**LUNG CANCER DETECTION USING**

**CT SCAN IMAGE PROCESSING**

**TEH YI CHANG**

**28695127**

**THUNG HUI LING**

**27208699**

**FIT3081 Image Processing**

**Implementation project**

**Semester 1, Year**

**School of IT**

**Monash University Malaysia**

**June, 2019**

**ABSTRACT**

Image processing techniques are widely used in medical research, especially for cancer detection nowadays. In this paper, we aim to do a thorough literature review and evaluate the existing image processing techniques used in lung cancer detection. We study and implement the image processing methods that we have learned from the unit and from the research papers, to detect and classify between abnormal and normal lung on a set of lungs computer tomography (CT) scan images. Different techniques were analysed, and from the experimental results, we found the effectiveness of our approach. However, future improvements are needed to produce an even more accurate results and for the purpose of this, overall limitation, drawbacks were pointed out.

**TABLE OF CONTENTS**

**CHAPTER**

1. **INTRODUCTION**
   1. **Background**
   2. **Motivation**
   3. **Aims & Objectives**
2. **LITERATURE REVIEW**
3. **METHODOLOGY**
   1. **Overall System Architecture**
   2. **Image Acquisition**
   3. **Pre-processing**
      1. **Smoothing Algorithms**
      2. **Image Enhancement Algorithms**
      3. **Image Segmentation Algorithms**
   4. **Feature Extraction**
      1. **Binarization & Masking Algorithms**
      2. **General Characteristics Algorithms**
4. **RESULTS AND DISCUSSIONS**
   1. **Pre-processing**
      1. **Smoothing Algorithms**
      2. **Image Enhancement Algorithms**
      3. **Image Segmentation Algorithms**
   2. **Feature Extraction**
      1. **Binarization & Masking Algorithms**
      2. **General Characteristics Algorithms**
5. **CONCLUSSION AND RECOMMENDATIONS**

**REFERENCES**

**APPENDICES**

**CHAPTER 1**

**INTRODUCTION**

* 1. **Background**

Lung cancer is one of the most dangerous and deadliest cancers. According to American Cancer Society (2019), more people die of lung cancer than other cancers combined each year. There are two type of lung cancer, which are small cell lung cancer and non-small cell lung cancer. Lung cancer happens when cells mutate. When those abnormal cells multiplying and growing, it can turn into a tumour. Cancer cells can spread from the lungs to other part of the body through bloodstream and lymph fluid. This mutation can be caused by various factors, for example, exposure to radon, dangerous and harmful chemicals, second-hand smoke, particle pollution.

* 1. **Motivation**

The process of early detection plays an important role to prevent cancer cells from spreading. Although computed tomography (CT) scan imaging is one of the best imaging techniques for medical uses, it is hard to interpret and identify just from CT scan images. Hence, image processing is used to enhance the detection process. For this project, we plan to use our knowledge we learn from the FIT3081 class and the research papers we read up to come out an effective program that could help to contribute in this field. Our system can be adapted and used as a tool to benefit the doctors in the following ways:

1. Provides a rough idea whether the CT scan image likely belong to normal people or lung cancer patient
2. Provides an overview of the characteristics features of the lung nodules to judge the possible existence of tumour in the lungs.

There are several diagnostic imaging techniques such as MRI, PET, Chest X-ray and CT scan. MRI scans are less popular due to low availability and costly (Aggarwal, Furqan, & Kalra, 2015). On the other hand, CT scans are more sensitive and provides a higher contrast than chest X-ray. A low-dose computed tomography (LDCT) scan can detect early stage diseases 6-10 times more frequent than X-ray (Hollings & Shaw, 2002). However, there are some limitations to LDCT scans. Peikert et. al. (2018) has raised that high false positive rates from CT leads to unnecessary invasive procedures and patient anxiety. Hollings and Shaw (2012) highlighted the difficulty in differentiating benign and malignant nodules.

According to Dolejsi (2007), image processing and visualization methods would help to reduce the workload of radiologist sifting through the huge amount of data per examination, which complicates the interpretation due to the tedious process. This implies that image processing techniques help reduce false-negative rates that are caused by the radiologist’s overlook. Another motivation would be to increase the sensitivity of detection rate, especially for smaller nodules of 1-7mm in diameter that are only detected 63% of the time (Dolejsi, 2007). Our methodology would point out the smaller nodules often left undetected by radiologist, hence contributing to our cause of early detection.

TO ADD: Dr Anuja Comments – what is the impact of looking at a ct image by a human eye vs automating it with image processing tools?

**1.3 Aims and Objectives**

The goal of our project is to obtain high quality images by using image processing techniques we learned from the class and classify whether the given image belongs to normal people or lung cancer patient.

Our objectives will be:

1. To design and develop a system that can correctly classify images into normal lungs or abnormal lungs by using MATLAB.
2. To do a thorough literature review, analyse and compare different algorithms we explore in class and find out which techniques are the most suitable.
3. To point out overall limitation, drawbacks for future improvements.
4. To collect a proper dataset for our experiments

**CHAPTER 2**

**LITERATURE REVIEW**

Several researchers have proposed and implemented different image processing techniques on detecting lung cancer using CT scan image processing. From the research paper written by Makaju et. al. (2018), the team first turned the CT scan images into grayscale and applied smoothing algorithm to remove noises from the input images in their pre-processing step. The reason they are doing this is because noises may be detected as cancer nodules and it aids in false detection. They have used median filter to remove salt and pepper noises and gaussian filter to smooth and remove speckle noise. After pre-processing the images, features like Area, Perimeter, Centrod, Diameter, Eccentricity and Mean Intensity of the pixels were extracted by using watershed segmentation algorithm. They have also used Support Vector Machine (SVM) to classify nodule as malignant or benign. Their proposed model detects the cancer with 92% accuracy.

Abdillah et. al. (2016), acquire their sample data which are a set of lung CT scan images from the VIA and ELCAP public Access Database. They then used Gabor filter to enhance the images. They implemented and evaluated three image segmentation methods such as Region Growing, Marker Controlled Watershed and Marker Controlled Watershed with Masking for analysing lung cancer. They concluded that Marker Controlled Watershed with Masking gave them the best performance in term of segmentation result and time complexity. Finally, the team have used binarization and masking to compare the CT scan images and determine the condition of lung to normal or abnormal. Binarization is the process of differentiate the colour of the pixel values into black or white. Threshold value will be compared after applying the algorithm and getting the number of black and white pixels from the image. If number of black pixels greater than the threshold value, it is said to be normal lungs, else it will be defined as abnormal lungs. They used number 17178.48 for their threshold value.



Image above is the result that has enhanced by Gabor, as shown in the paper by Abdillah et. al. (2016)

As proposed by Prasad (2013), the team have divided the work into three stages, which are Image Enhancement stage, Image Segmentation stage and Features Extraction stage. They compared Gabor filter, auto enhancement algorithm and FFT Fast Fourier Transform in the Image Enhancement stage. In Image Segmentation stage, they implemented Otsu thresholding approach and Marker-Controlled Watershed Segmentation in their research. In the last stage, they used Binarization and Masking Approach to obtain the general features of the enhanced segmented image. They found out that Gabor filter gave the best result and Marker-Controlled Watershed Segmentation is better than thresholding approach. They have used the thresholding value which is 17179.

Aggarwal et. al. (2015) have proposed to detect lung nodules in 4 stages: pre-processing of original image; segmentation of lung volume; nodule detection; feature extraction. They have used LDA (linear discriminate analysis) to cladsify the nodules. For pre-processing, median filter is used to remove noise and improve contrast. After that, they have applied optimal thresholding as it is faster and does not require initialization process when compared to deformable models such as active contours and level set based methods. Morphological closing operation is applied after that to obtain the lung area template. For nodule detection, they have selected a threshold value of T=0.44. The nodules’ feature such as area, perimeter, roundness, equivalent diameter, centroid, convex-area, eccentricity and solidity are extracted. They have proposed that roundness as a suitable parameter to differentiate blood vessels and nodules as they both have similar gray level values but blood vessels often have slender-like structures while nodules are more circular. They have also used gray level co-occurence matrix (GLCM) to obtain info on the texture of image.

Gray level co-occurrence matrix (GLCM) for feature extraction is also suggested by Tun and Khaing (2014). The GLCM function calculates the frequency of occurrence of pairs of pixels with specific values in a specified spatial relationship (Tun & Khaing, 2014). Parameters extracted are contrast, correlation, energy and entropy.

**CHAPTER 3**

**METHODOLOGY**

* 1. **Overall System Architecture**

A screenshot of a cell phone

Description automatically generated

**3.2 Image Acquisition**

In this project, we got our datasets which are the lungs CT scan image from Cancer Imaging Archive (<https://www.cancerimagingarchive.net/>). It is an open source database that provide different kind of medical images. Images we used are from the RIDER Lung CT collection that contains lung CT scans in patients with non–small cell lung cancer and from the Lung Image Database Consortium image collection (LIDC-IDRI) which is useful for identifying all lung nodules. We have collected around 400 images. However, most of the images have the cancer nodules structure which are situated close to the border. As such, we picked around 20 images that are structurally different for testing.

* 1. **Pre-processing**

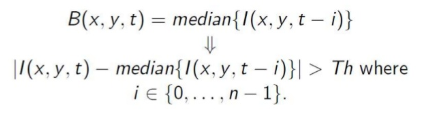
**3.3.1 Image Enhancement Algorithms**

Pre-processing plays an important role to get the accurate result in the entire process and is normally the first step to carry out. As suggested by some researchers, we have planned to convert our images into grayscale and use median filter and gaussian filter to remove noises from the CT scan images. To convert images into grayscale, we used the function in MATLAB:

*I = rgb2gray(RGB); % RGB is the original image*

Median filtering also known as rank filtering. It determines the output pixel value by the median of the neighbourhood pixels. Hence, it can remove outliers without reducing the sharpness of the image.

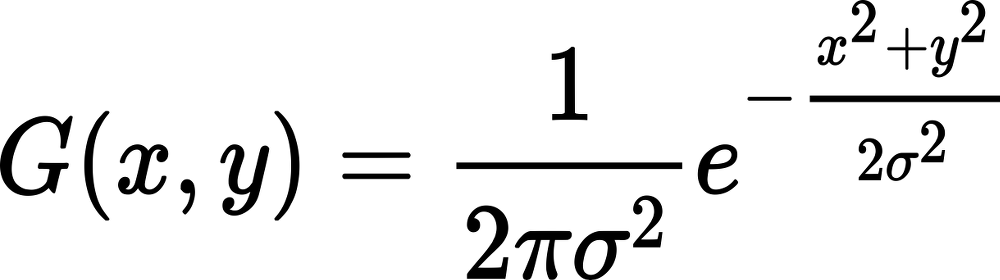
Median filter formula as shown below:



In MATLAB code, we simply used the built-in function:

*K = medfilt2(J); % J is the image with Salt and Pepper noise*

Gaussian filter is used to remove speckle noise and blur the image. Its formula shown as below:

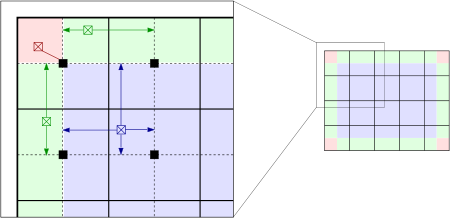


To use gaussian filter, we called built-in function as:

*Iblur = imgaussfilt(I, 2); % Image with standard deviation of 2*

**3.3.2 Image Enhancement Algorithms**

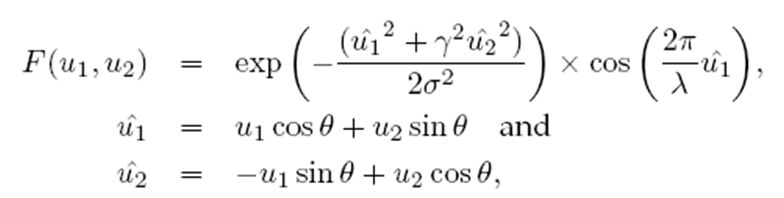
In order to get a better understanding and analysis on the images, image enhancement is necessary. This step is to improve the image quality and appearance. One of the conventional methods for image enhancement is adaptive histogram equalization as it can be used to adjust on the intensity of image pixels to enhance the image contrast. We used adaptive histogram equalization to sharpen the image features. While general histogram equalization uses the entire image to adjust the contrast, adaptive histogram equalization works on each tile and hence enhances the contrast of each small regions in the image. This method is more balanced, and it is suitable for improving the local contrast and enhancing the definitions of edges.



This is achieved by using MATLAB function:

*J = adapthisteq(I) % I is the original image*

Gabor filter is also one of the image enhancement technique that has been widely used by most of the researchers. Gabor filter is a linear filter for texture analysis. It can analyse specific frequency content in the image in a localized region. Its formula is:



To use the function, we imported an external library that we found from Mathworks. The function is written by Gao Yang and the web link to it can be found on Appendices section.

**3.3.3 Image Segmentation Algorithms**

We have used Region Based Active Contour Segmentation to obtain lung volume. We have learned this from the unit, this algorithm also has another name called Snake Algorithm. This function is to segment images whose foregrounds and backgrounds are statistically different and homogeneous. We used a library that is coded by Shawn Lankton which implemented based on the paper “Active Contours without Edges” by Chan Vese. The web link to it can be found on Appendices section.

From the segmented lungs, we can also find the possible biggest tumour location by calculating the size of potential cancer nodules.

* 1. **Feature Extraction**

**3.4.1 Binarization & Masking Algorithms**

To determine whether the input lungs CT scan image belongs to normal lungs / abnormal lungs, we used binarization and masking. In order to perform image binarization, MATLAB code:

*BW = imbinarize(I, T)*

This function creates a binary image from image I, using T thresholding. As mentioned in the research paper we read, if number of black pixels greater than threshold, it’s very likely that the input image belongs to normal lungs, otherwise, it belongs to abnormal lungs. We have used the value proposed by both Abdillah et. al. (2016) and Prasad (2013) which is 17179.

However, this method raises a high error rate to be justified as a final result as other features which does not appear as a black pixel such as tissues, organs, lymph nodes and blood vessels would also be considered as a nodule. Furthermore, different slices of chest CT images would give a different lung volume. This means that slices with larger lung volumes would most likely to be a healthy lung as it contains more black pixels.

**3.4.2 General Characteristics of the Nodule**

Properties of a nodule could be obtained from regionprops function on a binarized image containing only blobs of nodules

*labeledImage = bwlabel(binaryImage); % Label each blob so we can make measurements of it*

*blobMeasurements=regionprops(labeledImage,originalImage,'all'); % Get all the blob properties*

**Table 1:** Features that are extracted from nodules using regionprops function.

|  |  |
| --- | --- |
| **Feature** | **Methodology** |
| Perimeter | *blobPerimeter = blobMeasurements.Perimeter;* |
| Area | *blobArea = blobMeasurements.Area;*  OR  *bwarea(binarized\_image);* |
| Eccentricity | *blobEccentricity = blobMeasurements.Eccentricity;* |
| Eccentricity | *blobRoundness = blobMeasurements.Eccentricity;* When the value of F is close to 0, it is more circular. |

The properties obtained are quoted in number of pixels. As we are using DICOM images, we can get the actual dimension in millimeters (mm) by obtaining pixel spacing metadata from the DICOM header. An example of calculating the tumour area would be:

*dinfo = dicominfo('TheFileName.dcm');*

*spacing = dinfo.PixelSpacing;*

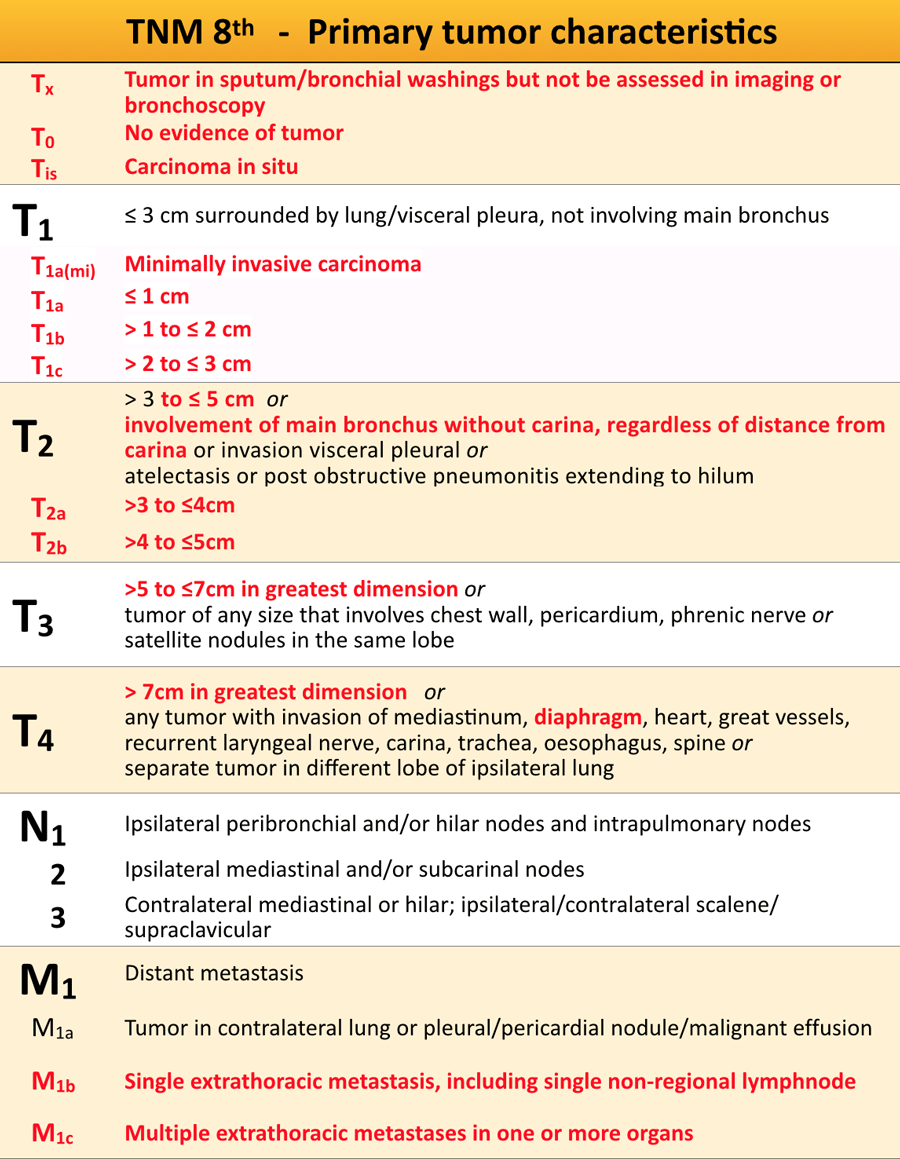
*per\_pixel\_area = spacing(1) \* spacing(2);*

*num\_white\_pixels = nnz(binarized\_nodule);*

*total\_white\_area =num\_white\_pixels \* per\_pixel\_area;*

After required metrics are acquired, we would classify the stages of tumor based on the TNM Staging System for Lung Cancer as shown in Table 2. It is a widely used and globally recognised standard for classifying cancer spread. However, we could only access the nodules based on the Tumour (T) classification from the greatest diameter of nodules obtained from regionprops. The Nodes (N) and Metastases (M) classifications are omitted as it requires more data such as the any organs or vessels the location of the nodules are being situated at which increases risk of metastases.

**Table 2:** Classifications of tumor based on the TNM Classification.



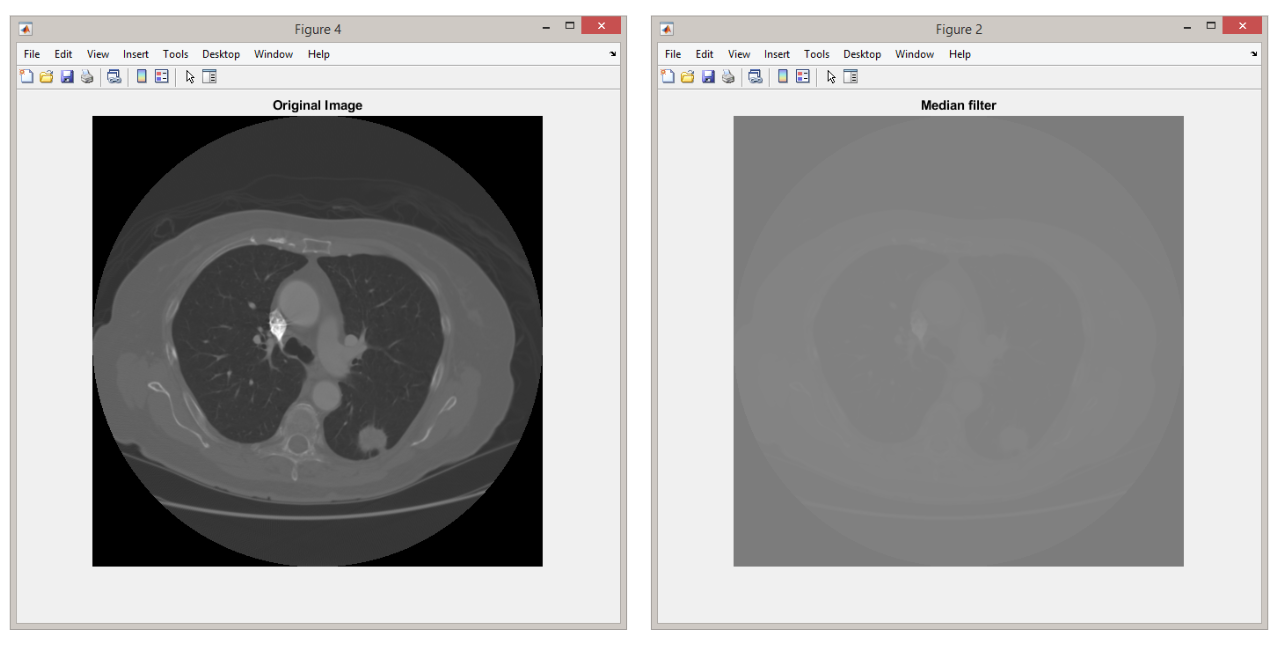
**CHAPTER 4**

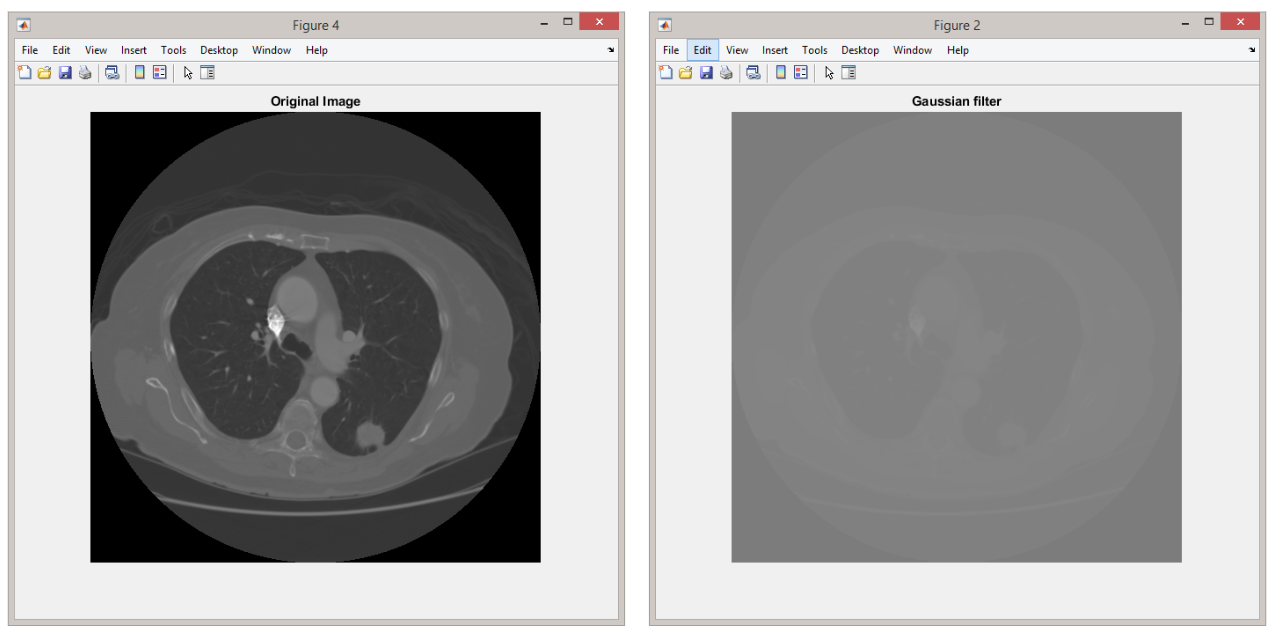
**RESULTS AND DISCUSSIONS**

**4.1 Pre-processing**

**4.1.1 Smoothing**

We realized all the images obtained from the database are in a very good quality and contain high resolution with no noise. Hence, after implementing smoothing functions, which include converting to grayscale and to apply median filter, gaussian filter, the image didn’t get improved much. Here are the results we can see after applying Median filter and Gaussian filter to the original image.

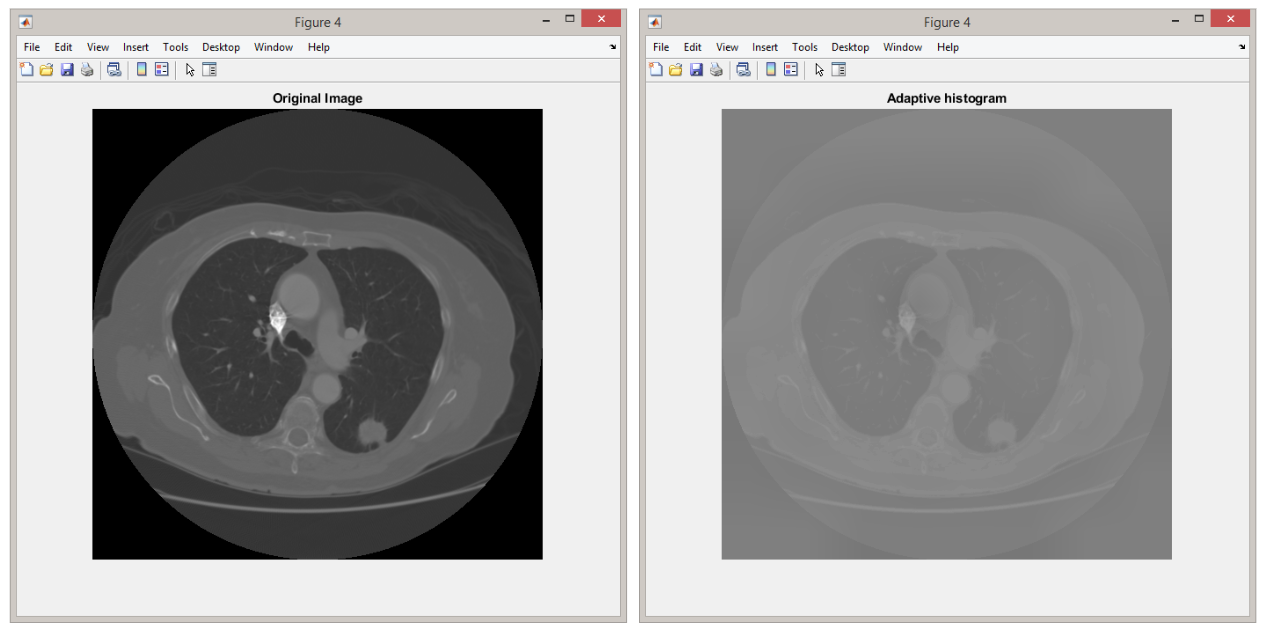
****

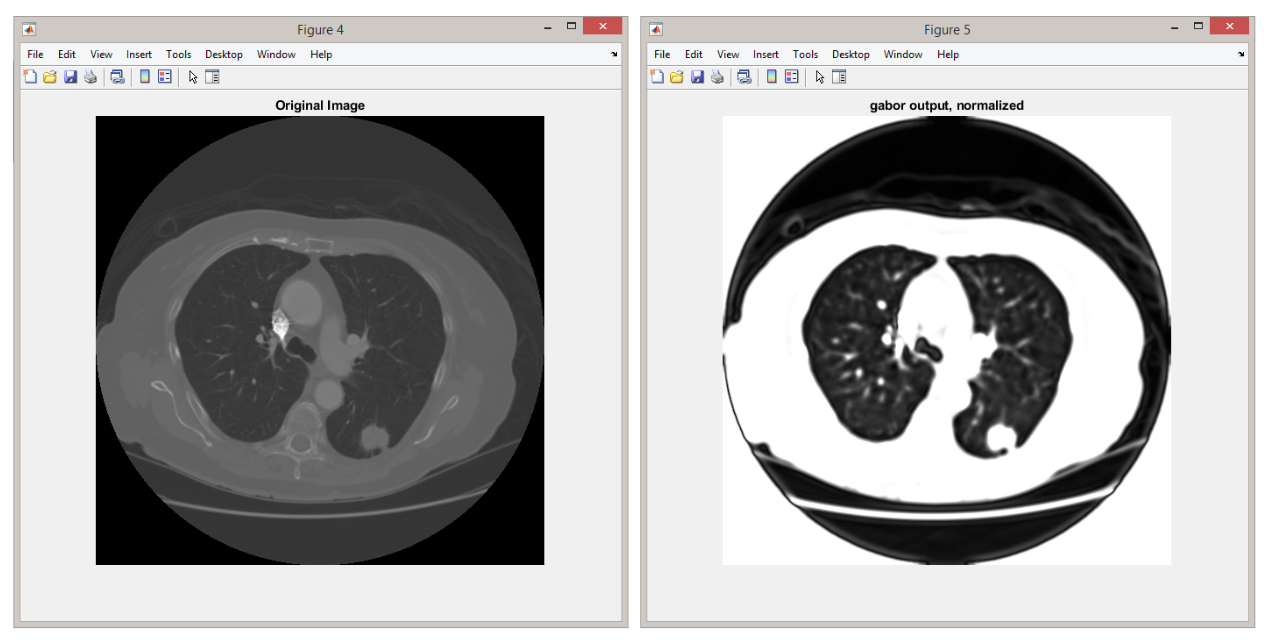
****

Therefore, smoothing was omitted as it did not make significant improvements in the image.

**4.1.2 Image Enhancement**

Next, we compared both Adaptive Histogram Equalization and Gabor filter. Below are the results.

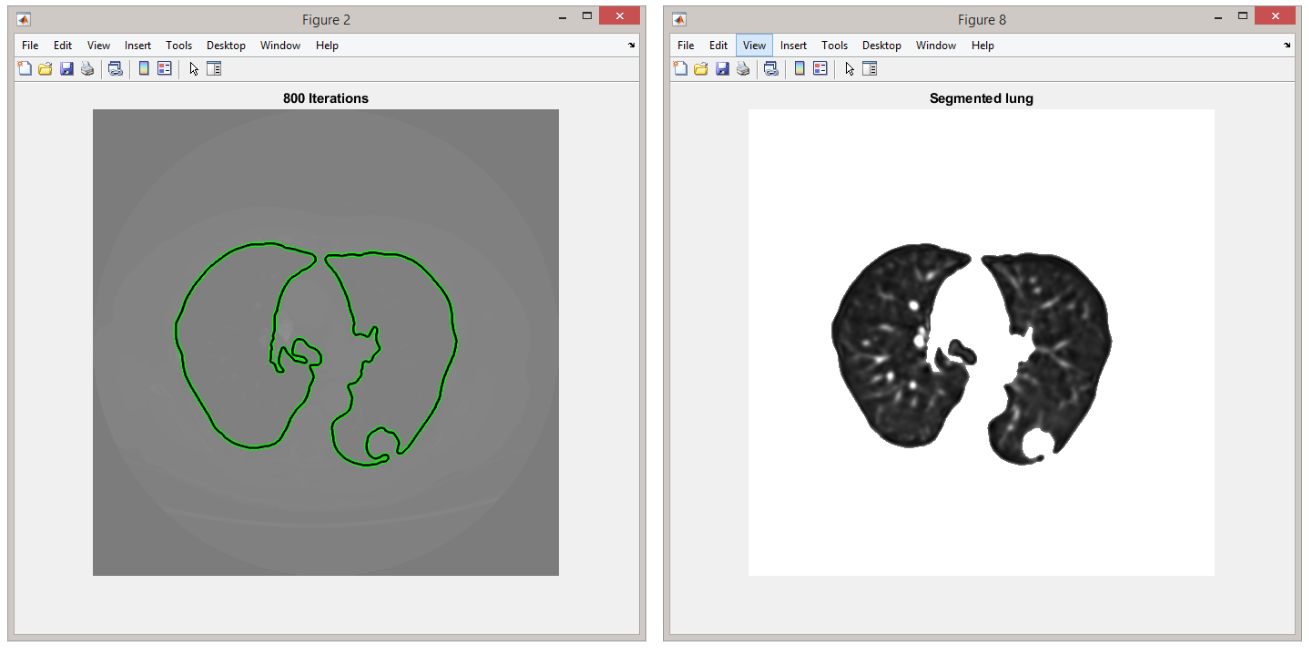




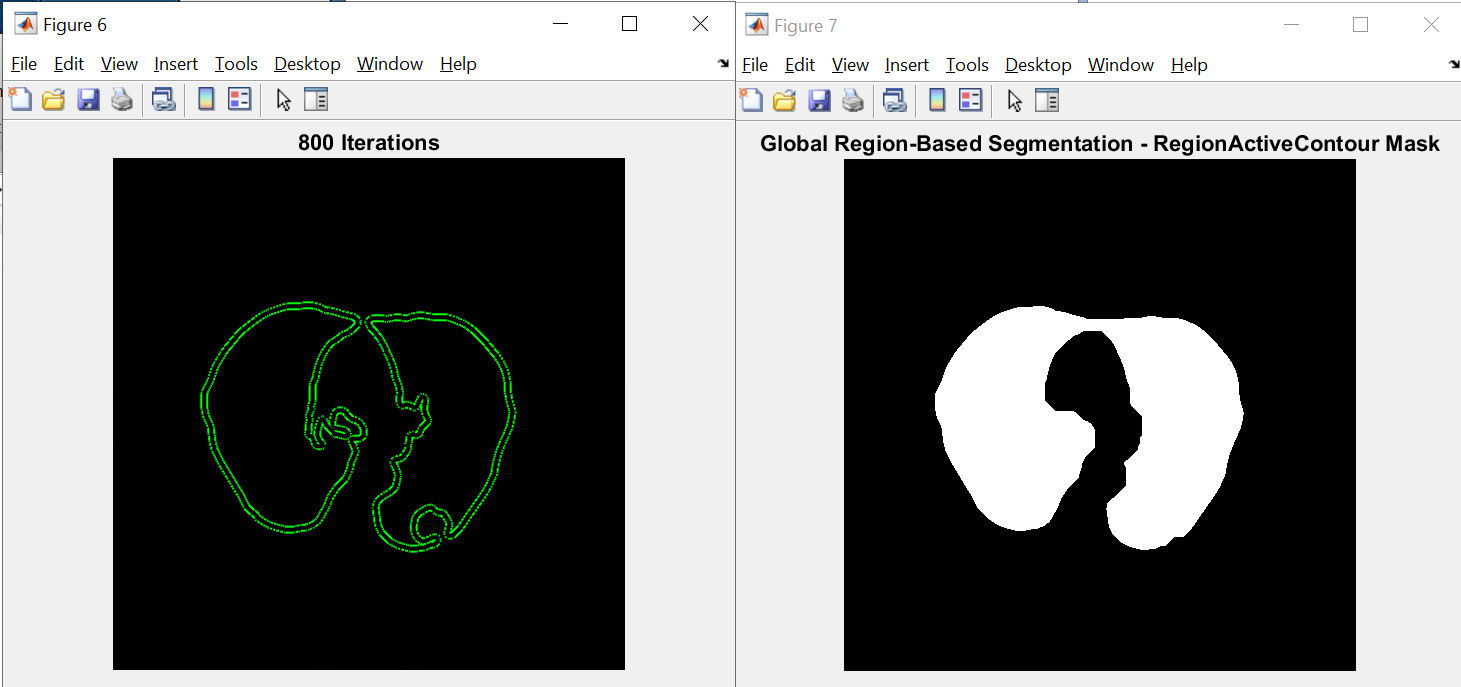
Even though the outcome from Adaptive Histogram Equalization is better than Median filter and Gaussian filter, we realized by just using Gabor filter is good enough for image enhancement stage.

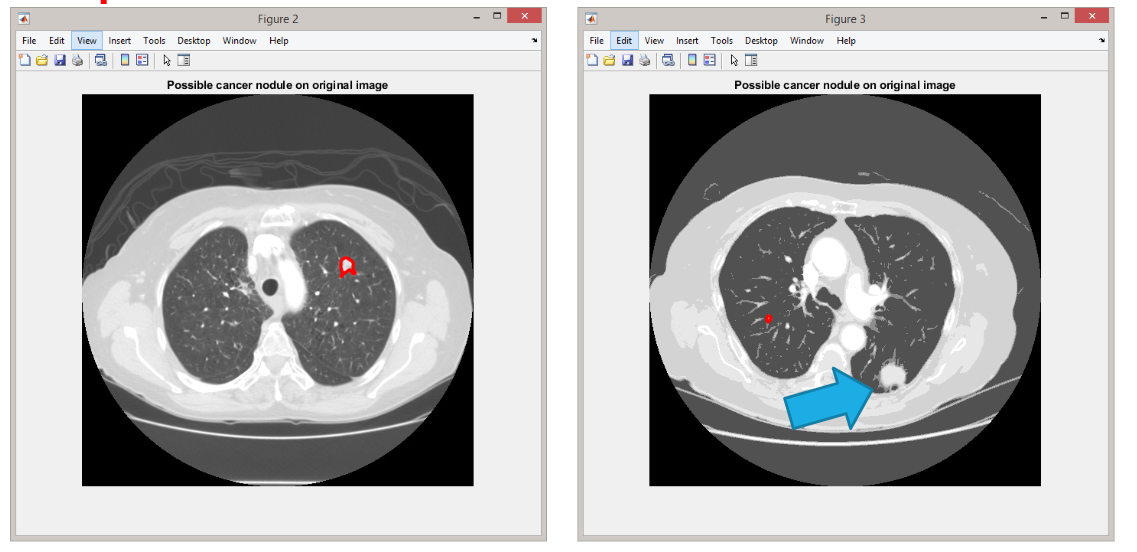
**4.1.3 Image Segmentation**

In order to obtain the lung volume, we used Region Based Active Contour algorithm. We have hardcoded the iteration value to 800 as we realized it is the most suitable value after testing several images. We couldn’t get a perfectly segmented lungs image if iteration value is set less than 800 and longer waiting time is needed if iteration value is set more than 800.



We have implemented a function to detect the possible tumour location. However, we removed this function from our main script as our system fails to detect cancer nodules that are close to the border. After segmentation, for some images we have, the tumour appears to be connected to the border and will be segmented it out. To overcome this, we have applied morphological closing using a structural element of disk with n=15 to close up the borders if there is a juxtapleural nodule, after that we fill up the holes that the juxtapleural nodule was in. Although this method segments out a perfect lung volume, the nodules were still not included during binarization.

Hence, we couldn’t find the location of the tumour for most of the images we have.



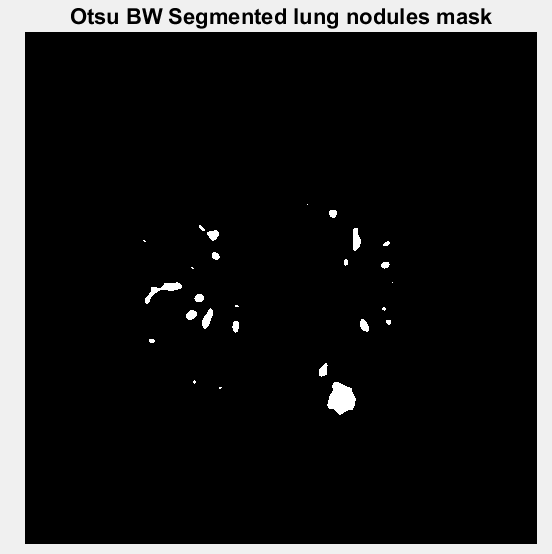
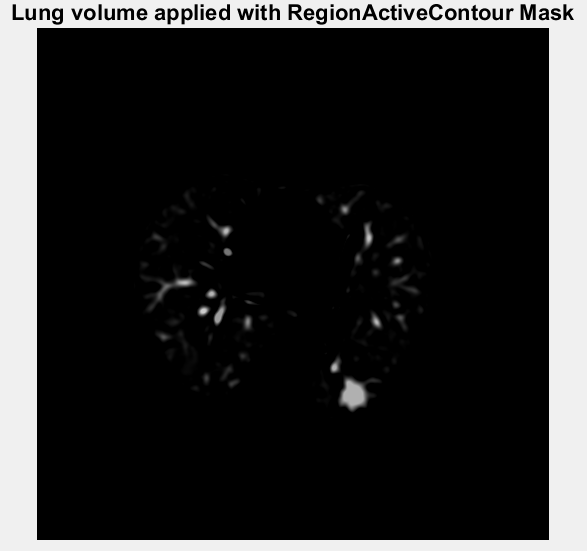
**4.2 Feature Extraction**

**4.2.1a Alternative 1: Binarization & Masking Algorithm**

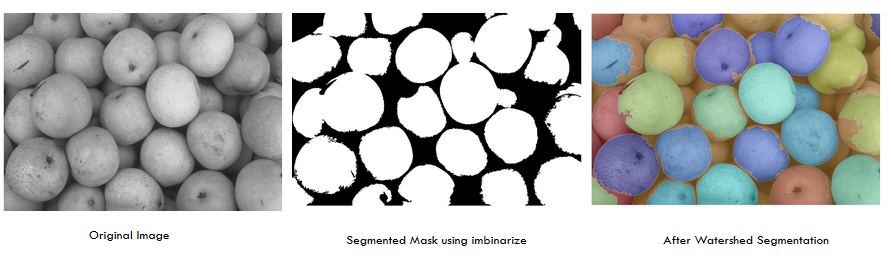
We were trying to count the number of black pixels within the lungs to classify whether the CT scan image belongs to normal people or lung cancer patient. However, by doing so, it will result us false-positive responses where blood vessels and cancer nodules are regarded as objects within the lungs which will increase the number of white pixels, hence we are unable to judge just by the number of pixels. We decided to discard this function.

**4.2.1b Alternative 2: Nodules Detection by Further Binarization**

After we have obtained the lung volume, we will segment out the nodules using imbinarize which implements Otsu Thresholding. However, this does not yield optimum results as the tumor only consist of a small area of the whole image, which have skewed the image towards lower pixel values (more black pixels). Otsu thresholding requires a bimodal pixel distribution to work well, and as shown in the image below, the the blobs are connected together when in reality they are separated.

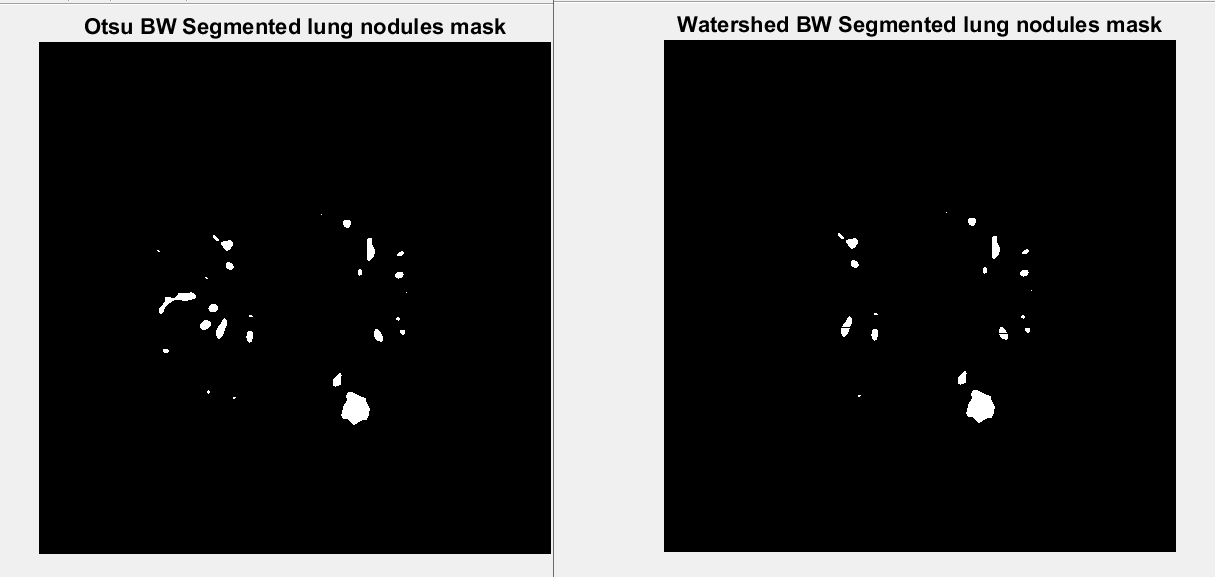


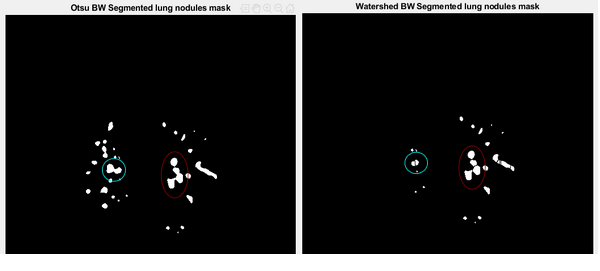
To overcome the connected blobs, we have applied watershed segmentation to separate blobs that are interconnected.



The image set above is obtained from MathWorks and successfully shown that interconnected blobs will be disjointed after watershed segmentation. However, some blobs would be considered as background (orange) after the process.

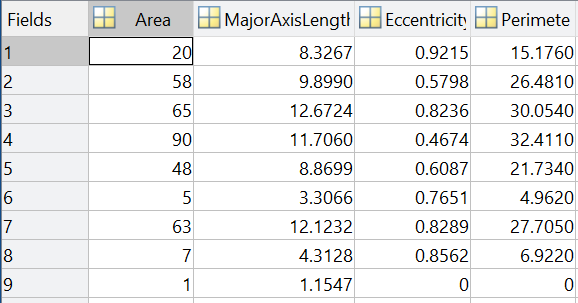
A few examples of successful watershed segmentation to obtain disjointed tumors are shown below. It is important to keep nodule blobs separated as we will be using eccentricity feature from regionprops to filter out elongated blobs which are likely to be blood vessels. Having interconnected blobs will disguise tumors as blood vessels.



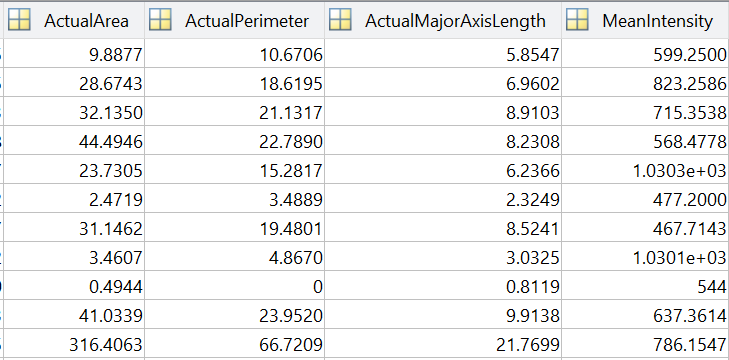


**4.2.2 Feature Extraction**

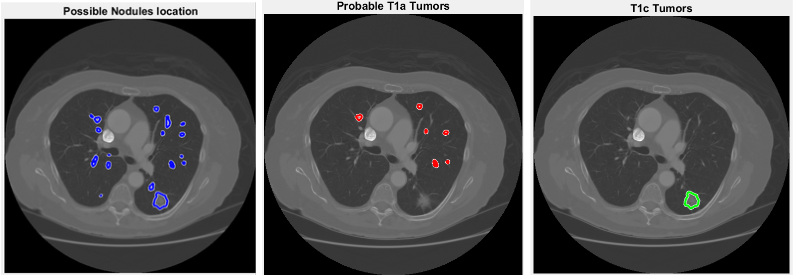
After the nodules blobs are obtained, their properties can be obtained using regionprops. A few parameters that we have used are area, perimeter, major axis length, eccentricity, mean pixel intensity of blob.



However, the data obtained are quoted in number of pixels and we need actual dimensions in millimeter in order to classify them based on TNM 8th edition. Hence, we add extra info on the actual dimensions based on DICOM header’s pixel spacing metadata.

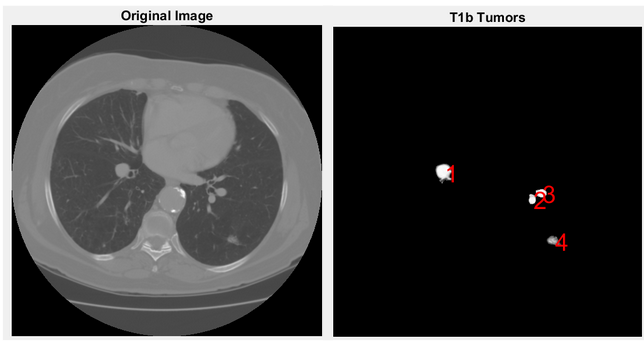


After the blobs dimensions and required parameter are added, we classify them based on TNM 8th edition where the tumor major axis length (greatest diameter) is considered.

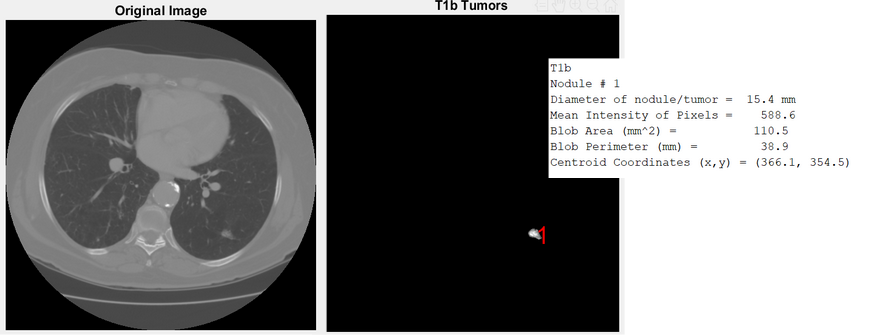


Although the official TNM classification considers any blob smaller than 1cm to be stage T1a, we will only consider those between 3mm to 1cm, any nodules smaller than 3mm are not considered as T1a in our system as it raises false positives. However, they would still be shown in the possible nodules location image if the radiologist would like to reexamine.

We have also refined our filter to remove blood vessels by removing elongated blobs using the eccentricity feature. After analysing a couple number of images, we have also found out that blood vessels have a higher pixel intensity than nodules. Hence, an extra filter predicate is entered into the conditions.



Before refining the filter, only the blobs diameter is considered. Blobs 1,2,3 are blood vessels after verification using a 3D volume viewer. Only Blob 4 is the true tumor.



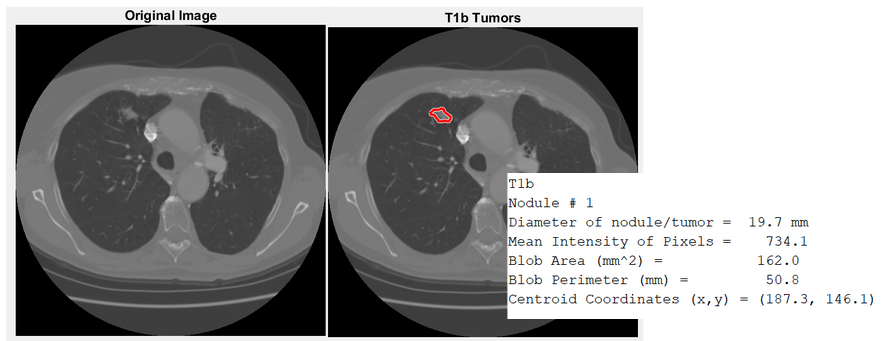
We have added eccentricity and mean pixel intensity as parameters for the conditions to filter out blood vessels and it has worked successfully.

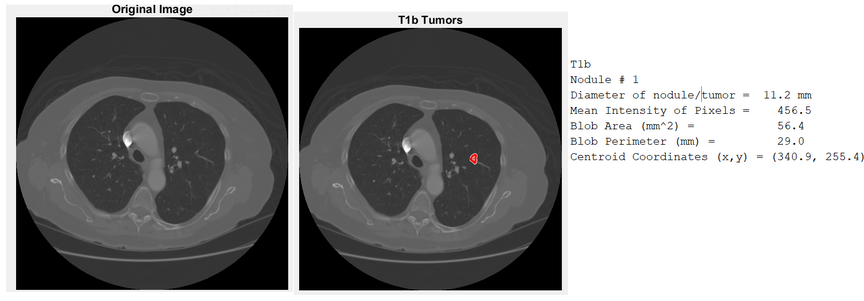
**4.3 Showing Results**

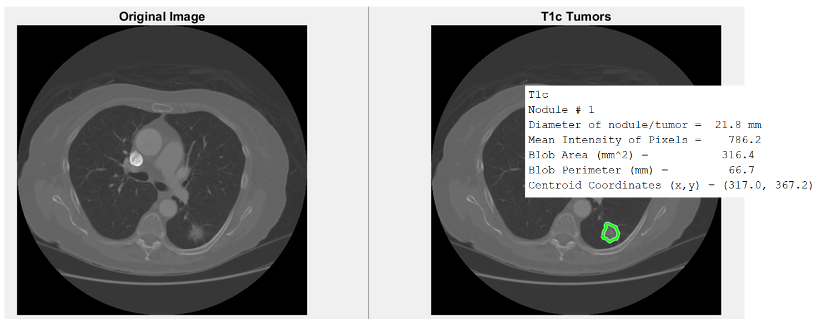
The command window will show that it is a healthy lung if there are no nodules detected or there are no nodules other than T1a stage detected. This is because T1a nodules are often to be diagnosed as benign, however, we would prompt the user for further diagnosis showing all probable T1a nodules and their statistics. On the other hand, if there are any tumors after and including T1b detected, we would output it to show that tumors are detected.

Some examples of our result:









**CHAPTER 5**

**CONCLUSION AND RECOMMENDATIONS**

This project is challenging for us as we do not have the medical background and knowledge to determine cancer nodules and tumour. Lot of research and studies are needed for us to understand the topic better. In our system, although we are able to perform image processing techniques to achieve our main purpose, which is to obtain nodules characteristics and classify the images into normal/abnormal category, we cannot detect subpleural nodules for some of the lung cancer images we have, resulted in us discarding this method. For some functions such as Gabor filter and Region Based Active Contour segmentation algorithm, we have used the coding work that have developed by other researchers as we do not have the knowledge to write the functions from scratch.

Small, ground-glass nodules might be undetected due to difficult segmentation, poorly defined borders and low contrast from surrounding pulmonary tissue (Firmino et. al., 2014).

Another limitation to our implementation would be only a single CT slice is processed. As there are no info given in the dataset about which blobs are considered as a tumor, we have to use a 3D slice viewer to differentiate a tumor from other white blobs of similar intensity such as blood vessels. This could be improved by using multiple CT slices or a detection model based on 3D lung volume.

In the future, we can also train a neural network to automatically classify images that have the same structure (tumour location, size and such). Hopefully for this refinement will be able to help classify images correctly. Another improvement to our system will be to reduce the time needed to produce faster results.

**REFERENCES**

Abdillah, B., Bustamam, A., & Sarwinda, D. (2016). Image processing based detection of lung cancer on CT scan images. The Asian Mathematical Conference 2016 (AMC). doi:10.1088/1742-6596/893/1/012063

Aggarwal, T., Furqan, A., & Kalra, K. (2015). Feature extraction and LDA based classification of lung nodules in chest CT scan images. 2015 International Conference on Advances in Computing, Communications and Informatics (ICACCI). doi:10.1109/icacci.2015.7275773

American Cancer Society. (2019). Lung Cancer Prevention and Early Detection. Retrieved from: <https://www.cancer.org/cancer/lung-cancer/prevention-and-early-detection.html>

Dolejsi, M. (2007). Detection of Pulmonary Nodules from CT Scans. *Research Reports of CMP, Czech Technical University in Prague, No. 5, 2007.* Retrieved June 03, 2019, from https://pdfs.semanticscholar.org/aabe/7dcff29cdce8869d308232cb6a67746d83b6.pdf

Firmino, M., Morais, A. H., Mendoça, R. M., Dantas, M. R., Hekis, H. R., & Valentim, R. (2014). Computer-aided detection system for lung cancer in computed tomography scans: review and future prospects. *Biomedical engineering online*, *13*, 41. doi:10.1186/1475-925X-13-41

Hollings, N. & Shaw, P. (2002). Diagnostic imaging of lung cancer. European Respiratory Journal, 19, 722-742. <https://doi.org/10.1183/09031936.02.00280002>

Ignatious, S., & Joseph, R. (2015). Computer aided lung cancer detection system. 2015 Global Conference On Communication Technologies (GCCT), DOI: 10.1109/GCCT.2015.7342723.

Makaju, S., Prasad, P. W. C., Alsadoon, A., Singh, A. K., Elchouemi, A. (2018). Lung Cancer Detection using CT Scan Images. Procedia Computer Science, 125, pp. 107-114.

Peikert, T., Duan, F., Rajagopalan, S., Karwoski, R. A., Clay, R., ... , Maldonado F. (2018). Novel high-resolution computed tomography-based radiomic classifier for screen-identified pulmonary nodules in the National Lung Screening Trial. PLOS ONE 13(10): e0205311. <https://doi.org/10.1371/journal.pone.0205311>

Prasad, D. V. R. (2013). Lung Cancer Detection Using Image Processing Techniques. International Journal of Latest Trends in Engineering and Technology (IJLTET).

Tun, K. M. M., & Khaing, A. S. (2014). Feature Extraction and Classification of Lung Cancer Nodule using Image Processing Techniques. International Journal of Engineering Research & Technology (IJERT), 3(3).

## **APPENDICES**

APPENDIX A: Source code (only the main script\*)

\*Please refer to the source code submission link for other functions

clc;

clear;

close all;

%% Eric open folder for 3d volume - not used

% fileFolder = fullfile(pwd, 'LIDC-IDRI-0001','01-01-2000-30178','3000566-03192');

% files = dir(fullfile(fileFolder, '\*.dcm'));%specify data file diectory

% fileNames = {files.name};

%% Read single DICOM Image

dInfo = dicominfo(fullfile('Patient1','000123.dcm'));

%dReference = imread('abnormal1.jpg');

% dImage = uint8(dicomread(dInfo));

dImage = dicomread(dInfo);

%img\_in = imhistmatch(dImage, dReference);

%%

img\_in = dImage;

figure, imshow(img\_in, []), title('Original Image');

%extract size for planeXY, XZ, YZ from meta data

voxel\_size = [dInfo.PixelSpacing; dInfo.SliceThickness];

%% Smoothing - Apply median filter

% img\_in = medfilt2(dImage);

%% Smoothing - Gaussian filter

% img\_in = imgaussfilt(img\_in,2);

% figure, imshow(img\_in, []), title('Gaussian filtering');

%% Smoothing - Anisotropic diffusion filtering of images - preserves the sharpness of edges better than Gaussian blurring.

% img\_in\_2 = imdiffusefilt(dImage);

% figure, imshow(img\_in\_2, []), title('Anisotropic diffusion filtering');

%% Adaptive histogram - Not using\*

% img\_in = adapthisteq(img\_in);

%% Image Enhancement - Gabor filter

lambda = 25;

theta = 0;

bw = 3;

psi = [0 0];

gamma = 2;

N = 4;

img\_out = zeros(size(img\_in,1), size(img\_in,2), N);

for n=1:N

gb = gabor\_fn(bw,gamma,psi(1),lambda,theta)...

+ gabor\_fn(bw,gamma,psi(2),lambda,theta);

img\_out(:,:,n) = imfilter(img\_in, gb, 'symmetric');

theta = theta + pi/4;

end

img\_out\_disp = sum(abs(img\_out).^2, 3).^0.5;

img\_out\_disp = img\_out\_disp./max(img\_out\_disp(:));

figure, imshow(img\_out\_disp), title('gabor output, normalized');

%% Active Contour using masking to get Lung Volume

Ir = img\_out\_disp;

se = strel('disk', 20);

Ie = imerode(Ir, se);

Iobr = imreconstruct(Ie, Ir);

Iobrd = imdilate(Iobr, se);

Iobrcbr = imreconstruct(imcomplement(Iobrd), imcomplement(Iobr));

Iobrcbr = imcomplement(Iobrcbr);

bw = imbinarize(Iobrcbr, graythresh(Iobr));

figure, imshow(bw), title('Binarized')

Ia=bw;

m = zeros(size(Ia,1),size(Ia,2));

m(200:320,95:180) = 1;

m(186:321,348:410) = 1;

seg = region\_seg(Ia, m, 800); % Run segmentation with 800 iteration

figure, imshow(seg); title('Global Region-Based Segmentation - RegionActiveContour Mask')

%% Binarization for image classification (masked Image)

tmp=ones(512,512);

black=0;

for i=1:512

for j=1:512

if seg(i,j)==1

tmp(i,j)=img\_out\_disp(i,j);

if tmp(i,j)<=0.12

black=black+1;

end

else

tmp(i,j)=1;

end

end

end

tmp = imclearborder(tmp);

figure, imshow(tmp), title('Lung volume applied with RegionActiveContour Mask')

%% Eric to extract nodules using Watershed Segmentation

% https://www.mathworks.com/company/newsletters/articles/the-watershed-transform-strategies-for-image-segmentation.html

% This is marker controlled watershed using masking to get nodules

I\_eq = adapthisteq(tmp);

% figure, imhist(tmp), title('Histogram of lung volume image');

% figure, imhist(I\_eq), title('Histogram of lung volume image after equalization');

tmpBW = imbinarize(tmp, graythresh(I\_eq)); % Built-in Otsu Thresholding to get nodules - blobs will clump together

figure, imshow(tmpBW), title('Otsu BW Segmented lung nodules mask')

watershed\_nodule = watershedTransform(tmpBW); % Watershed to separate out blobs (obtained from MathWorks)

watershed\_nodule = imbinarize(watershed\_nodule);

figure, imshow(watershed\_nodule), title('Watershed BW Segmented lung nodules mask');

maskedNodule = img\_in;

maskedNodule(~watershed\_nodule) = 0;

figure,imshow(maskedNodule, []), title('Nodules using WatershedSegmentation');

%% Visualize the nodules on original image

% Show tumour boundaries

boundary = bwboundaries(watershed\_nodule);

figure, imshow(dImage, []), title('Possible Nodules location')

hold on

visboundaries(boundary, 'Color', 'b');

%% Tumor parameters to reduce false positives - hard coded (does not stage tumor based on TNM)

% holesAccurate = bwareafilt(watershed\_nodule, [50 1000]); % malignant tumour are usually larger than 50

% boundary = bwboundaries(holesAccurate);

% figure, imshow(dImage, []), title('Possible Nodules location (area considered)');

% hold on

% visboundaries(boundary, 'Color', 'r');

%% Tumor parameters to reduce false positives

holesAccurate = bwareafilt(watershed\_nodule, [50 1000]); % malignant tumour are usually larger than 50

labeledImage = bwlabel(holesAccurate); % Label each blob so we can make measurements of it

blobMeasurements = regionprops(labeledImage,dImage,'all'); % Get all the blob properties

%% Tumor parameters to reduce false positives - hard coded - more parameters involved

% Binarized image of nodules using watershed segmentation

BW = watershed\_nodule;

% Find connected components (blobs)

cc = bwconncomp(BW);

% Find statistics/properties of each blob

stats = regionprops(cc, 'all');

%% Create new column (dervied) in regionprops (stats) structure array

% DICOM header - extract pixel size for planeXY, XZ, YZ from DICOM meta data

pixel\_spacing = dInfo.PixelSpacing;

per\_pixel\_area = pixel\_spacing(1)\*pixel\_spacing(2);

% Area based on per pixel area

actual\_area = num2cell([stats.Area]\*[per\_pixel\_area]);

[stats.ActualArea] = actual\_area{:};

% Actual perimeter based on pixel spacing

actual\_perimeter = num2cell([stats.Perimeter]\*[pixel\_spacing(1)]);

[stats.ActualPerimeter] = actual\_perimeter{:};

% Actual major axis based on pixel spacing

actual\_diameter = num2cell([stats.MajorAxisLength]\*[pixel\_spacing(1)]);

[stats.ActualMajorAxisLength] = actual\_diameter{:};

% Mean Gray Level Intensity of nodule

for k = 1 : length(stats) % Loop through all blobs.

thisBlobsPixels = stats(k).PixelIdxList; % Get list of pixels in current blob.

MeanIntensity = mean(dImage(thisBlobsPixels)); % Find mean intensity (in original image!)

stats(k).MeanIntensity = MeanIntensity;

end

%% Find desired Parameters

% Get index in regionprops stucture array that satisfy property according

% to TNM 8th edition

% Stage t1a

if (length(stats) ~= 0)

idx = find([stats.ActualMajorAxisLength] > 3 & [stats.ActualMajorAxisLength] <= 10 & [stats.Eccentricity] < 0.8 & [stats.MeanIntensity] < 800);

t1a = ismember(labelmatrix(cc), idx);

% figure, imshow(t1a), title('Stage t1a mask - Discard due to high false positive');

t1a\_stats = stats(idx);

% Stage t1b

idx = find([stats.ActualMajorAxisLength] > 10 & [stats.ActualMajorAxisLength] <= 20 & [stats.Eccentricity] < 0.8 & [stats.MeanIntensity] < 800);

t1b = ismember(labelmatrix(cc), idx);

% figure, imshow(t1b), title('Stage t1b mask');

t1b\_stats = stats(idx);

% Stage t1c

idx = find([stats.ActualMajorAxisLength] > 20 & [stats.ActualMajorAxisLength] <= 30 & [stats.Eccentricity] < 0.8 & [stats.MeanIntensity] < 800);

t1c = ismember(labelmatrix(cc), idx);

% figure, imshow(t1c), title('Stage t1c mask');

t1c\_stats = stats(idx);

% Stage 2

idx = find([stats.ActualMajorAxisLength] > 30 & [stats.ActualMajorAxisLength] <= 50 & [stats.Eccentricity] < 0.8 & [stats.MeanIntensity] < 800);

t2 = ismember(labelmatrix(cc), idx);

% figure, imshow(t2), title('Stage t2 mask');

t2\_stats = stats(idx);

% Stage 3

idx = find([stats.ActualMajorAxisLength] > 50 & [stats.Eccentricity] < 0.8 & [stats.MeanIntensity] < 800);

t3 = ismember(labelmatrix(cc), idx);

% figure, imshow(t3), title('Stage t3 mask');

t3\_stats = stats(idx);

end

%% Display Result on Command Window

% Discard stage t1a due to high false positive rate

% If there are no tumors > stage t1a, we consider as normal

if length(stats) == 0

('normal lung')

else

if (length(t1b\_stats) + length(t1c\_stats) + length(t2\_stats) == 0)

('normal lung')

% Visualize output

fprintf('\nEarly Detection saves lives! Possible nodules:\n\n');

if length(t1a\_stats) ~= 0

t1a\_holes = bwlabel(t1a);

boundary = bwboundaries(t1a\_holes);

figure, imshow(dImage, []), title('Probable T1a Tumors');

hold on

visboundaries(boundary, 'Color', 'r');

maskedImageT1a = dImage;

maskedImageT1a(~t1a\_holes) = 0;

disp('T1a\n');

figure, imshow(maskedImageT1a, []), title(' Probable T1a Tumors');

showNoduleStats(t1a\_stats);

end

else

('tumor detected')

% Visualize output of t1a nodules

if length(t1a\_stats) ~= 0

t1a\_holes = bwlabel(t1a);

boundary = bwboundaries(t1a\_holes);

figure, imshow(dImage, []), title('Probable T1a Tumors');

hold on

visboundaries(boundary, 'Color', 'r');

maskedImageT1a = dImage;

maskedImageT1a(~t1a\_holes) = 0;

disp('T1a');

figure, imshow(maskedImageT1a, []), title(' Probable T1a Tumors');

showNoduleStats(t1a\_stats);

end

% Visualize output of t1b nodules

if length(t1b\_stats) ~= 0

t1b\_holes = bwlabel(t1b);

boundary = bwboundaries(t1b\_holes);

figure, imshow(dImage, []), title('T1b Tumors');

hold on

visboundaries(boundary, 'Color', 'r');

maskedImageT1b = dImage;

maskedImageT1b(~t1b\_holes) = 0;

disp('T1b');

figure, imshow(maskedImageT1b, []), title('T1b Tumors');

showNoduleStats(t1b\_stats);

end

% Visualize output

if length(t1c\_stats) ~= 0

t1c\_holes = bwlabel(t1c);

boundary = bwboundaries(t1c\_holes);

imshow(dImage, []), title('T1c Tumors');

hold on

visboundaries(boundary, 'Color', 'g');

maskedImageT1c = dImage;

maskedImageT1c(~t1c\_holes) = 0;

figure, imshow(maskedImageT1c, []), title('T1c Tumors');

disp('T1c');

showNoduleStats(t1c\_stats);

end

% Visualize output

if length(t2\_stats) ~= 0

t2\_holes = bwlabel(t2);

boundary = bwboundaries(t2\_holes);

figure, imshow(dImage, []), title('T2 Tumors');

hold on

visboundaries(boundary, 'Color', 'c');

maskedImageT2 = dImage;

maskedImageT2(~t2\_holes) = 0;

figure, imshow(maskedImageT2, []), title('T2 Tumors');

disp('T2');

showNoduleStats(t2\_stats)

end

% Visualize output

if length(t3\_stats) ~= 0

t3\_holes = bwlabel(t3);

boundary = bwboundaries(t3\_holes);

figure, imshow(dImage, []), title('T3 Tumors');

hold on

visboundaries(boundary, 'Color', 'y');

maskedImageT3 = dImage;

maskedImageT3(~t3\_holes) = 0;

figure, imshow(maskedImageT3, []), title('T3 Tumors');

disp('T3');

showNoduleStats(t3\_stats)

end

end

end

APPENDIX B: Others

Gabor filter by Gao Yang (Mathworks):

<https://www.mathworks.com/matlabcentral/fileexchange/23253-gabor-filter?s_tid=srchtitle>

Active Contour Segmentation by Shawn Lankton (Mathworks):

<https://www.mathworks.com/matlabcentral/fileexchange/19567-active-contour-segmentation>

Lung Cancer Detector by Abdillah, B., Bustamam, A., & Sarwinda, D. (2016) on paper: Image processing based detection of lung cancer on CT scan images. (Github)

<https://github.com/bariqi/Image-Processing-for-Lung-Cancer-Classification>