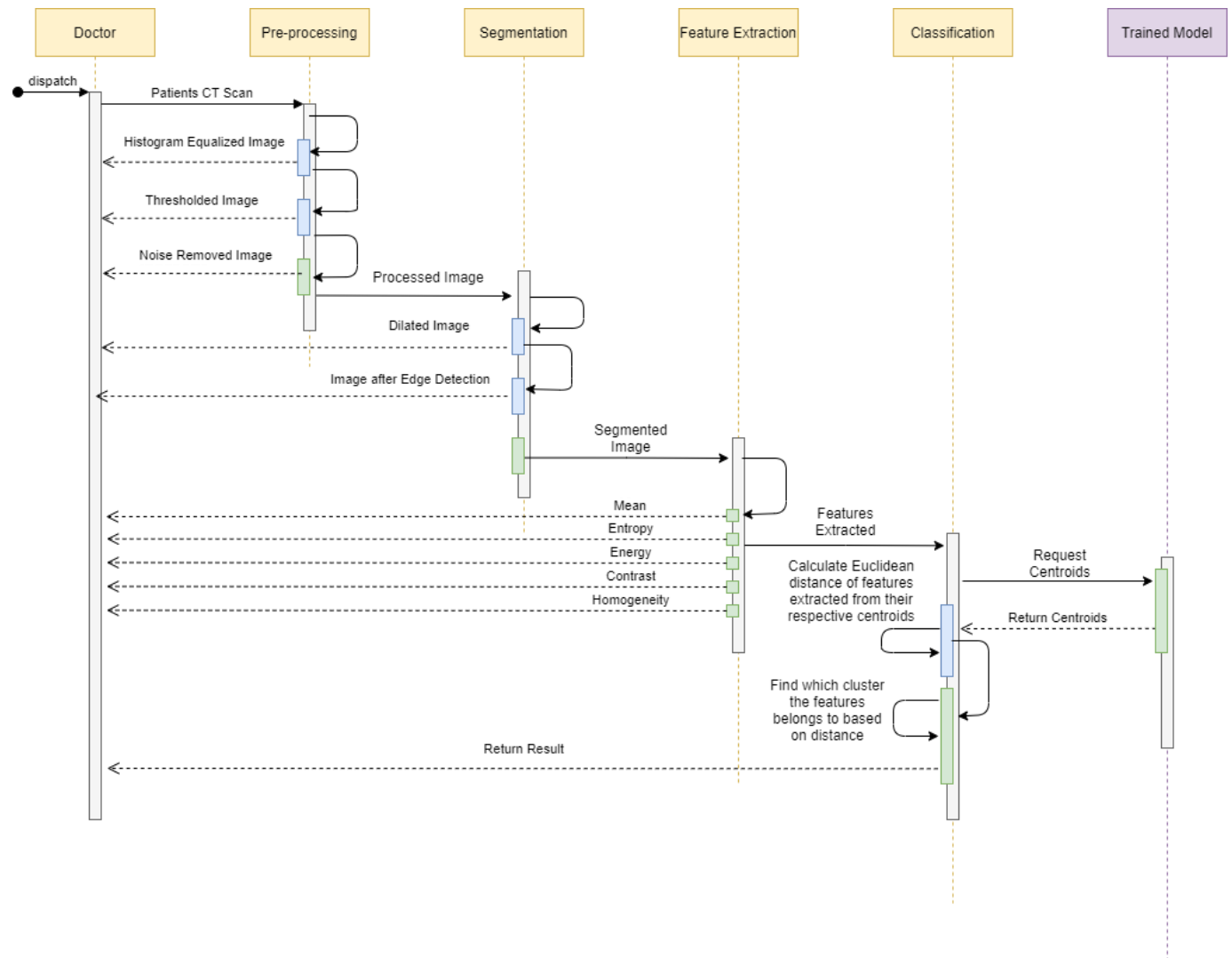
Fig 3.1.3 – State Diagram



3.2 Component Interactions and Collaborations

3.2.1 Sequence diagram

Fig 3.2.1 – Sequence Diagram

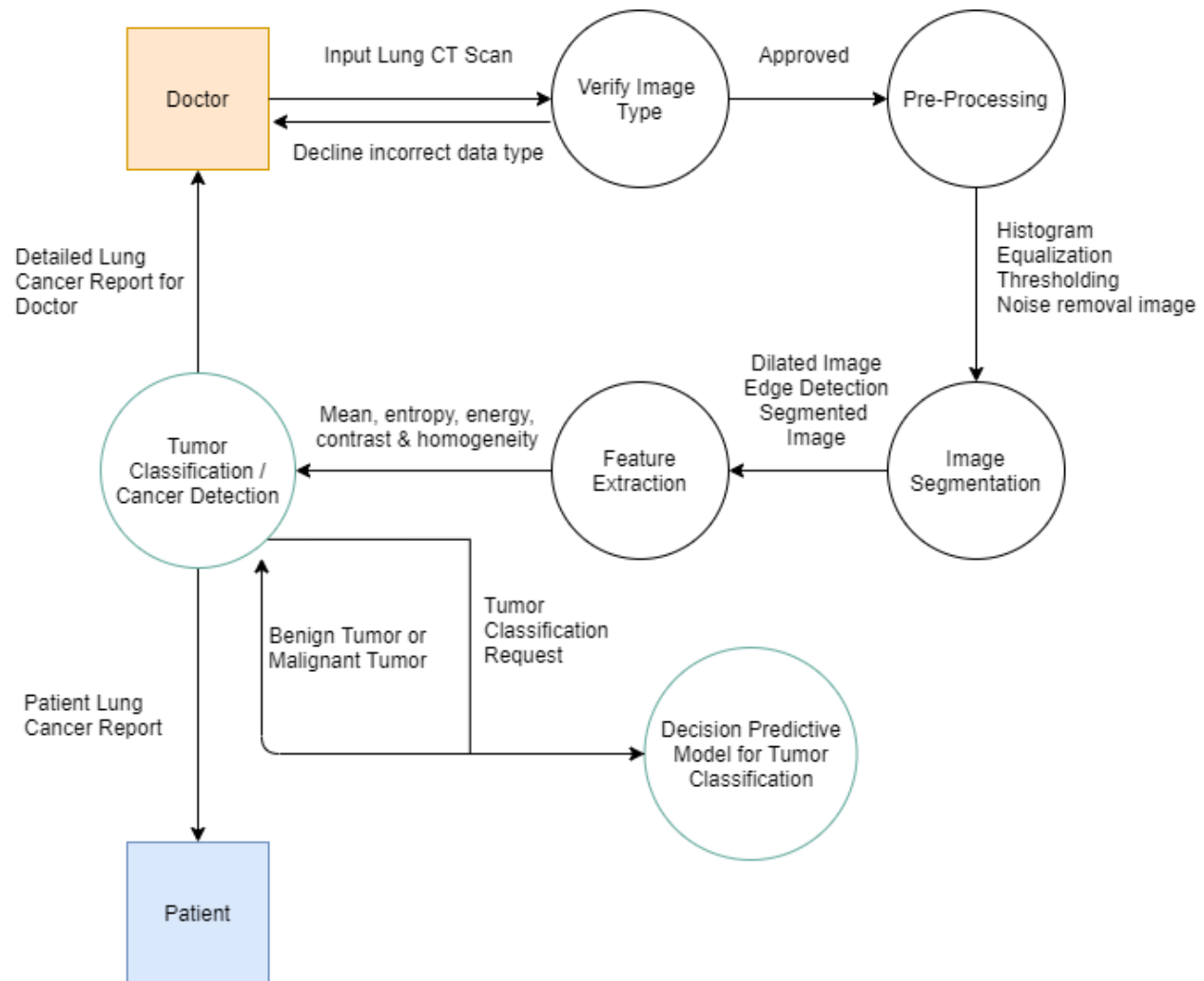


The sequence diagram in **fig 3.2.1** shows how the different components of the lung cancer detection system work together and interact with each other. The proper flow of the system can be seen in depth in the diagram. It is important to note that in the classification stage the classifier takes the features extracted from the segmented image and sends a request over to the trained model for the centroids of features. The model is trained using K-means clustering and two clusters are generated based on the features of our training set. The training set includes **80%** of patient's data from the original dataset. After the classifier

receives the centroids from the trained model, it then calculates the Euclidean distance of segmented image features from their respective centroids. Finally, it classifies the tumor type based on the cluster it belongs to and returns the result to the doctor.

3.2.2 Data flow diagram

Fig 3.2.2– Data Flow Diagram



The exchange of information and data between all the components of the lung cancer detection system is shown in the data flow diagram in **fig 3.2.2**

3.3 System Architecture

3.3.1 Preprocessing

The Preprocessing function is responsible for taking the CT scan image as an input and applying different filters to it to enhance some features which are important for further processing. First Histogram equalization is performed on the image to adjust the contrast of the image.

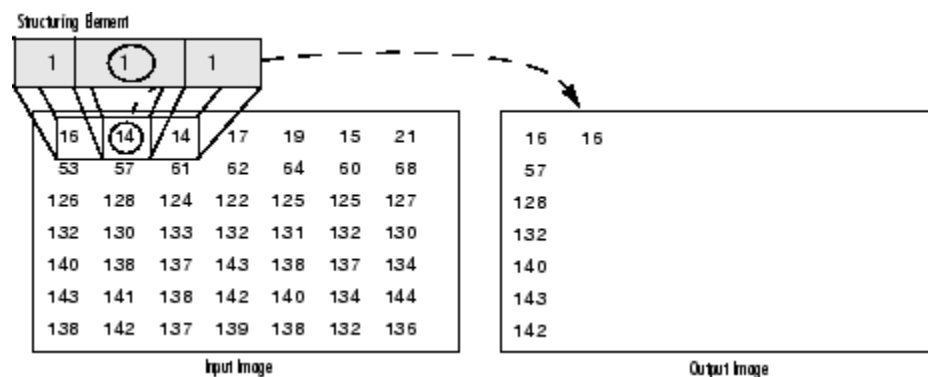
Next global image thresholding is done using Otsu's method.

Lastly in the pre-processing stage, the Laplacian filter is used to sharpen the image and reduce noise. The final processed image is then passed on to the segmentation function.

3.3.2 Segmentation

The segmentation function takes the processed image as an input and first applies dilation on the image which is used to add pixels to the boundaries of the image. The structuring element moves across the image and the highest value of all the pixels in the neighborhood is applied to the output image. A visual representation is shown in **fig 3.3.2**.

Fig 3.3.2- Dilation on a grayscale image (Source: mathworks.com)



Next edge detection is performed on the dilated image to extract the structures of objects in the image as we need to find irregularities. Malignant tumors have an irregular shape. Finally, image segmentation is performed based on the irregularities found in the image after edge detection based on the trained model. The final segmented image is then passed on for feature extraction.

3.3.3 Feature Extraction

Here we are extracting five features from the segmented image.

1. Mean

$$\overline{X} = \frac{\sum X}{n}$$

2. Entropy

$$-\sum_{i=1}^K \sum_{j=1}^K p_{ij} \log_2 p_{ij}$$

3. Energy

$$\sum_{i=1}^K \sum_{j=1}^K p_{ij}^2$$

4. Contrast

$$\sum_{i=1}^K \sum_{j=1}^K (i - j)^2 p_{ij}$$

5. Homogeneity

$$\sum_{i=1}^K \sum_{j=1}^K \frac{p_{ij}}{1 + |i - j|}$$

These features are then passed on to the classification function.

3.3.4 Classification using k-means clustering

Once the model was trained, the features of the CT scan are passed to the classifier. The classifier calculates the Euclidian distance of the features to both centroids and returns a label for it. Finally, the label i.e. cancer or not cancer is displayed on the screen.

3.3.5 Training Model

The training dataset is loaded and all the steps required before the phase of classification (pre-processing, segmentation, and feature extraction) are applied to all the patients. The features that are extracted for each patient are stored in a CSV file for the use of training a model.

K means clustering was used to train the model. Two 5d points were created randomly in the range of the features extracted and were set as the initial centroids for the learning data. The file containing the features of training data was read and the Euclidian distance was calculated of each patient to both centroids. The patient belonged to the centroid cluster to whom the distance was shorter. Once all the patients had a cluster assigned, the mean of clusters was calculated and these were the new centroids. The process of assigning clusters was iterated 100 times or till the value of the centroid stopped changing. A CSV file was created after the completion of the learning process, labeling the learned data and assigning them to a cluster. The number of patients in each cluster were displayed as shown in the **fig**

3.3.5.1

Fig 3.3.5.1 – Elements in clusters

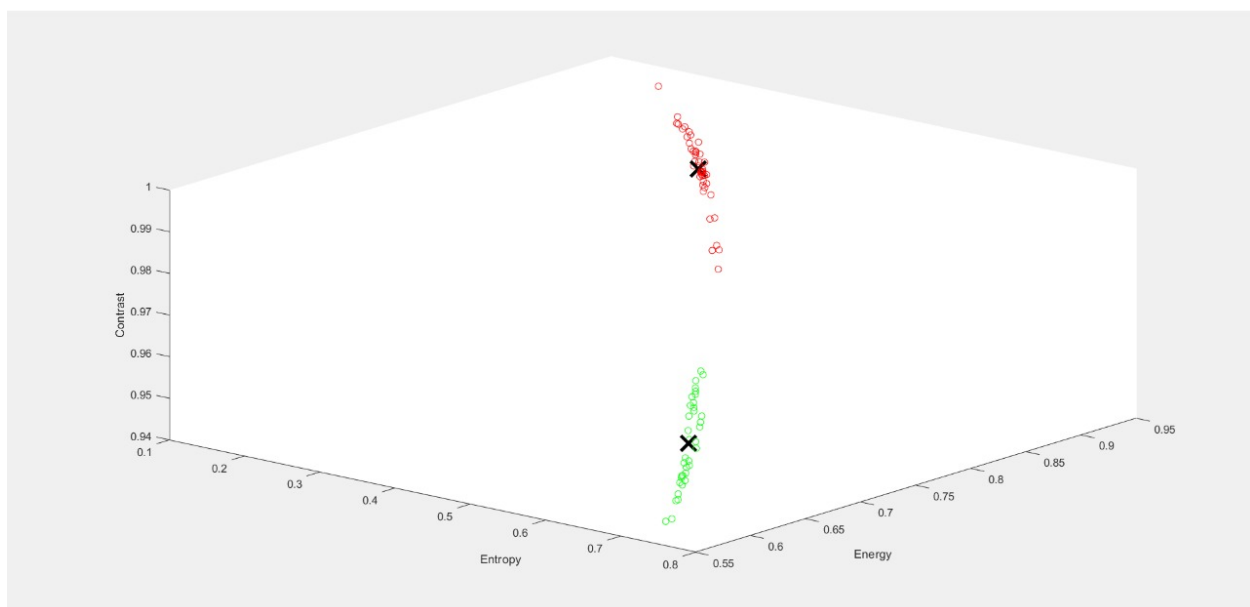
```
LCD.m x kmeansClustering.m x featuresCSV.m x +
1 - close all; clear; clc;
2
3
4 %% Input dataset and number of clusters
5 - k = 2; % number of clusters
```

Command Window

```
Dataset of 80 points will be divided in 2 clusters.
TRAINING...
Values in cluster 1 are: 43.
Values in cluster 2 are: 37.
FILES CREATED
fx >>
```

The clusters were plotted to check if they were meaningful and unique. The data was 5d as there were five features, so it could not be plotted in Matlab or any other environment. The three most important features were chosen and the clusters were plotted in 3d. The features chosen for the plotting were entropy, energy, and contrast. The plot can be seen in **fig 3.3.5.2**

Fig 3.3.5.2 –plot of clusters



3.4 System Evaluation

3.4.1 K-fold cross-validation

The dataset was divided randomly into 10 equal portions. 1 portion was used for testing and the other 9 were used for training. This process was repeated 10 times, changing the testing dataset each time. After the completion of each iteration, the testing data was run on the trained model. The generated labels were compared with the original labels and percentage accuracy was calculated.

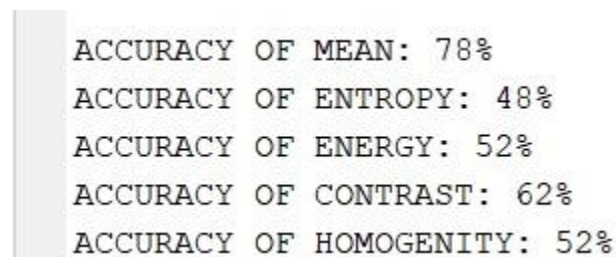
Once all the 10 iterations were completed the average accuracy of the iterations was selected as the final accuracy of the software.

3.4.2 Comparison with other features

The software uses five features to train and test the model. To check how better the selected features were when combined, five models were trained initially. For each feature individually, its accuracy was calculated using the k-fold cross-validation method.

The average accuracies returned by the Matlab software for the individually trained models are as shown in **fig 3.4.2**

Fig 3.4.2 – Accuracy of features



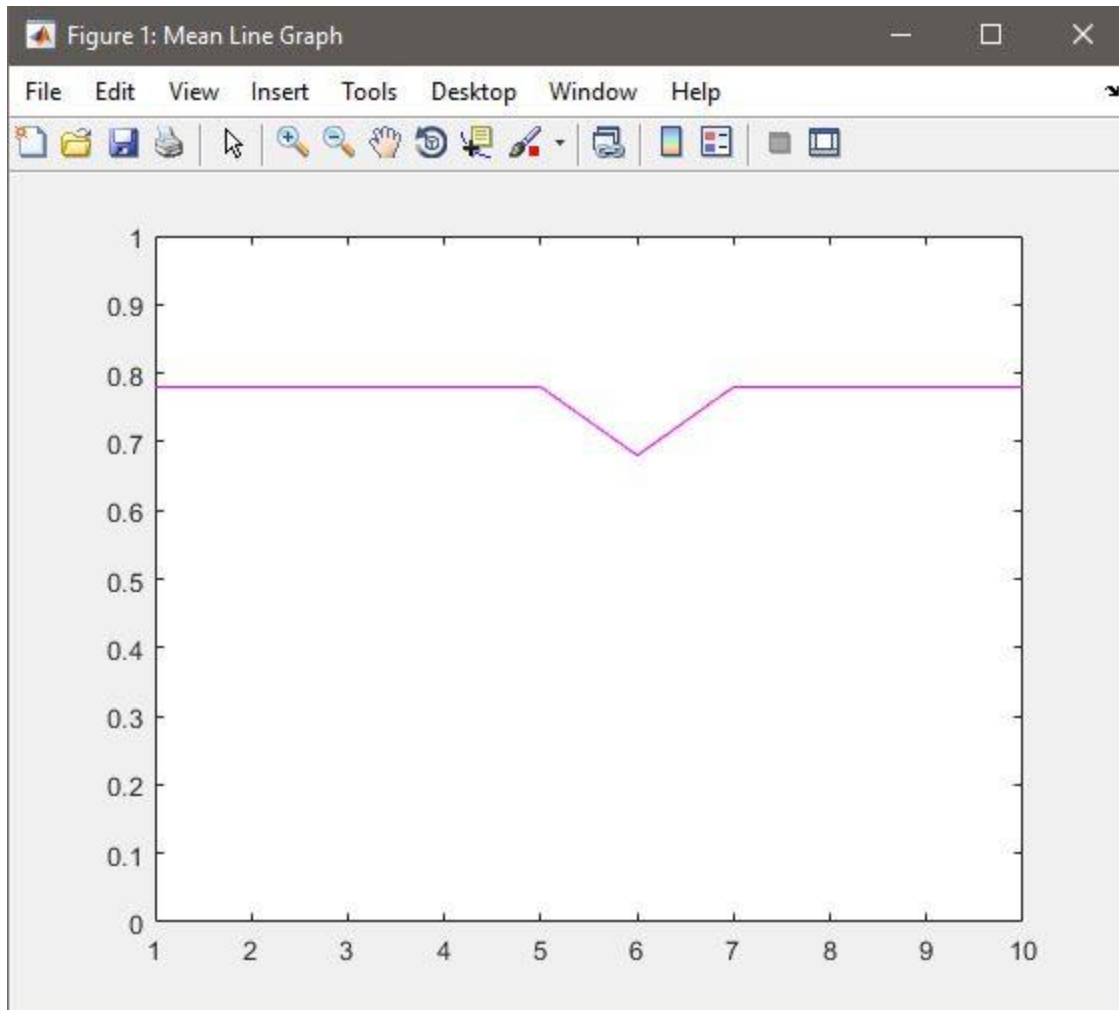
ACCURACY OF MEAN:	78%
ACCURACY OF ENTROPY:	48%
ACCURACY OF ENERGY:	52%
ACCURACY OF CONTRAST:	62%
ACCURACY OF HOMOGENITY:	52%

The accuracy for each model is plotted in the line charts below.

3.4.2.1 Accuracy of the model trained on mean

Fig 3.4.2.1 shows the accuracy when the model was trained on the mean of the segmented image. The X-axis represents the iteration of the k fold and Y-axis represents the accuracy calculated for that iteration.

Fig 3.4.2.1 – Accuracy of model trained on mean

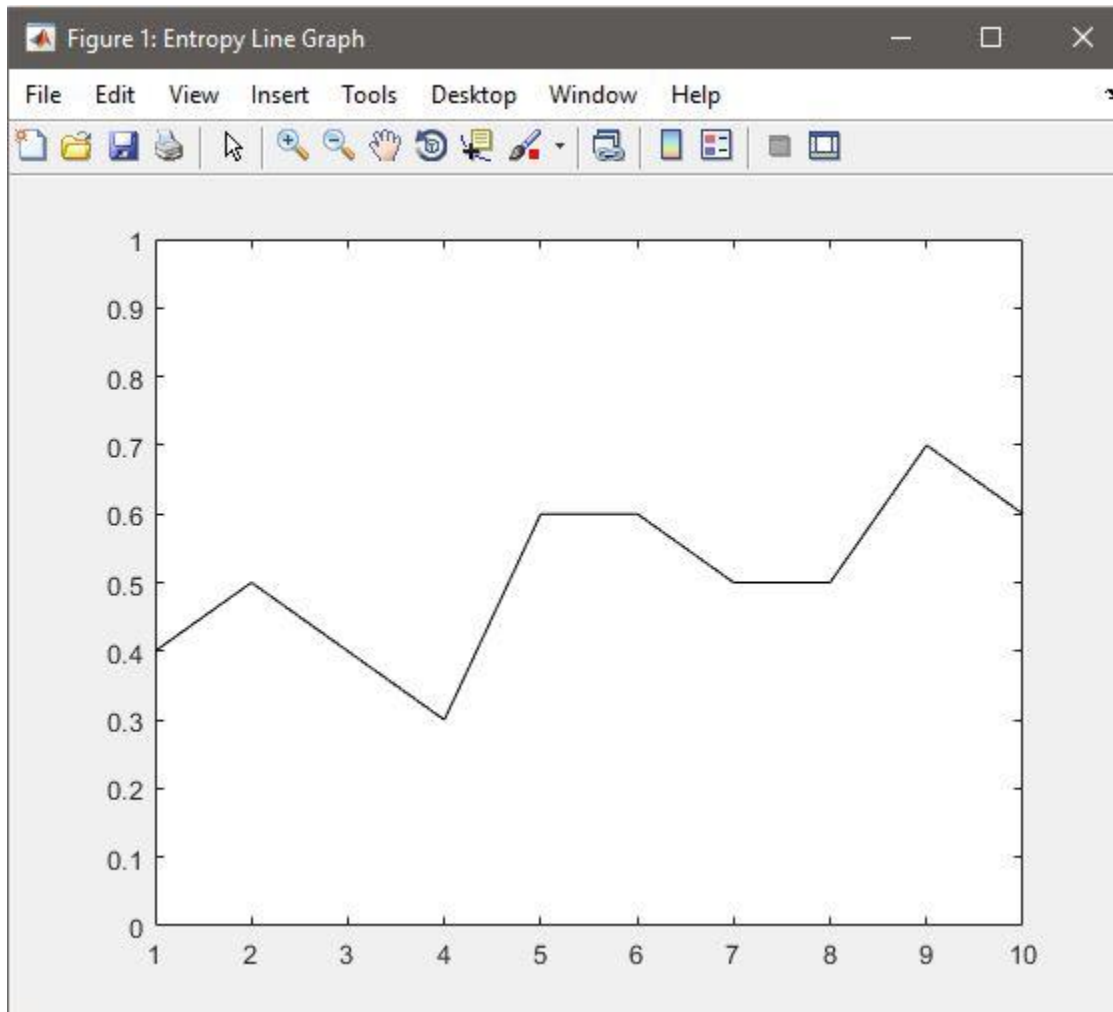


The average accuracy of mean was 78%

3.4.2.2 Accuracy of model trained on entropy

Fig 3.4.2.2 shows the accuracy when the model was trained on the entropy of the segmented image. The X-axis represents the iteration of the k fold and Y-axis represents the accuracy calculated for that iteration.

Fig 3.4.2.2 – Accuracy of model trained on entropy

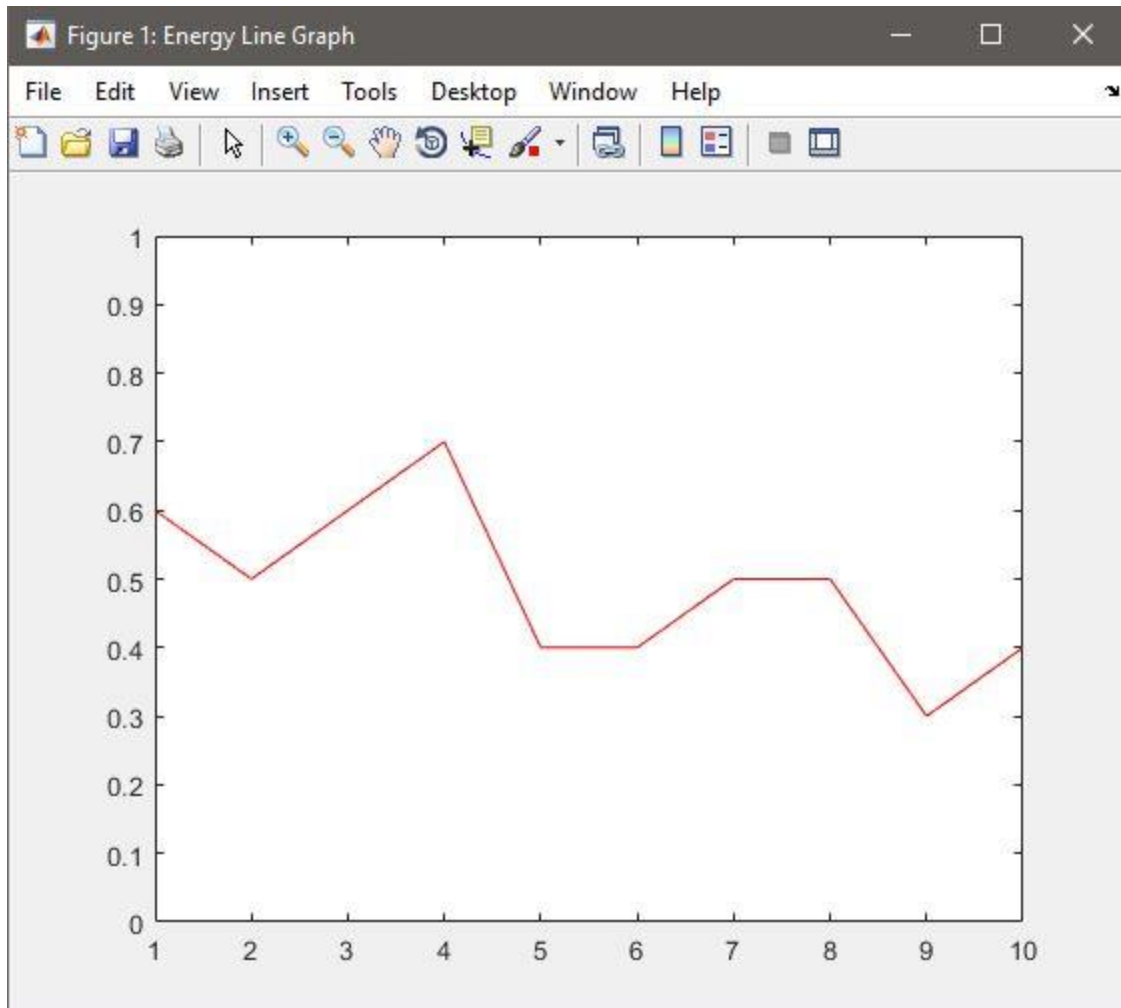


The average accuracy of entropy was 48%

3.4.2.3 Accuracy of model trained on energy

Fig 3.4.2.3 shows the accuracy when the model was trained on the energy of the segmented image. The X-axis represents the iteration of the k fold and Y-axis represents the accuracy calculated for that iteration.

Fig 3.4.2.3– Accuracy of model trained on energy

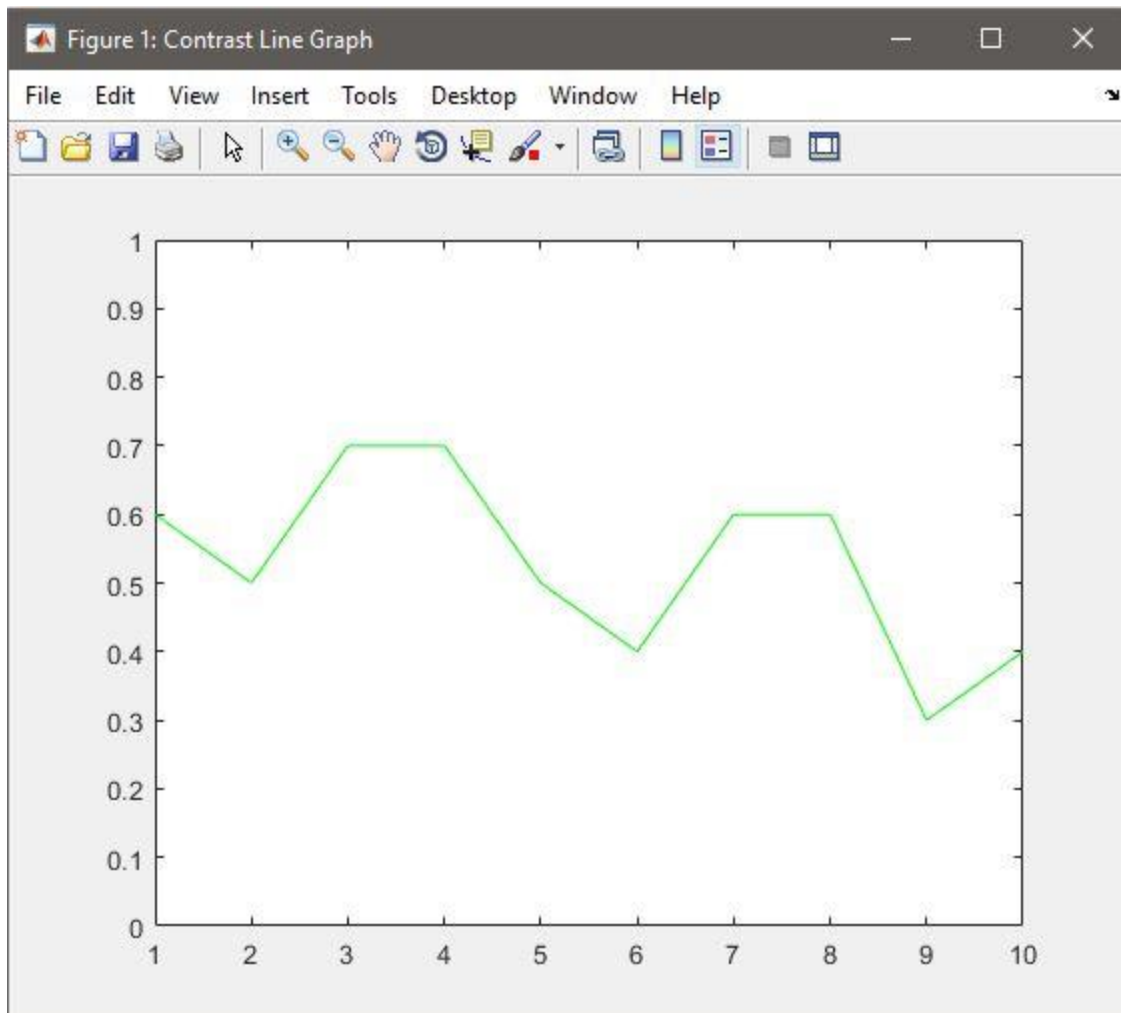


The average accuracy of energy was 52%

3.4.2.4 Accuracy of model trained on contrast

Fig 3.4.2.4 shows the accuracy when the model was trained on the contrast of the segmented image. The X-axis represents the iteration of the k fold and Y-axis represents the accuracy calculated for that iteration.

Fig 3.4.2.4 – Accuracy of model trained on contrast

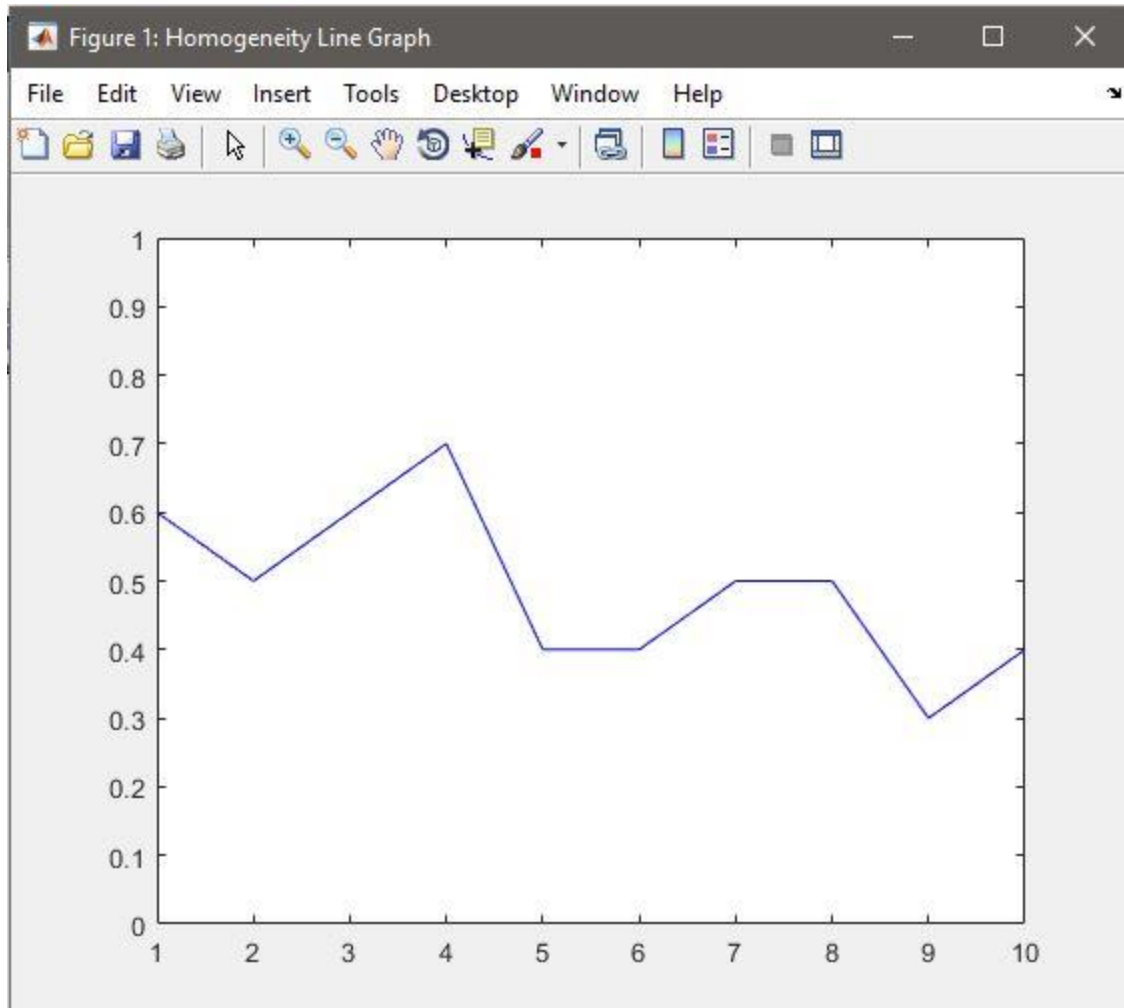


The average accuracy of contrast was 62%

3.4.2.5 Accuracy of model trained on homogeneity

Fig 3.4.2.5 shows the accuracy when the model was trained on the homogeneity of the segmented image. The X-axis represents the iteration of the k fold and Y-axis represents the accuracy calculated for that iteration.

Fig 3.4.2.5 – Accuracy of model trained on homogeneity



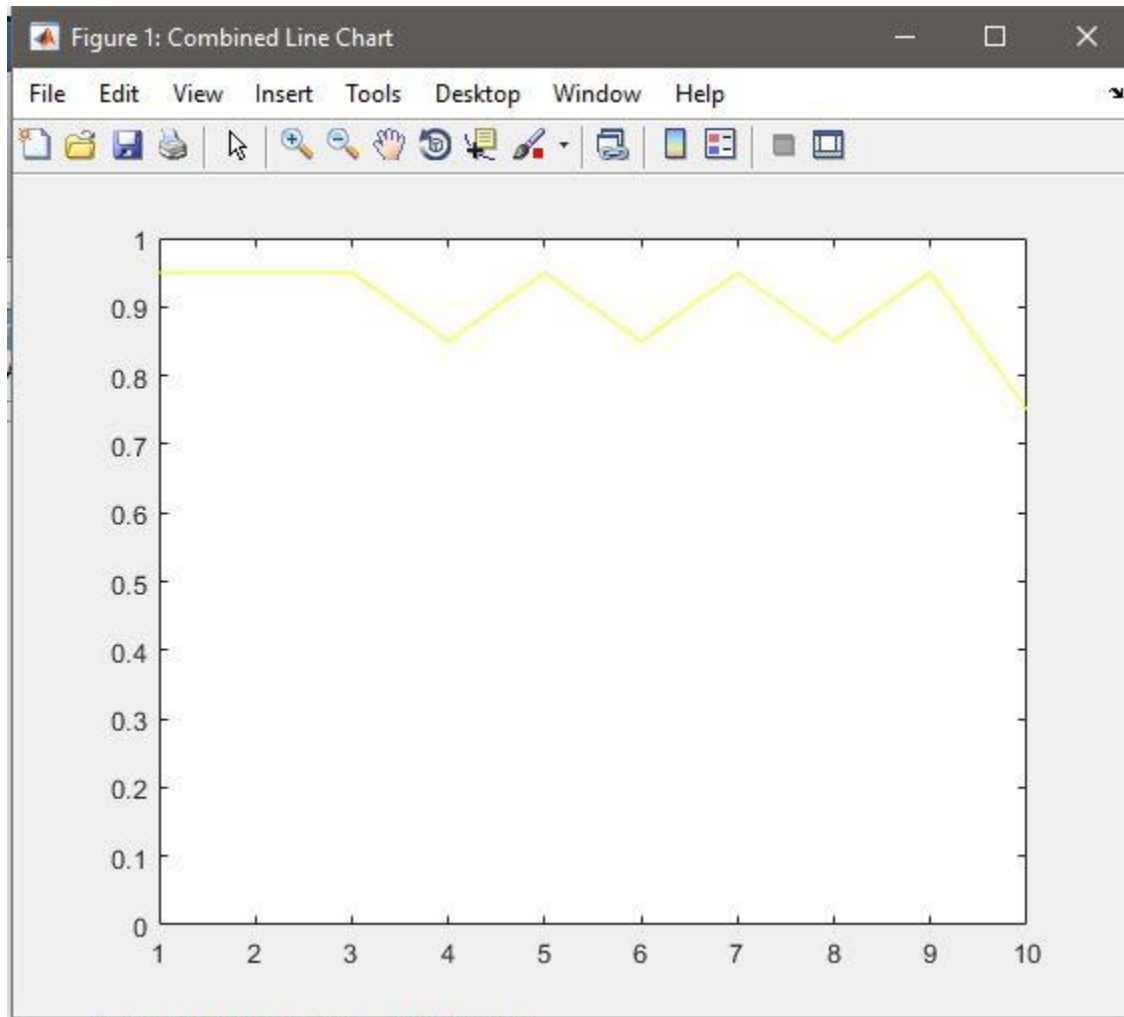
The average accuracy of homogeneity was 52%

3.4.3 Accuracy of the final trained model

The final model was trained using all the features above. It was tested by k-fold cross-validation technique, the accuracy was plotted as a line chart and compared with other accuracies

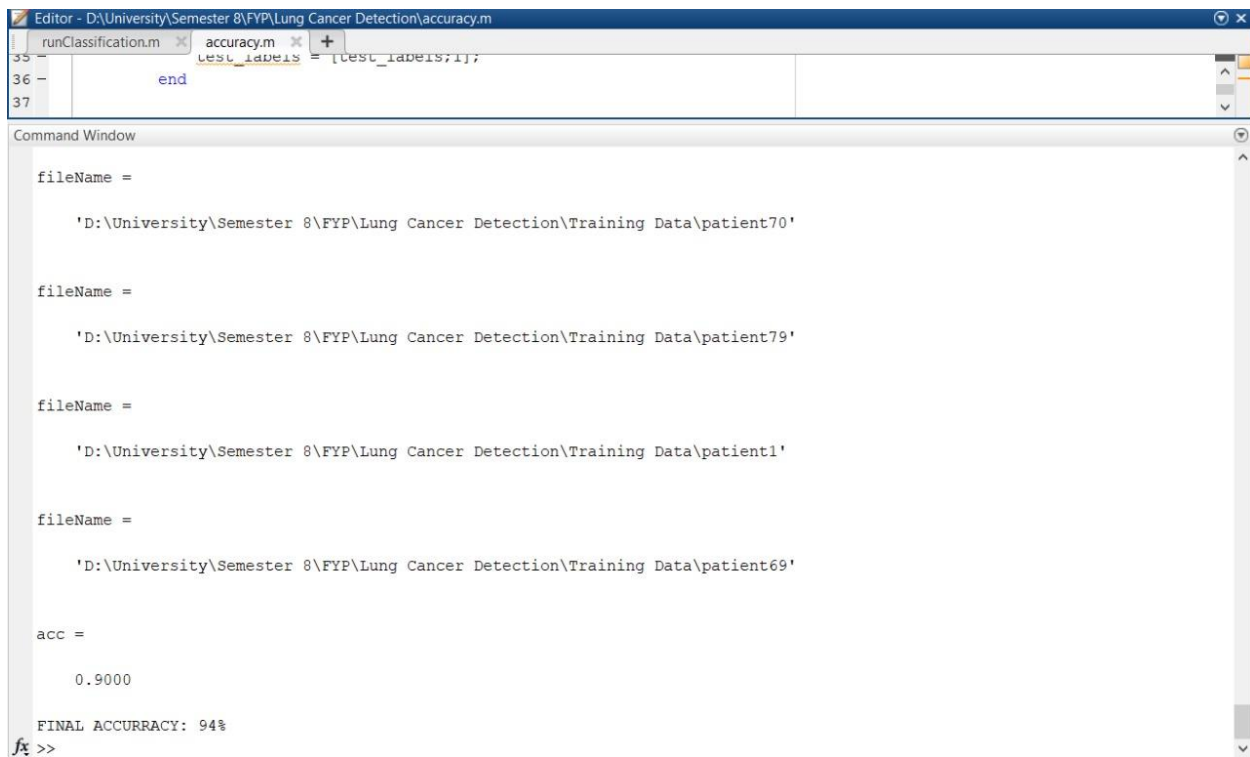
Fig 3.4.3.1 shows the line chart of the trained model accuracy

Fig 3.4.3.1 – Accuracy of final trained model



The final accuracy achieved was 94% as shown in **fig 3.4.3.2**.

Fig 3.4.3.2 – Screenshot of the accuracy of final model



The screenshot displays the MATLAB environment. The Editor window at the top shows a script named 'accuracy.m' with the following code:

```
35 test_labels = [test_labels;1];  
36 end  
37
```

The Command Window below shows the execution output:

```
fileName =  
    'D:\University\Semester 8\FYP\Lung Cancer Detection\Training Data\patient70'  
  
fileName =  
    'D:\University\Semester 8\FYP\Lung Cancer Detection\Training Data\patient79'  
  
fileName =  
    'D:\University\Semester 8\FYP\Lung Cancer Detection\Training Data\patient1'  
  
fileName =  
    'D:\University\Semester 8\FYP\Lung Cancer Detection\Training Data\patient69'  
  
acc =  
    0.9000  
  
FINAL ACCURACY: 94%  
fx >>
```

3.5 Component-External Entities Interface

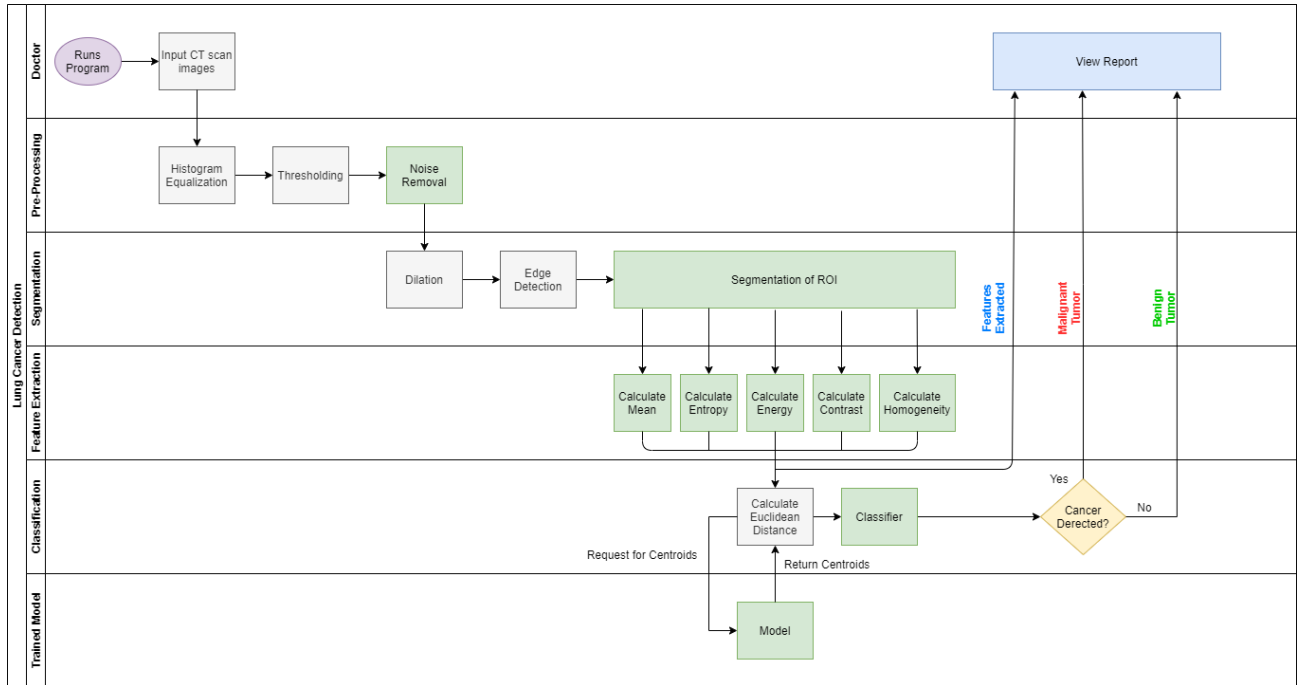
The project will be connected with the radiology department of the hospital. All the CT scans images of the lungs will be transferred to the doctor. The doctor will have access to CT scans and can test them on the project.

3.6 Screenshots/Prototype

3.6.1 Workflow

The swim lane diagram in **fig 3.6.1** shows the role that each component plays in the system in the form of a flowchart.

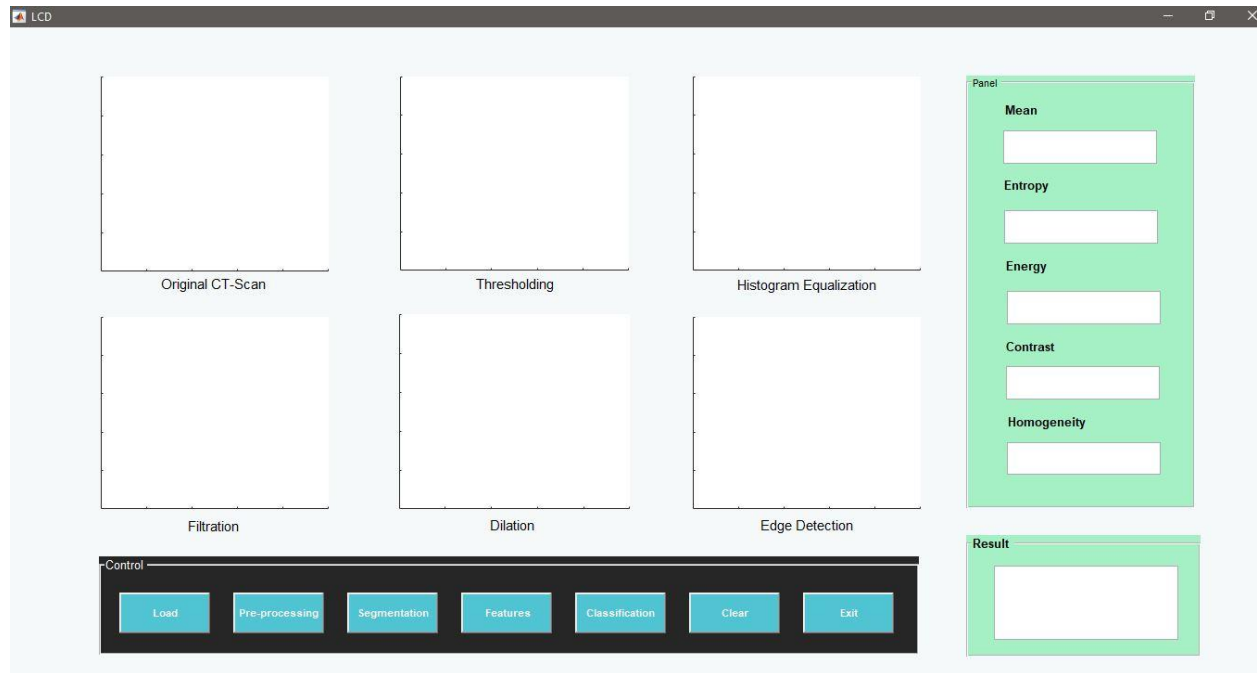
Fig 3.6.1 – Swim Lane Diagram



3.6.2 Main screen

All the inputs and outputs will be displayed on the main screen. The project will be controlled from the main screen.

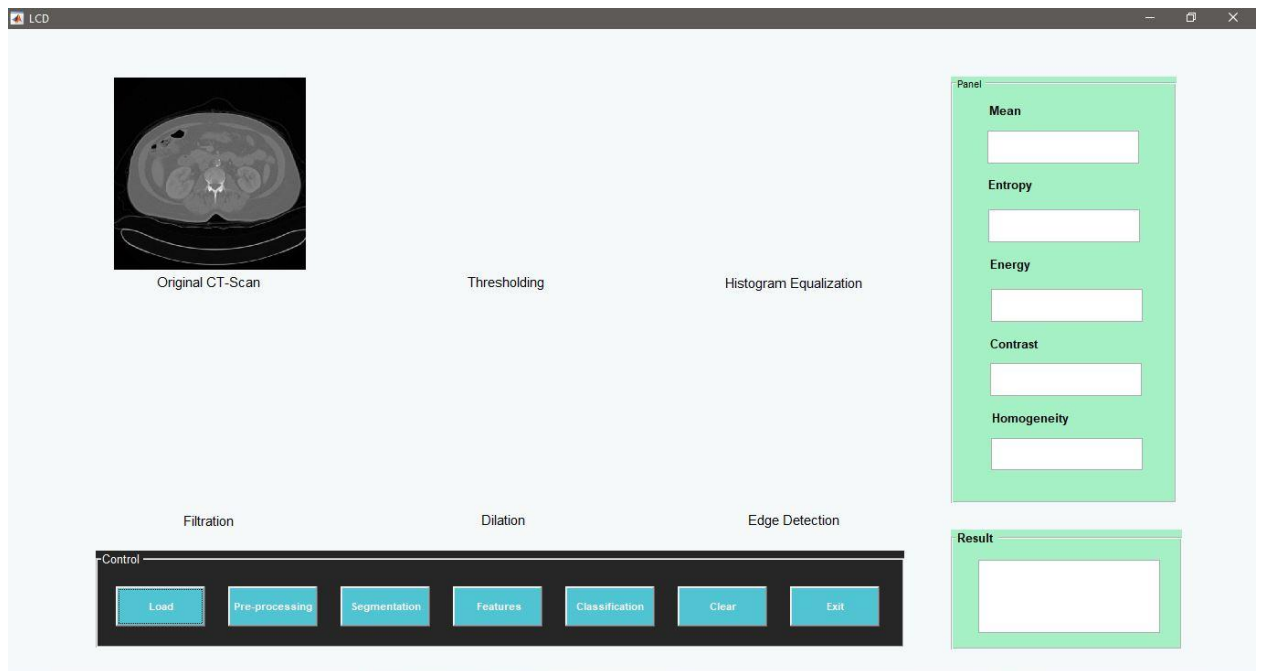
Fig 3.6.2 – Main screen



3.6.3 Software Execution

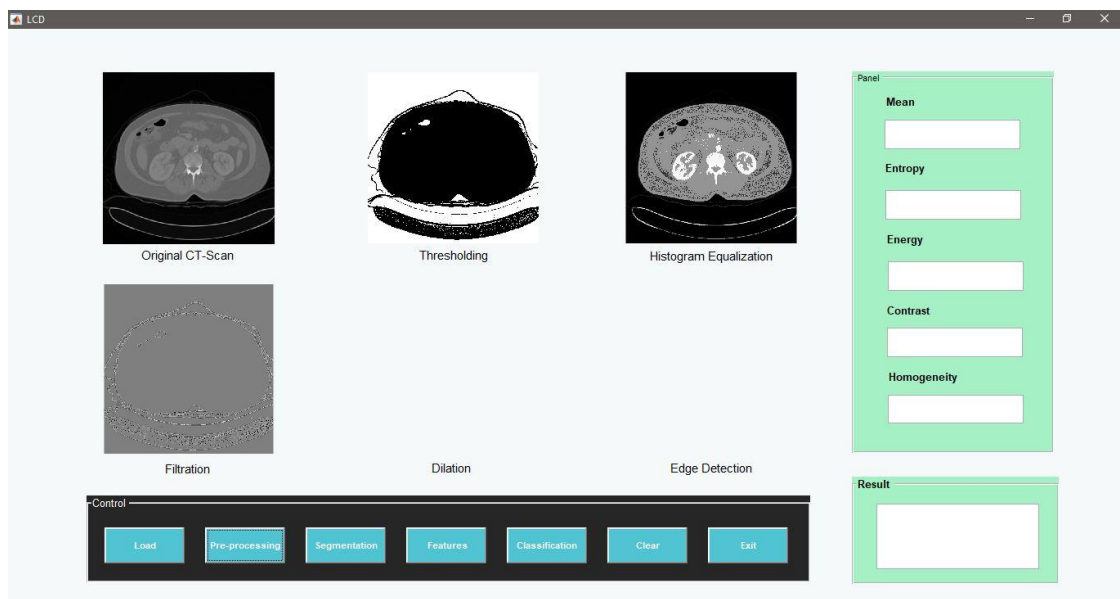
- 1) Loading an image from the dataset

Fig 3.6.3.1 - Loading image screen



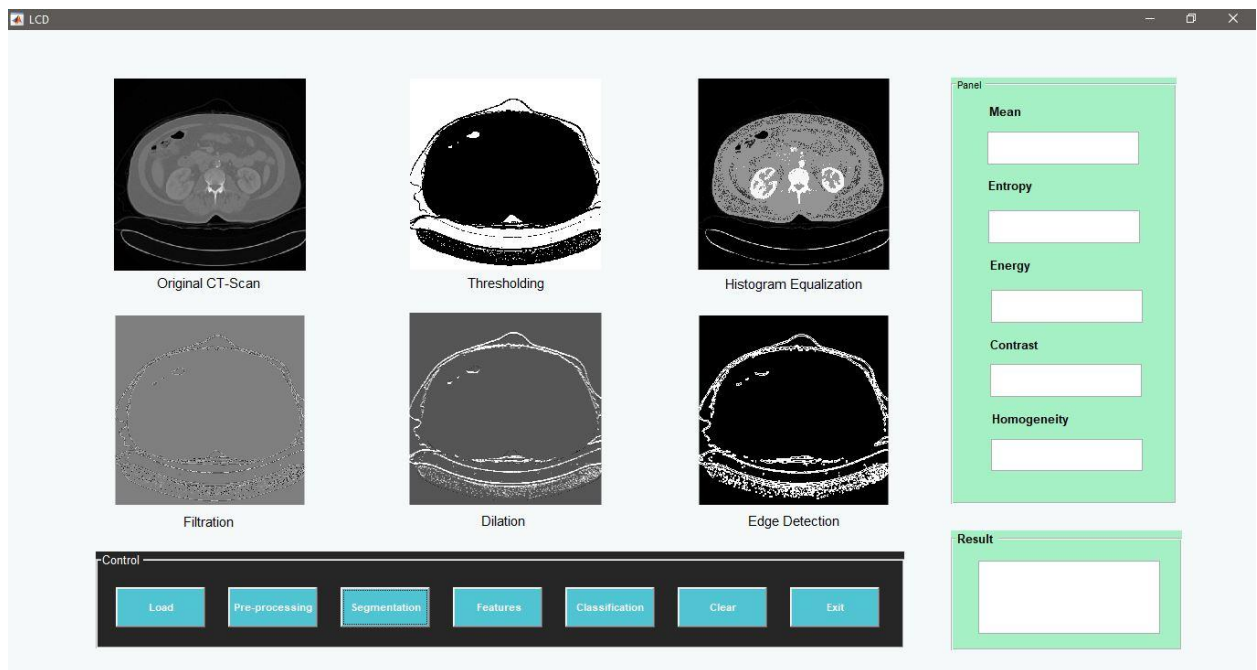
- 2) Applying pre-processing on the CT scan

Fig 3.6.3.2 - Pre-processing image screen



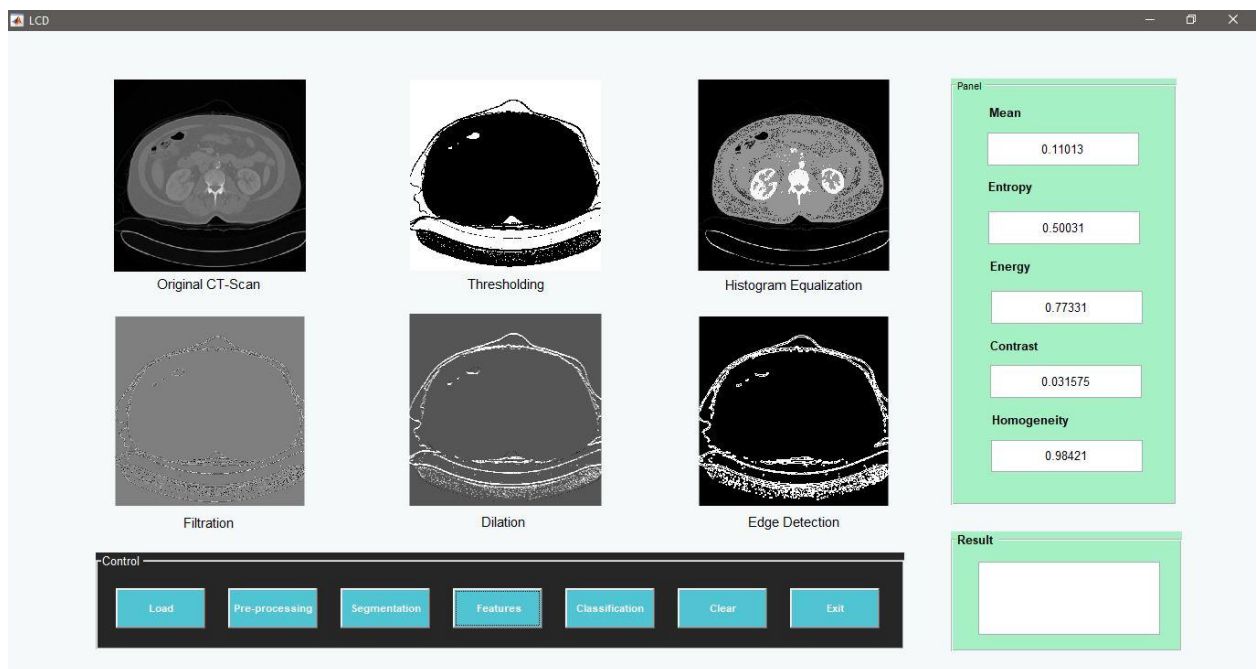
3) Segmenting the region of interest from the CT scan

Fig 3.6.3.3 – Segmenting image screen



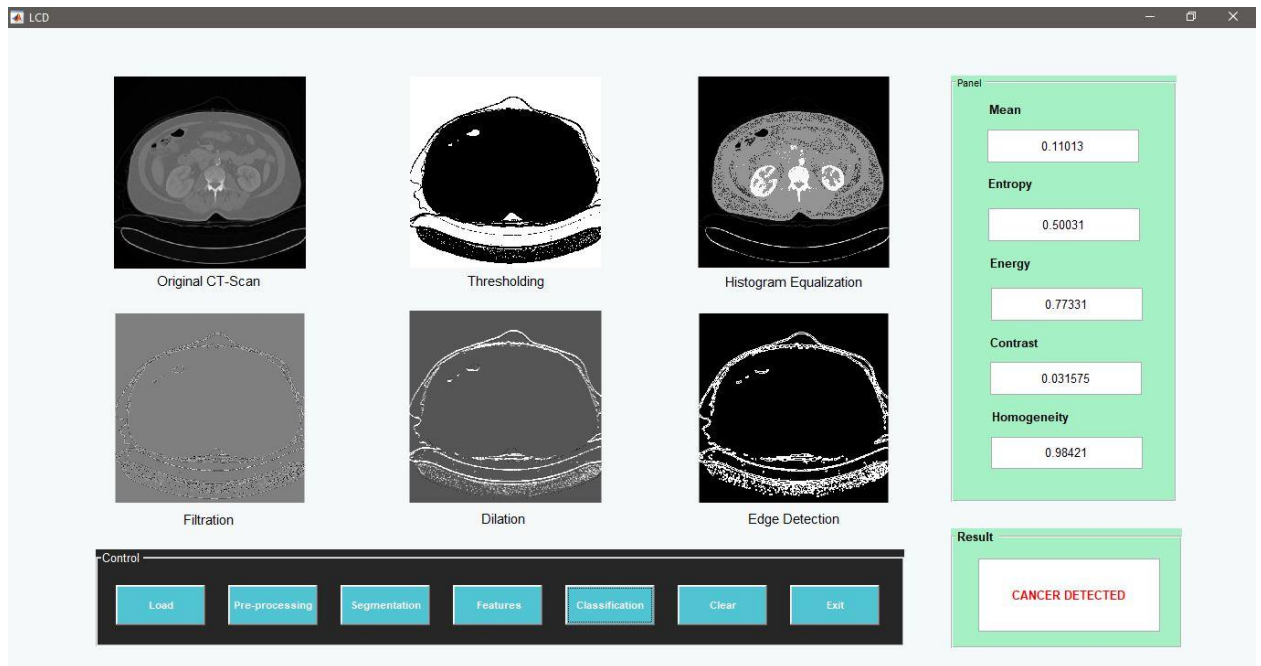
4) Extracting features from the segmented CT scan

Fig 3.6.3.4 – Feature extracting screen



5) Classifying the CT scan

Fig 3.6.3.5 –Classifying image screen



Chapter 4. Test Specification and Results

4.1 Test Case Specification

Identifier	TC-1
Related requirements(s)	CT scans exist in local with Patient name/case number as folder name
Short description	Open Directory to load CT scan for a Patient
Pre-condition(s)	CT scans exist in local directory; Directory is opening up successfully
Input data	Click on "Browse" to open directory
Detailed steps	<ol style="list-style-type: none">1. Click on "browser" button to open window explorer2. Go to directory that contain all patients CT scans3. Check format of CT scans it should be DICOM4. Click on Patient folder for which you want to upload CT scan of5. Click "Open" after selection so CT can be loaded into the system
Expected result(s)	CT scan image should be shown in "Original CT scan" image box
Post-condition(s)	Image of the CT scan for a patient will be successfully loaded in the software
Actual result(s)	Image of CT scan is loaded and displayed in Original CT scan box
Test Case Result	PASS

Table 4.1: TC-1

Identifier	TC-2
Related requirements(s)	CT scan is successfully upload and displaying in the software
Short description	Conversion of CT scan into Binary image
Pre-condition(s)	CT scan is successfully upload and displaying in the software in “Original CT scan” box
Input data	Click on “Pre-processing” button to perform action on CT scan
Detailed steps	<ol style="list-style-type: none"> 1. After successful upload and display of CT scan 2. Click on “Pre-processing” button to process Image 3. Grey and RGB CT scan image will be changed into black and white image 4. After processing the Image, it will be shown in “Thresholding”, “Histogram Equalization” and “Filtration” image boxes.
Expected result(s)	Black and white CT scan processed images it will show in “Thresholding”, “Histogram Equalization” and “Filtration” image boxes.
Post-condition(s)	Three boxes will show images of processed CT scans in black and white colors
Actual result(s)	Images are shown in “Thresholding”, “Histogram Equalization” and “Filtration” image boxes.
Test Case Result	PASS

Table 4.2: TC-2

Identifier	TC-3
Related requirements(s)	CT scan is successfully changed into binary and displayed in 4 image boxes
Short description	Dilation and edge refining of CT scan image
Pre-condition(s)	CT scan is successfully upload and displaying in the software in all 4 boxes
Input data	Click on “Segmentation” button to perform dilation and edge refinement of CT scan
Detailed steps	<ol style="list-style-type: none"> 1. After successful pre-processing and display of CT scan in all 4 boxes 2. Click on “Segmentation” button to process Image for dilation 3. Image will be filtered for missing pixels and edges will be refined in black and white color 4. After Dilation of the Image, it will be shown in “Dilation” and “Edge Detection” image boxes.
Expected result(s)	Black and white CT scan processed images with better pixel and edge refinement will be show in “Dilation” and “Edge Detection” image boxes.
Post-condition(s)	All boxes will show images of processed CT scans in black and white colors
Actual result(s)	Images are shown in “Dilation” and “Edge Detection” image boxes.
Test Case Result	PASS

Table 4.3: TC-3

Identifier	TC-4
Related requirements(s)	CT scan is successfully processed and displaying in all image boxes of software
Short description	Calculate and display the mean of segmented CT scan
Pre-condition(s)	CT scan is successfully segmented and displayed in the software in “Dilation” and “Edge Detection” box
Input data	Click on “Feature Extraction” button to get mean from segmented CT scan
Detailed steps	<ol style="list-style-type: none"> 1. After successful segmentation of CT scan and images are present in all 6 image boxes of software 2. Click on “Feature Extraction” button to get mean extracted from segmented CT scan images 3. Software will collect mean from the image 4. Software will then look for Entropy in image
Expected result(s)	Mean will be calculated but will only display once all calculations are processed
Post-condition(s)	All calculations need to be processed and displayed in correct right-hand boxes with title name of calculation processed in software
Actual result(s)	Mean will display in numeric format once all calculations are extracted and processed
Test Case Result	PASS

Table 4.4: TC-4

Identifier	TC-5
Related requirements(s)	CT scan is successfully processed and displaying in all image boxes of software
Short description	Calculate and display the Entropy of segmented CT scan
Pre-condition(s)	CT scan is successfully segmented and displayed in the software in “Dilation” and “Edge Detection” box
Input data	Click on “Feature Extraction” button to get entropy from segmented CT scan
Detailed steps	<ol style="list-style-type: none"> 1. After successful segmentation of CT scan and images are present in all 6 image boxes of software 2. Click on “Feature Extraction” button to get entropy extracted from segmented CT scan images 3. Software will collect entropy from the image 4. Software will then look for Energy in image
Expected result(s)	Entropy will be calculated but will only display once all calculations are processed
Post-condition(s)	All calculations need to be processed and displayed in correct right-hand boxes with title name of calculation processed in software
Actual result(s)	Entropy will display in numeric format once all calculations are extracted and processed
Test Case Result	PASS

Table 4.5: TC-5

Identifier	TC-6
Related requirements(s)	CT scan is successfully processed and displaying in all image boxes of software
Short description	Calculate and display the energy of segmented CT scan
Pre-condition(s)	CT scan is successfully segmented and displayed in the software in “Dilation” and “Edge Detection” box
Input data	Click on “Feature Extraction” button to get energy from segmented CT scan
Detailed steps	<ol style="list-style-type: none"> 1. After successful segmentation of CT scan and images are present in all 6 image boxes of software 2. Click on “Feature Extraction” button to get energy extracted from segmented CT scan images 3. Software will collect energy from the image 4. Software will then look for contrast in image
Expected result(s)	Energy will be calculated but will only display once all calculations are processed
Post-condition(s)	All calculations need to be processed and displayed in correct right-hand boxes with title name of calculation processed in software
Actual result(s)	Energy will display in numeric format once all calculations are extracted and processed
Test Case Result	PASS

Table 4.6: TC-6

Identifier	TC-7
Related requirements(s)	CT scan is successfully processed and displaying in all image boxes of software
Short description	Calculate and display the contrast of segmented CT scan
Pre-condition(s)	CT scan is successfully segmented and displayed in the software in “Dilation” and “Edge Detection” box
Input data	Click on “Feature Extraction” button to get contrast from segmented CT scan
Detailed steps	<ol style="list-style-type: none"> 1. After successful segmentation of CT scan and images are present in all 6 image boxes of software 2. Click on “Feature Extraction” button to get contrast extracted from segmented CT scan images 3. Software will collect contrast from the image 4. Software will then look for Homogeneity in image
Expected result(s)	Contrast will be calculated but will only display once all calculations are processed
Post-condition(s)	All calculations need to be processed and displayed in correct right-hand boxes with title name of calculation processed in software
Actual result(s)	Contrast will display in numeric format once all calculations are extracted and processed
Test Case Result	PASS

Table 4.7: TC-7

Identifier	TC-8
Related requirements(s)	CT scan is successfully processed and displaying in all image boxes of software
Short description	Calculate and display the Homogeneity of segmented CT scan
Pre-condition(s)	CT scan is successfully segmented and displayed in the software in “Dilation” and “Edge Detection” box
Input data	Click on “Feature Extraction” button to get Homogeneity from segmented CT scan
Detailed steps	<ol style="list-style-type: none"> 1. After successful segmentation of CT scan and images are present in all 6 image boxes of software 2. Click on “Feature Extraction” button to get Homogeneity extracted from segmented CT scan images 3. Software will collect Homogeneity from the image 4. Software will now display all calculated data into respective boxes. “Mean”, “Entropy”, “Energy”, “Contrast” and “Homogeneity” in numeric format.
Expected result(s)	Homogeneity will be calculated and all of the boxes will contain results to all calculations for Feature Extractions
Post-condition(s)	All calculations need to be processed and displayed in correct right-hand boxes with title name of calculation processed in software
Actual result(s)	“Mean”, “Entropy”, “Energy”, “Contrast” and “Homogeneity” will be displayed in numeric format in boxes
Test Case Result	PASS

Table 4.8: TC-8

Identifier	TC-9
Related requirements(s)	CT scan is successfully processed and all boxes for images and calculations are prefilled now
Short description	Display result for cancer detection after processing
Pre-condition(s)	CT scan is successfully processed and all boxes contain images and calculation results
Input data	Click on “Classification” button to display final result
Detailed steps	<ol style="list-style-type: none"> 1. After successful prefilled all processing information in all present boxes of software 2. Click on “Classification” button to get final result displayed in “Result” box for cancer detection 3. If cancer is detected, the box will display “Cancer Detected” text in Result Box 4. If cancer is not detected, the box will display “Cancer Not Detected” text in Result Box
Expected result(s)	Final result of cancer detected or not detected will be displayed in “Result” box
Post-condition(s)	Final result will be displayed in the box, and clear button can be used to refresh system for another processing on new patient
Actual result(s)	“Cancer Detected” or “Cancer Not Detected” will be displayed in the “Result” box
Test Case Result	PASS

Table 4.9: TC-9

Identifier	TC-10
Related requirements(s)	CT scan is successfully processed and all boxes for images and calculations are prefilled now
Short description	Display error in “Result” box if something went wrong
Pre-condition(s)	If any of the step is not properly processed or calculated then an error should be displayed
Input data	Click on “Classification” button to display final result
Detailed steps	<ol style="list-style-type: none"> 1. After successful prefilled all processing information in all present boxes of software 2. Click on “Classification” button to get final result displayed in “Result” box for cancer detection 3. If any of the step is not calculated properly or the image is not processed accurately, then an error stating “Error Occured” will be displayed in “Result” box.
Expected result(s)	Error will be displayed in “Result” box
Post-condition(s)	Error will be displayed in the box, so you can click on clear button to refresh software for another fresh processing
Actual result(s)	Error will be display in the “Result” box
Test Case Result	PASS

Table 4.10: TC-10

Identifier	TC-11
Related requirements(s)	CT scan is successfully processed and all boxes for images and calculations are prefilled now
Short description	Press “clean” button to clean all processed data
Pre-condition(s)	CT scan is successfully processed and all boxes contain data and final result is displayed
Input data	Click on “Clean” button to clean all the processed data
Detailed steps	<ol style="list-style-type: none"> 1. After successful processing of data, the result for cancer detection will be displayed 2. Click on “Clear” button to clean all the prefilled data for a fresh new processing 3. Data will all be cleared out and all boxes will be empty for another processing
Expected result(s)	All boxes should be empty
Post-condition(s)	All boxes will be empty for another processing
Actual result(s)	All boxes are empty and data is all cleared out
Test Case Result	PASS

Table 4.11: TC-11

Identifier	TC-12
Related requirements(s)	CT scans exist in local with Patient name/case number as folder name but doesn't contain DICOM format and is loaded successfully
Short description	Upload another format of CT scan in software
Pre-condition(s)	CT scans exist in local directory; Directory is opening up successfully and upload
Input data	Click on "Browse" to open directory
Detailed steps	<ol style="list-style-type: none"> 6. Click on "browser" button to open window explorer 7. Go to directory that contain all patients CT scans 8. Check format of CT scans it should be any format except DICOM 9. Click on Patient folder for which you want to upload CT scan of 10. Click "Open" after selection so CT cannot be loaded into the system and error will be displayed in "Result" Box
Expected result(s)	Error will be displayed in "result" box
Post-condition(s)	Error will be display and file need to be properly checked and re-uploaded into software
Actual result(s)	Error will be displayed
Test Case Result	PASS

Table 4.12: TC-12

Identifier	TC-13
Related requirements(s)	CT scan is successfully loaded and you press a button without following the normal flow
Short description	Display error in “Result” box if a step is missed out for processing
Pre-condition(s)	If any of the step is not properly processed or calculated then an error should be displayed
Input data	Click on any button but not following the normal flow
Detailed steps	<ol style="list-style-type: none"> 1. After successful upload of CT scan 2. Click on any button but not in proper flow 3. Then an error stating “Please complete the previous step first” will be displayed in “Result” box.
Expected result(s)	Error will be displayed in “Result” box
Post-condition(s)	Error will be displayed in the box, so you can click on clear button to refresh software for another fresh processing
Actual result(s)	Error will be display in the “Result” box
Test Case Result	PASS

Table 4.13: TC-13

4.2 Summary of Test Results

Module Name	Test cases run	Number of defects found	Number of defects corrected so far	Number of defects still need to be corrected
Unit Testing	All test cases are executed	3	3	0
Integration Testing	Test cases are executed	2	2	0
Regression Testing	End to End testing for all Test-cases	0	0	0

Table 4.14: Summary of All Test Results

Chapter 5. Conclusion and Future Work

5.1 Project summary

The project detects tumor cells in the lungs of a patient from CT scan images. The system takes Dicom images as an input which is the output of a CT scan, it is divided into 60 and above levels hence there are more than 60 Dicom images of a single patient. The project runs different pre-processing techniques and classification models on these dicom images to classify the tumor. A lung CT scan can be classified into two classes that are malignant tumor and benign or no tumor. This project will eliminate needless biopsies. This project will accurately detect malignant tumors hence there will be no need for a biopsy for a patient with a benign tumor. The project will end up saving both the doctor's and patient's time. In normal procedures, multiple tests need to be carried out for detecting and ensuring the presence of tumor, however, this project only needs a CT scan of the lungs. The project will classify based on different features extracted from the CT scan.

The accuracy of this project was calculated by the k fold cross-validation method, keeping the value of k to be 10. It was measured and compared with some other projects in this domain and it stamped out all of them with an accuracy of 94%.

5.2 Problems faced and lessons learned

5.2.1 Dataset

Hospitals keep the data of patients confidential. We could only find one dataset initially containing 80 CT scans. To increase our accuracy, we were data-hungry. Finding CT scans of normal healthy lungs was easy, but no one was ready to provide us data of cancerous patients as per their confidentiality acts.

5.2.2 Dicom Images

A patient's CT scan contained more than 60 dicom images. It was a challenge for us to understand the format of the images and choose methods to apply to them.

5.2.3 Segmentation

Segmentation of Dicom images was very strenuous as there were many levels on which the segmentation had to be applied. We had to try multiple methods and train different models to precisely segment the region of interest.

5.2.4 Choosing Features

There was an enormous number of features that could be extracted from these images. It was very challenging to select the best ones. We had to try different combinations of various features and train models on them to get our desired results.

5.2.5 Covid-19

Due to the ongoing pandemic, many of our efforts were limited, the quality of the work we had expected could not be met completely. We could not visit local hospitals to personally gather data from them and get our project deployed in Pakistan.

5.3 Future work

5.3.1 Bounding box tumor area

Bounding boxes added to the area containing the tumor in the CT scan and displayed on the GUI

5.3.2 Area and depth of tumor

Area and depth of tumor to be calculated from the precise segmentation. This will be a challenge as there are so many layers of dicom images.

5.3.3 Location of tumor

Exact coordinates of the area containing tumor to be calculated and displayed. It will be easier for the doctors to perform a biopsy on patients containing malignant tumors after the addition of this module, as the doctor will have prior knowledge of the tumor's location.

References

1. Senthil Kumar, K., Venkatalakshmi, K., & Karthikeyan, K. (2019). Lung Cancer Detection Using Image Segmentation by means of Various Evolutionary Algorithms. *Computational and Mathematical Methods in Medicine*, 2019, 1-16. doi: 10.1155/2019/4909846
2. Winn, N., Spratt, J., Wright, E., & Cox, J. (2014). Patient reported experiences of CT guided lung biopsy: a prospective cohort study. *Multidisciplinary Respiratory Medicine*, 9(1), 53. doi: 10.1186/2049-6958-9-53
3. Nasser, I., & Abu-Naser, S. (2019). Lung Cancer Detection Using Artificial Neural Network. *International Journal of Engineering And Information Systems (IJEAIS)*, 3(3), 17-23.
4. Bandyopadhyay, S. (2012). Edge detection from CT images of lung. *International journal of engineering science & advanced technology [ijesat]*, 2(1), 34-37.
5. Md. Badrul Alam Miah, Mohammad Abu Yousuf(2015). Detection of Lung Cancer from CT Image Using Image Processing and Neural Network. 2nd Int'l Conf on Electrical Engineering and Information & Communication Technology (ICEEICT) 20 IS Jahangirnagar University, Dhaka-1342, Bangladesh.
6. Rachid Sammouda (2016). Segmentation and Analysis of CT Chest Images for Early Lung Cancer Detection. 2016 Global Summit on Computer & Information Technology.
7. Wafaa Alakwaa, Mohammad Nassef, Amr Badr. (2017). Lung Cancer Detection and Classification with 3D Convolutional Neural Network (3D-CNN) *International Journal of Advanced Computer Science and Applications (IJACSA)*, Vol. 8, No. 8, 2017
8. Alzorgani, M., & Ugail, H. (2018, October 17th). Comparative Study of Image Classification using Machine Learning Algorithms. *The 2Nd Annual Innovative Engineering Research Conference (AIERC 2018)*. Bradford, UK.
9. Coleman, M., Forman, D., Bryant, H., Butler, J., Rachet, B., & Maringe, C. et al. (2011). Cancer survival in Australia, Canada, Denmark, Norway, Sweden, and the UK, 1995–2007 (the International Cancer Benchmarking Partnership): an

- analysis of population-based cancer registry data. *The Lancet*, 377(9760), 127-138. doi: 10.1016/s0140-6736(10)62231-3
9. El-Baz, A., Beache, G., Gimel'farb, G., Suzuki, K., Okada, K., & Elnakib, A. et al. (2013). Computer-Aided Diagnosis Systems for Lung Cancer: Challenges and Methodologies. *International Journal Of Biomedical Imaging*, 2013, 1-46. doi: 10.1155/2013/942353
 10. Lung Cancer Basics. (2021). Retrieved 21 May 2021, from <https://www.lung.org/lung-health-diseases/lung-disease-lookup/lung-cancer/learn-about-lung-cancer/lung-cancer-basics#:~:text=Lung%20cancer%20is%20cancer%20that,tumors%20are%20called%20malignant%20tumors>.
 11. Yarnall, B. (2021). The first complete picture of how long it takes to diagnose cancer in England - Cancer Research UK - Science blog. Retrieved 21 May 2021, from: <https://scienceblog.cancerresearchuk.org/2019/06/30/the-first-complete-picture-of-how-long-it-takes-to-diagnose-cancer-in-england/#:~:text=The%20average%20time%20for%20a%20diagnosis%20of%20bow>
 12. Lung Cancer Fact Sheet. (2021). Retrieved 23 May 2021, from <https://www.lung.org/lung-health-diseases/lung-disease-lookup/lung-cancer/resource-library/lung-cancer-fact-sheet>

Appendix A Glossary

Benign Tumor: A benign tumor is non-cancerous. It does not invade nearby tissue or spread to other parts of the body the way cancer can.

Malignant Tumor: Malignant tumors are cancerous. The cells can grow and spread to other parts of the body.

Inhomogeneity:

An MRI term for the lack of homogeneity or uniformity in a main magnetic field.

Pulmonary: of the nature of a lung; lung-like.

Bronchioles: Air passages inside the lungs that branch off like tree limbs from the bronchi—the two main air passages into which air flows from the trachea (windpipe) after being inhaled through the nose or mouth.

CAD: Computer-aided detection, also called computer-aided diagnosis, are systems that assist doctors in the interpretation of medical images

CNN: In deep learning, a convolutional neural network is a class of deep neural networks, most commonly applied to analyzing visual imagery. They are also known as shift invariant or space invariant artificial neural networks, based on their shared-weights architecture and translation invariance characteristics.

Invasive: involving the introduction of instruments or other objects into the body or body cavities.

Biopsy: an examination of tissue removed from a living body to discover the presence, cause, or extent of a disease.

Thresholding: In digital image processing, thresholding is the simplest method of segmenting images. From a grayscale image, thresholding can be used to create binary images.

HIPAA Standard: The HIPAA Privacy Rule establishes national standards to protect individuals' medical records and other personal health information and applies to health plans, health care clearinghouses, and those health care providers that conduct certain health care transactions electronically.

Data Protection Acts of 1998 and 2003: This was a United Kingdom Act of Parliament designed to protect personal data stored on computers or in an organized paper filing system. [Click to read more](#)

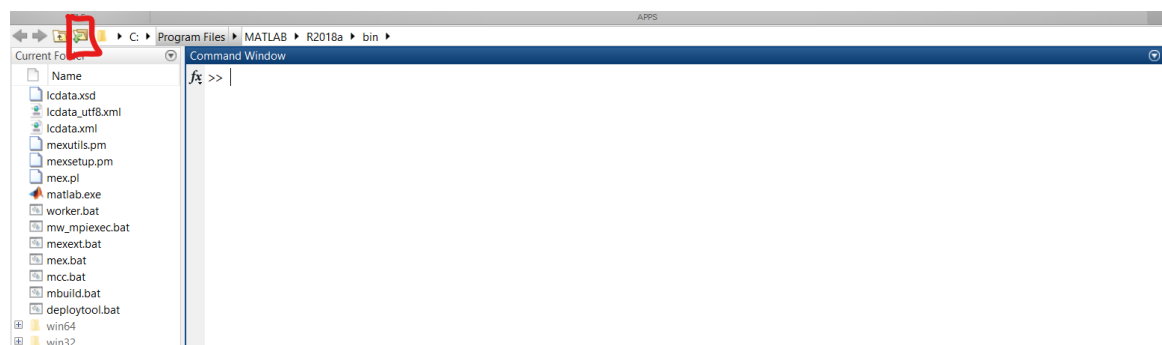
Neural Network: A neural network is a series of algorithms that endeavors to recognize underlying relationships in a set of data through a process that mimics the way the human brain operates.

Appendix B Deployment/Installation Guide

Follow the steps below to install and run the application on your system.

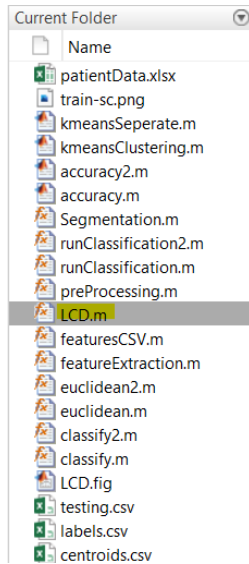
- Go to the [mathworks.com](https://www.mathworks.com) website and sign in to your account or create a new one. You need an account to purchase the Matlab license.
- After you are logged in simply visit <https://www.mathworks.com/downloads/> and click on Download R2021a or any latest version displaying there.
- After the zip file is downloaded, extract it and run the setup file as administrator.
- In the setup select install using file installation key and add the key provided to you by Matlab.
- Next enter the path to your license file place in the Matlab folder by the name license.lic
- Select installation destination and then select the products to install. For the lung cancer detection program you only need to install the Image processing toolbox and machine learning toolbox.
- Finally, click on install and run Matlab.
- Now click on the browse folder button as shown in **fig b1** below and select the Lung Cancer Detection Folder.

Fig b1 – Select Folder in Matlab



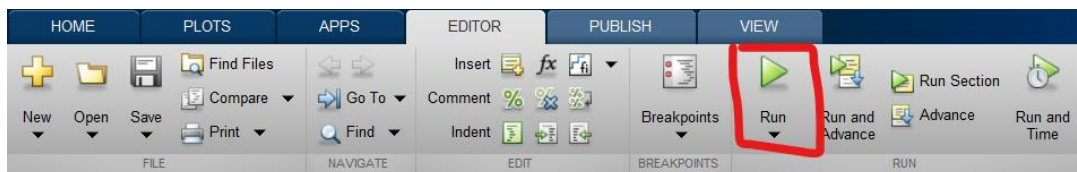
- Now choose **LCD.m** from the current folder path as shown in **fig b2** below.

Fig b2 – Select LCD.m in current folder



- Finally, simply click on '**Run**' to run the program as shown in **fig b3** below

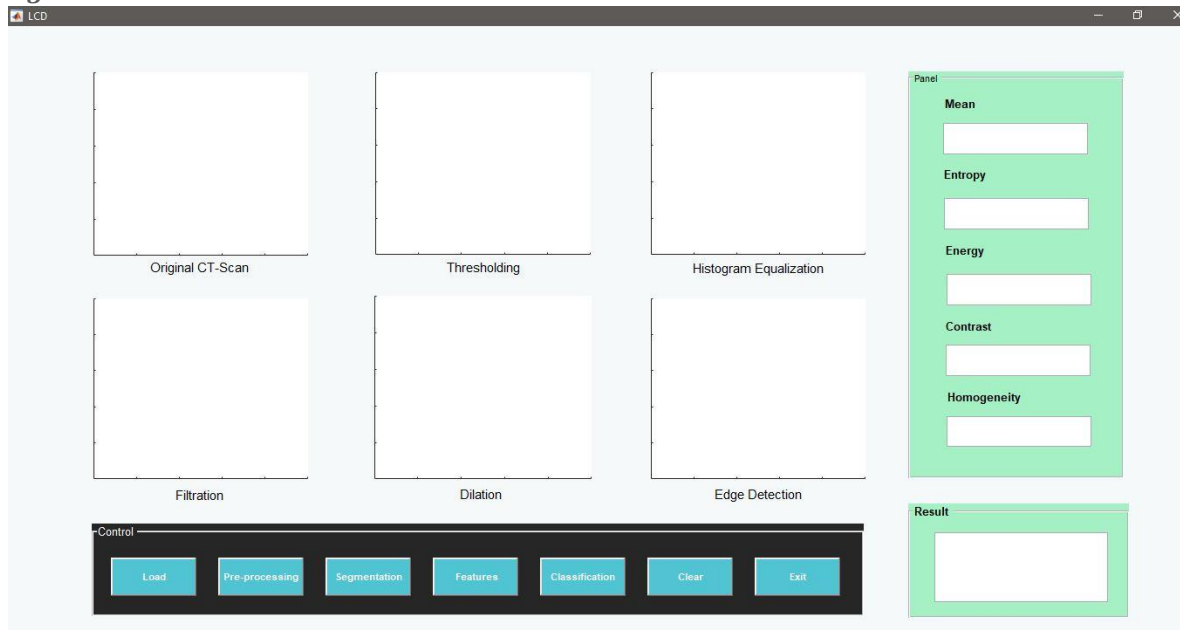
Fig b3 – Run



Appendix C User Manual

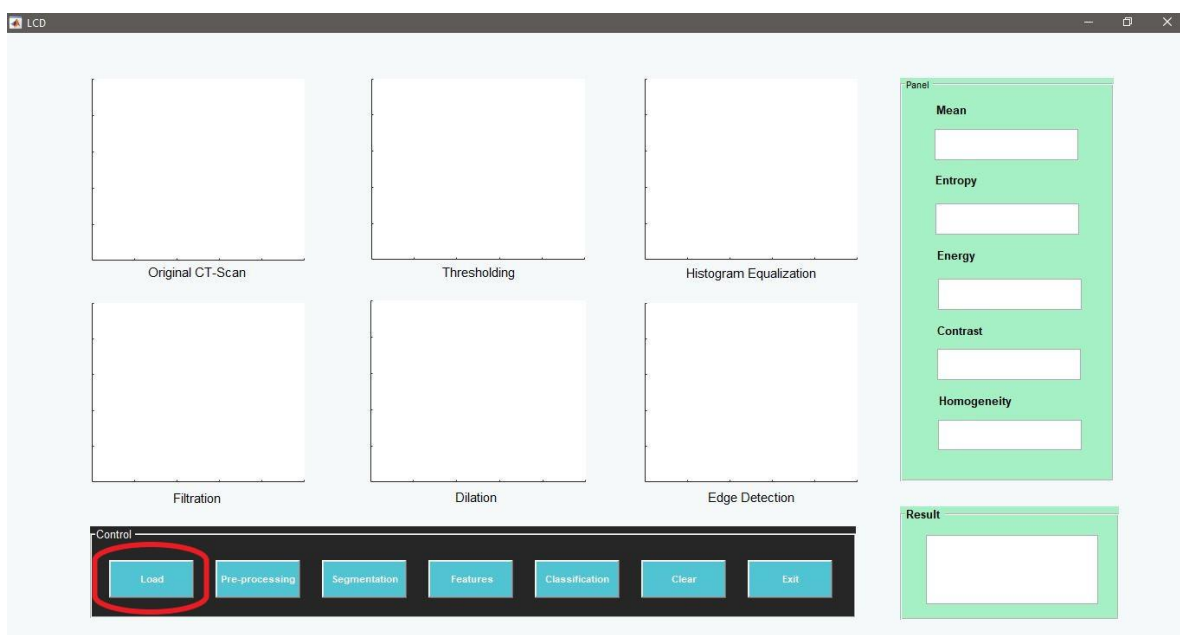
- 1) Run **LCD.m** file, the software will run as shown in **fig c1**

Fig c1 - Main GUI



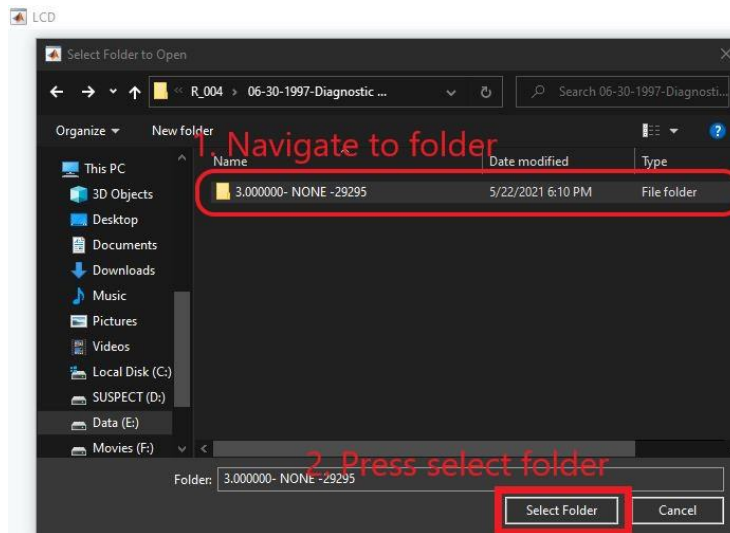
- 2) Press the **LOAD** button as shown in **fig c2** to open the directory window for the selection of folder containing DICOM images.

Fig c2 - Load



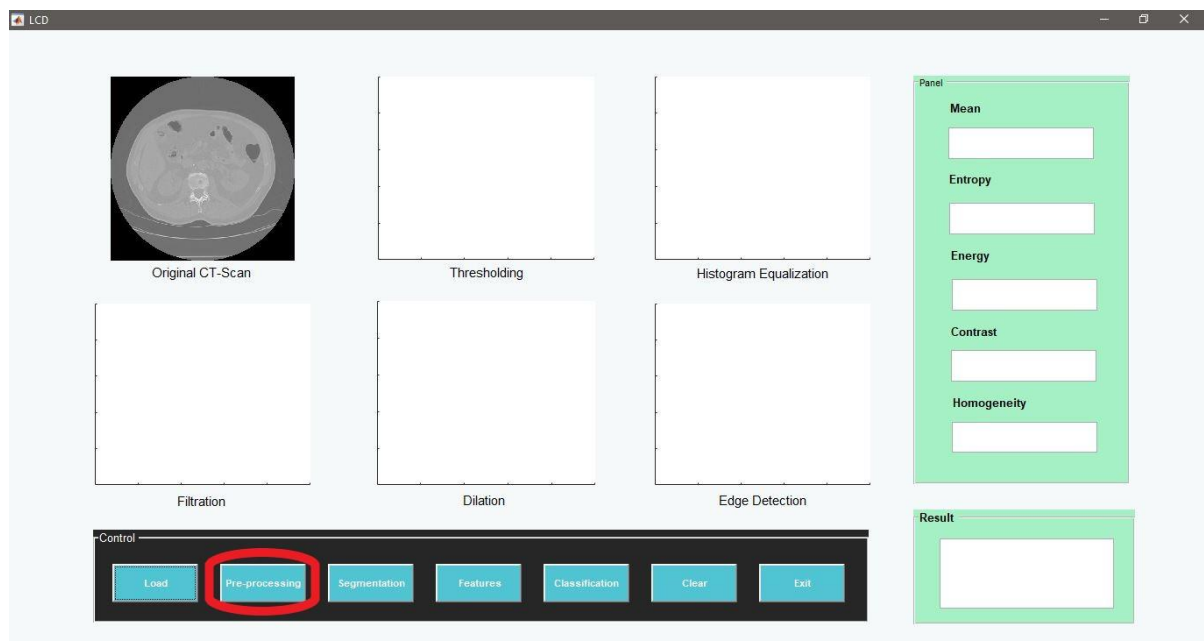
- 3) A directory window will open once the load button is pressed, select the folder containing the DICOM images of CT scan, and press the Select Folder button as shown in **fig c3**.

Fig c3 – browse folder path



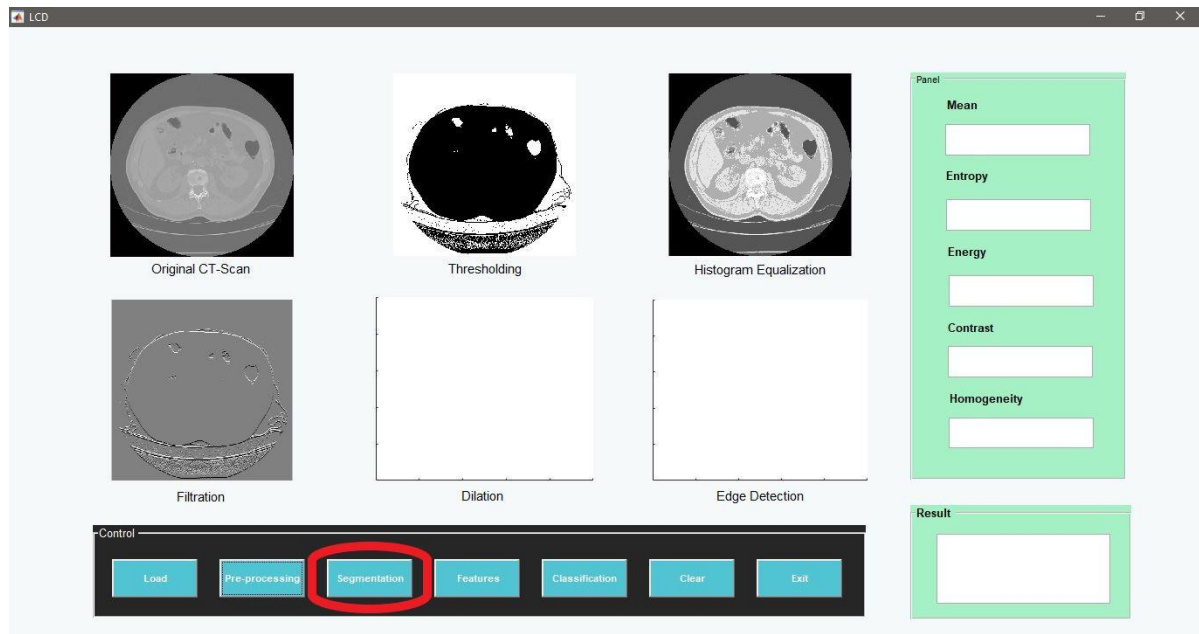
- 4) Once the CT scan is loaded, Press the **Pre Processing** button as shown in **fig c4**

Fig c4 – pre-processing button



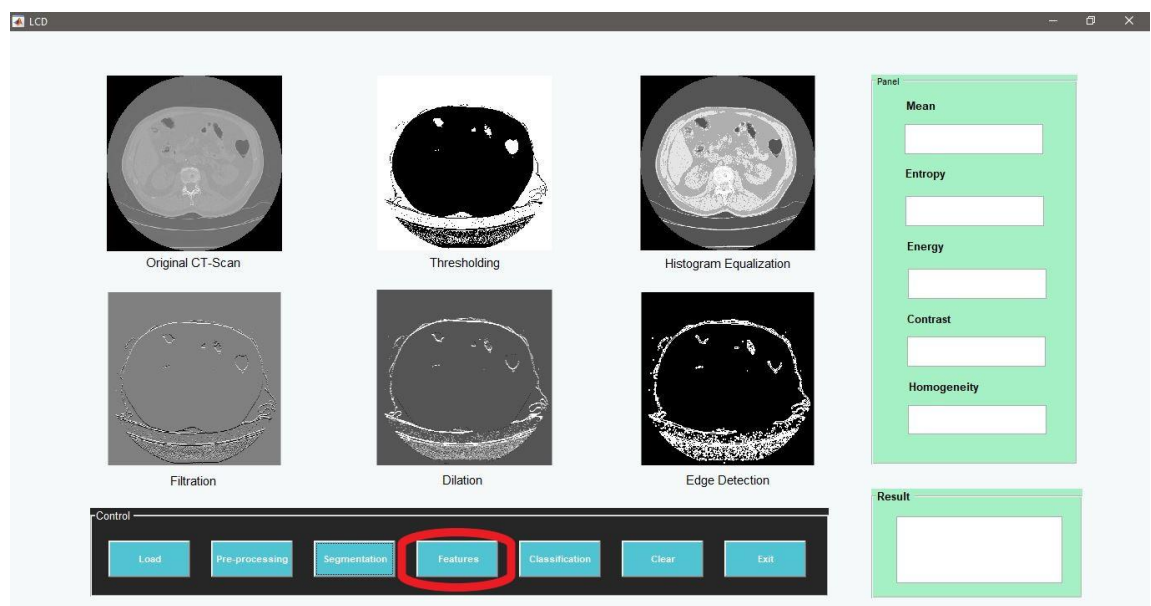
- 5) After completion of pre-processing, press the **Segmentation** button as shown in **fig c5**.

Fig c5 – Segmentation Button



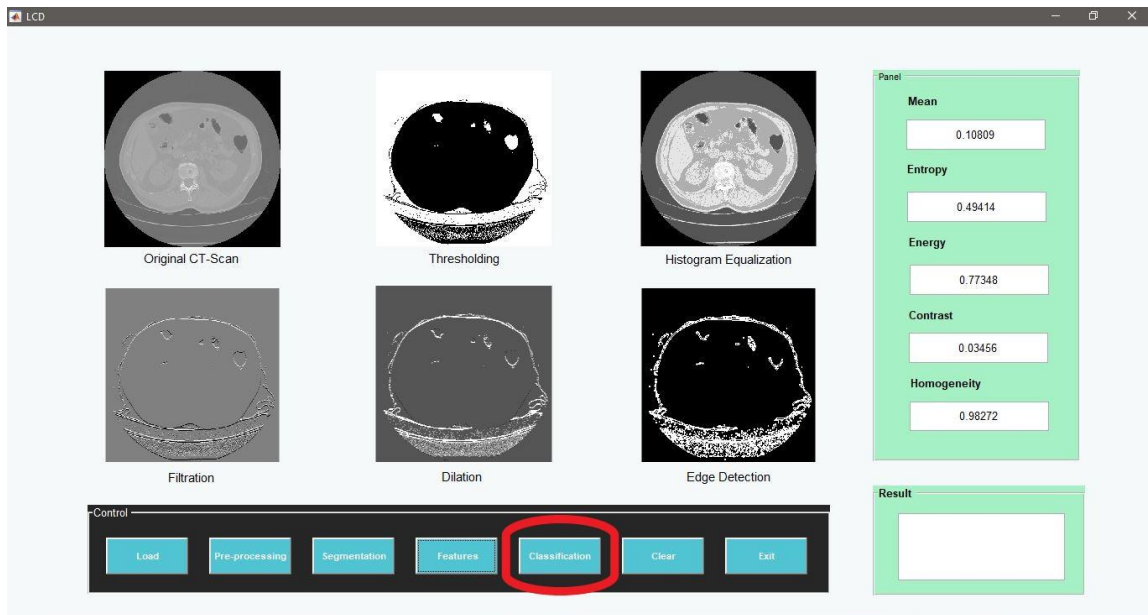
- 6) Press the **Features** button to extract all the features from the CT scan as shown in **fig c6**.

Fig c6 – Features Button



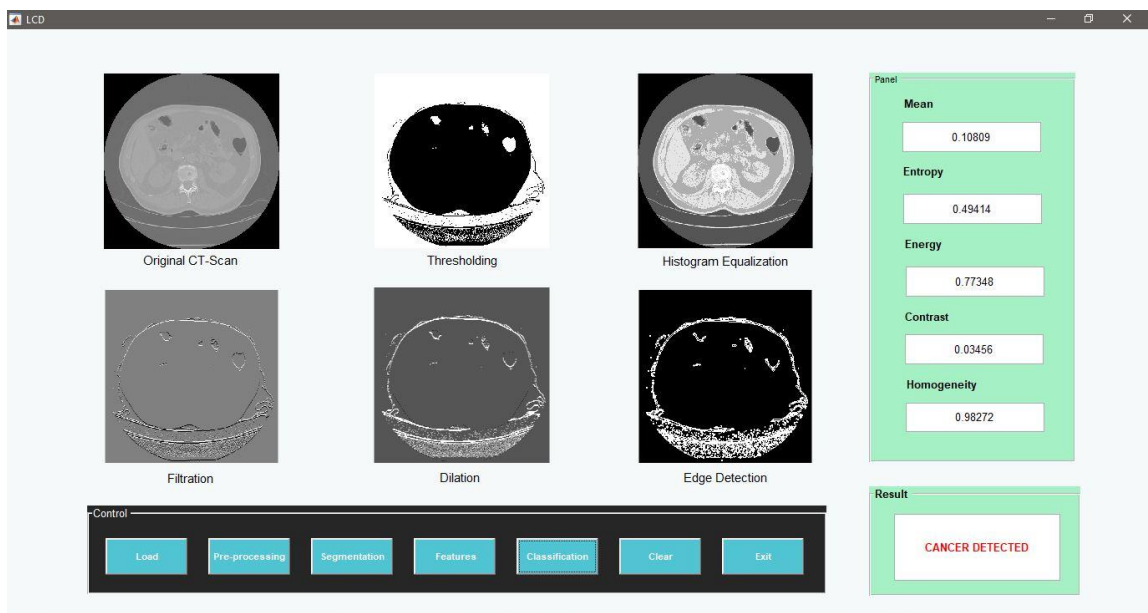
7) To classify the CT scan press the **Classification** button as shown in **fig c7**

Fig c7 – Classification



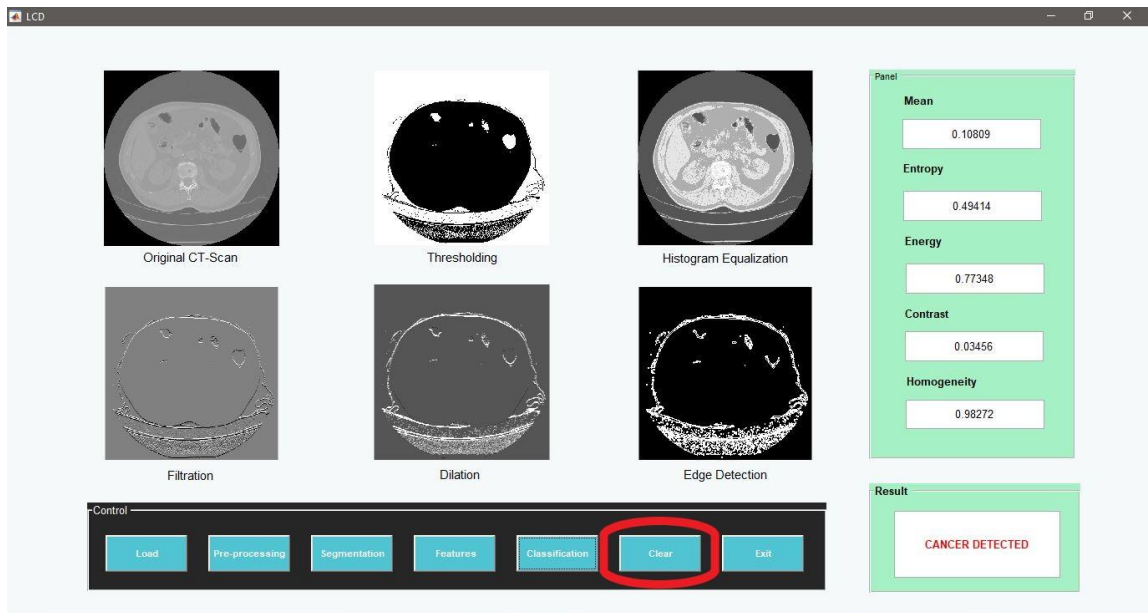
8) After the successful run of the program, result will be displayed as shown in **fig c8**.

Fig c8- Results



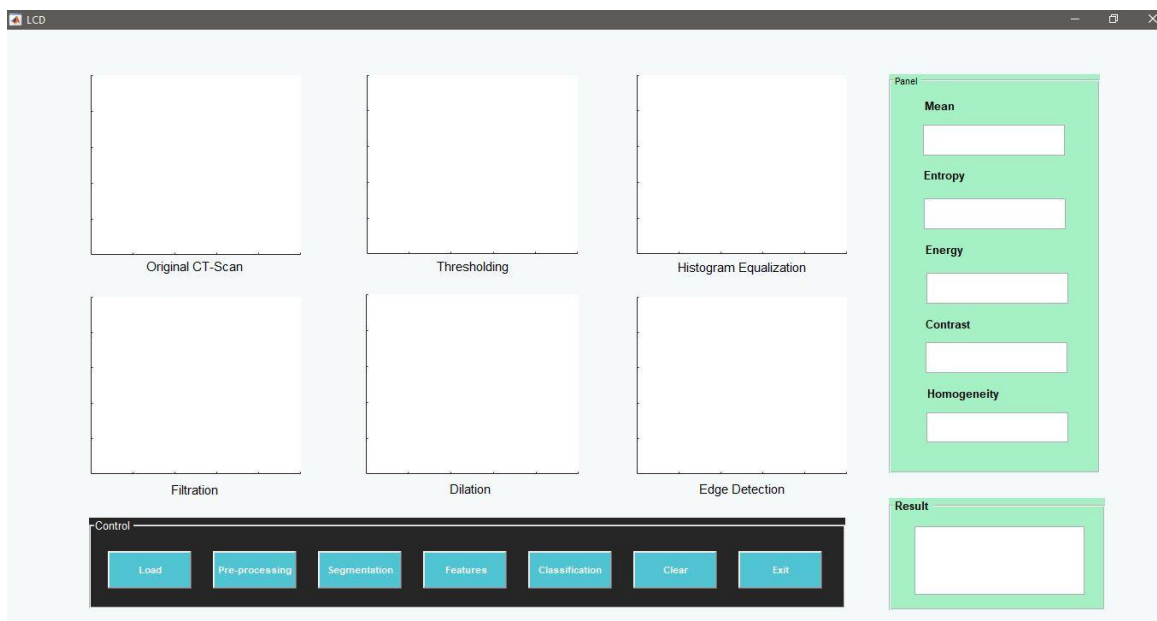
- 9) If you want to re-run the program for another patient, press the **Clear** button as shown in **fig c9**

Fig c9 – Clear Button



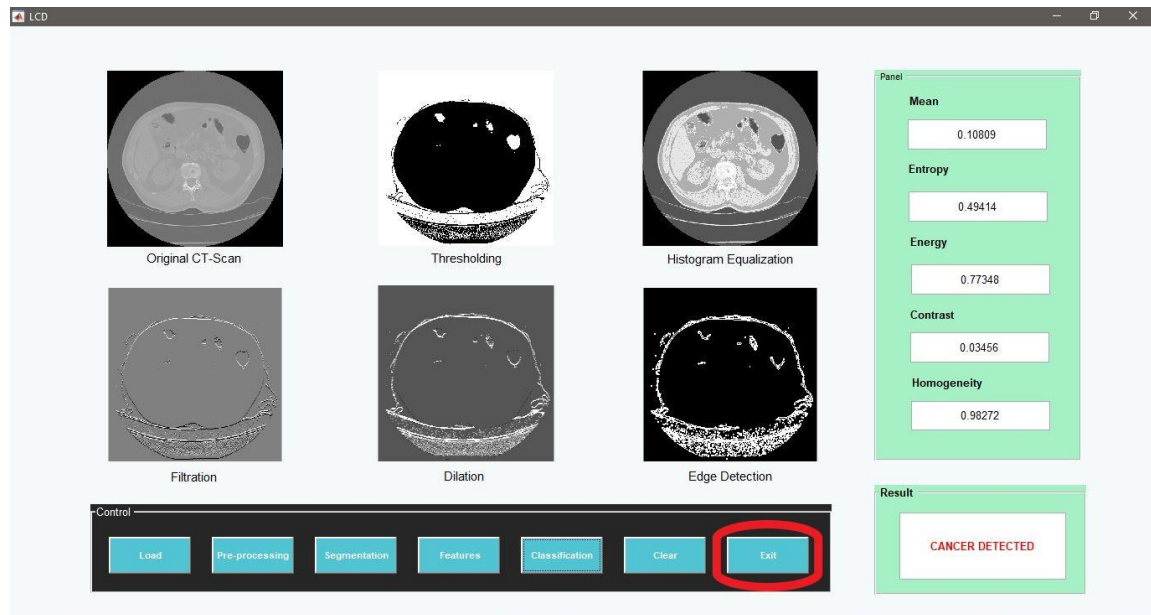
- 10) The window will reset as shown in **fig c10** and you can repeat the steps 2 to 8

Fig c10 – Cleared Window



11) To exit press the **Exit** button as shown in **fig c11**

Fig c11 – Exit Button



Appendix D Student Information Sheet

Roll No	Name	Email Address (FC College)	Frequently Checked Email Address	Personal Cell Phone Number
21-10579	Muneeb Amer	21-10579@formanite.fccollege.edu.pk	21-10579@formanite.fccollege.edu.pk	03124220350
21-10842	Mohid Ali Gill	21-10842@formanite.fccollege.edu.pk	21-10842@formanite.fccollege.edu.pk	03328464176
21-10053	Maham Jamil	21-10053@formanite.fccollege.edu.pk	21-10053@formanite.fccollege.edu.pk	03224116827

Appendix E Plagiarism Free Certificate

This is to certify that, I am **Muneeb Bin Amer S/D/o Amer Iqbal**, group leader of FYP under registration no **21-10579** at Computer Science Department, Forman Christian College (A Chartered University), Lahore. I declare that my Final year project report is checked by my supervisor and the similarity index is **17%** that is less than 20%, an acceptable limit by HEC. Report is attached herewith as Appendix F. To the best of my knowledge and belief, the report contains no material previously published or written by another person except where due reference is made in the report itself.

Date: 17/06/2021 Name of Group Leader: **Muneeb Bin Amer** Signature:



Name of Supervisor: _____ Co-Supervisor (if any): _____

Designation: _____ Designation: _____

Signature: _____ Signature: _____

Senior Project Management Committee Representative: _____

Signature: _____

Appendix F Plagiarism Report

lung cancer

ORIGINALITY REPORT

17%	10%	6%	13%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	Submitted to Higher Education Commission Pakistan Student Paper	8%
2	Submitted to Marquette University Student Paper	1%
3	gtext.gfz-potsdam.de Internet Source	<1%
4	Erik Bergenholtz, Emiliano Casalicchio, Dragos Ilie, Andrew Moss. "Chapter 3 Detection of Metamorphic Malware Packers Using Multilayered LSTM Networks", Springer Science and Business Media LLC, 2020 Publication	<1%
5	Md. Badrul Alam Miah, Mohammad Abu Yousuf. "Detection of lung cancer from CT image using image processing and neural network", 2015 International Conference on Electrical Engineering and Information Communication Technology (ICEEICT), 2015 Publication	<1%

prognosis.com