Brainomix Challenge

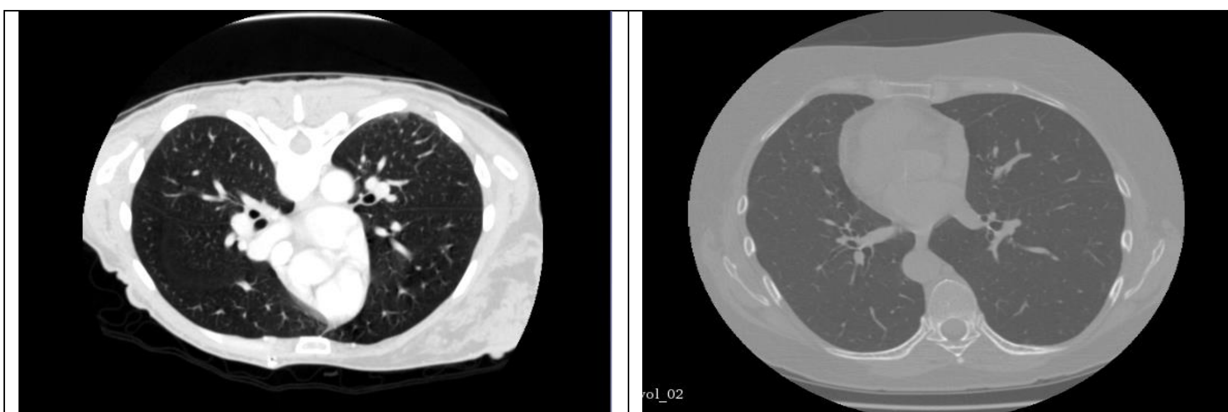
Introduction

The Brainomix challenge was to calculate the lung volume, the vessel-lung ratio of all scans and classify the 9 images. The images were 3D CT scans encoded with Hounsfield units (HU). Software like 3D Slicer and MATLAB was used for easier image analysis and then Python programming language was chosen as a platform to solve the problem.

Methodology

Lung-Vessel Segmentation

The images were viewed in 3D Slicer to understand the difficulties of using this dataset. It was found that there are two distinct types of images. This difference is based on the foreground and background differences, as shown in Figure 1.



a) Type 1	b) Type 2

Figure 1: Distinct types of scans

This may not seem to be a dramatic difference but when it comes to thresholding-based segmentation, the accuracy of mask estimation was hugely impacted. The normalisation of the voxel intensities did not sort this challenge, so finding the right threshold values was the solution. A MATLAB program was written to perform the histogram analysis that helped in finding the optimal threshold values for the body, lung, and vessel segmentation.

After finding the threshold values for body, lung, and vessels, corresponding masks were obtained. These masks were then passed onto several morphological operations to condition. The optimal sequence of these operations was identified in the MATLAB platform and then translated into Python scripts, whose sample result is shown in Figure 2.

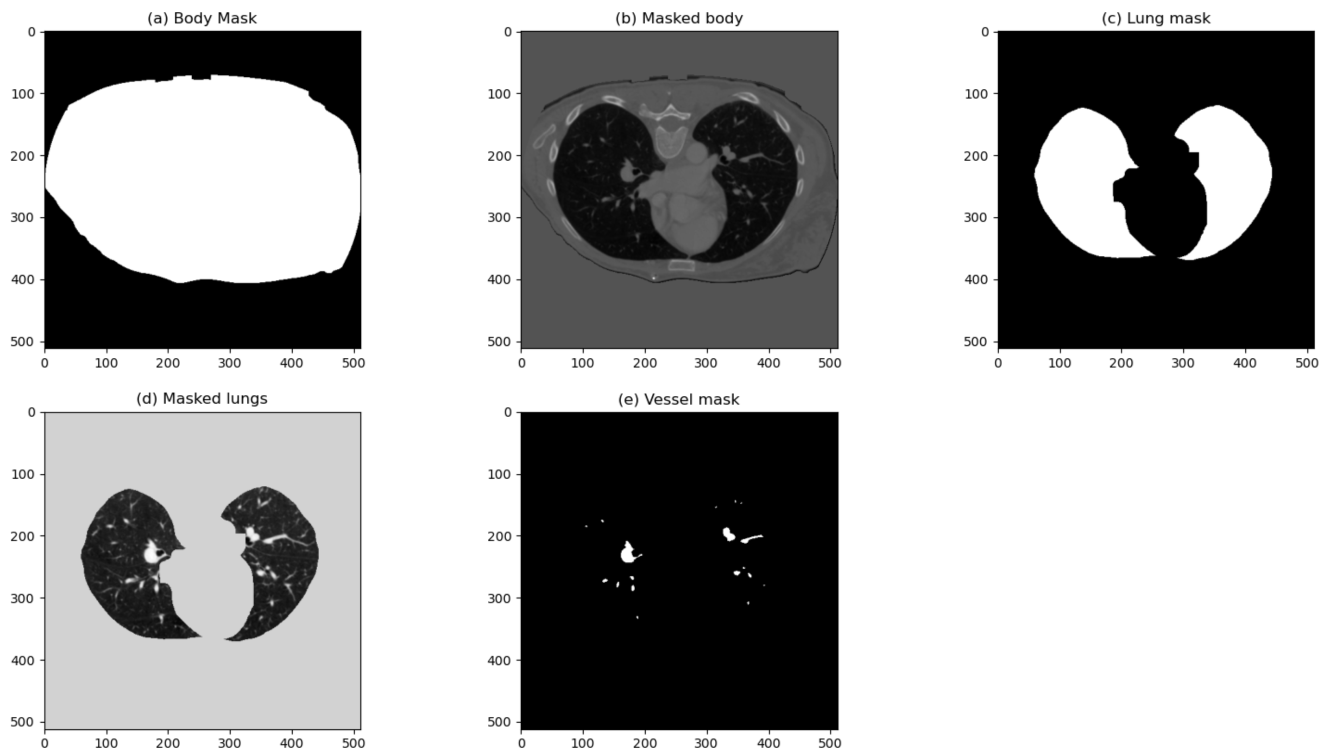


Figure 2: Images with their corresponding masks

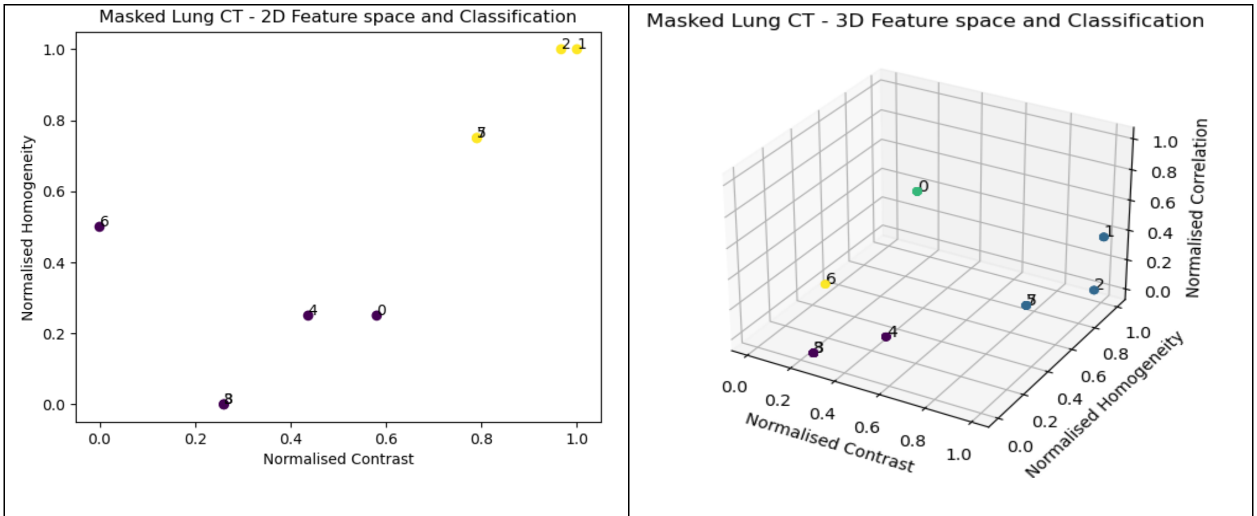
In the Python script, several packages were used to perform, i.e., SimpleITK was used for reading and writing images, Numpy was used for array processing, OpenCV was used for most morphological operations except the flood-filling, for which Scipy was used. Once the respective masks are obtained the volume of the segments was calculated using the Numpy function. Since the images did not have other DICOM metadata such as Pixel Spacing, the unit of the lung volume is in the number of pixels rather than metric units as listed in Table 1. Table 1 indicates poor lung health in many images; however, this could be due to the threshold that was chosen for segmentation. The voxel intensities of the brightest vessels were chosen as the threshold.

Table 1 Lung and vessel volume of images

Data	Lung Volume	Vessel Volume	Vessel-Lung %
Image1	2.47E+06	112461	4.55718
Image2	2.14E+06	94757	4.42222
Image3	9.13E+06	3.94E+06	43.2016
Image4	8.04E+06	2.42E+06	30.1455
Image5	1.07E+06	25195	2.35935
Image6	2.55E+06	376955	14.7539
Image7	1.01E+06	64918	6.44209
Image8	1.28E+06	308046	24.1361
Image9	726879	42545	5.85311

Image Classification

Texture features of the lungs (after masking) from all images were extracted by calculating the gray level co-occurrence matrix using the Skimage package. Then for each image, a set of features was computed that included contrast, dissimilarity, homogeneity, energy, and correlation. Then after analysis, three features were chosen contrast, homogeneity, and correlation. A built-in k-means clustering technique was applied to the 2D and 3D feature space and illustrated in Figure 3. In the 2D space, it appears there are two categories of lung pictures, but datapoint 6 is a bit far off from the cluster. Therefore, when the third feature is added then we can see four different groups, which need more investigation. From these clusters, we could see that the texture of the lungs from the images classifies the whole set of 9 images into two groups say {{1,2,5,7}, {0,3,4,6,8}} in 2D feature space and into four groups say {{0}, {1,2,5,7}, {6}, {3,4,8}}.



(a) (b)

Figure 3 k-means clustering based classification of lungs using textural features

Conclusion

To summarise, lung and vessel segmentation was performed on all 9 images. Then texture features were extracted to form a 2D and 3D feature sapce. Upon those features, k-means clustering was applied to classify the images. The software used for image analysis are 3D Slicer and MATLAB. A Python script was written to execute the image processing, segmentation, morphological operations, and classification.

<u>Instruction to run the code</u>: Make sure the data folder and the script are in the same location, since the script searches for the 'Images' folder in its current working directory.