PLAKSHA UNIVERSITY

RO4101 – DIAGNOSTIC TECHNOLOGIES

Lab Session #3

Learning Outcomes:
In continuation with Lab sessions 1 and 2, this lab exercise aims to introduce you to more image-processing functions including image thresholding, image enhancement, and noise removal that are crucial for extracting meaningful features from medical images.

By the end of this exercise, you will have the knowledge and tools to perform the following tasks in PYTHON:

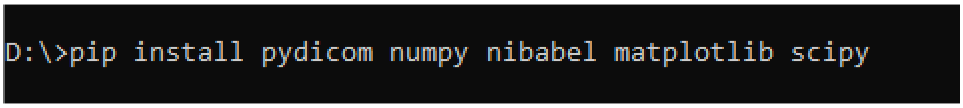
- Learn techniques to eliminate or minimize background noise with the help of thresholding.
- Apply image enhancement techniques including scaling and histogram equalization on X-ray based image modalities to enhance the imaging features.

Marks Allocated: 2.5% (25 points)

Computing Tools Required:

- **Python 3.x**
- **Libraries:** pydicom, numpy, nibabel, matplotlib, scipy, opencv-python (for image processing)

You can install the necessary libraries using pip by running the following command in the terminal:



```
D:\>pip install pydicom numpy nibabel matplotlib scipy
```

Instructions:

Download the data:
https://drive.google.com/file/d/1M8Lol4hkdXQ3Cy75CtMb1FW9CV9YUAS6/view?usp=sharing

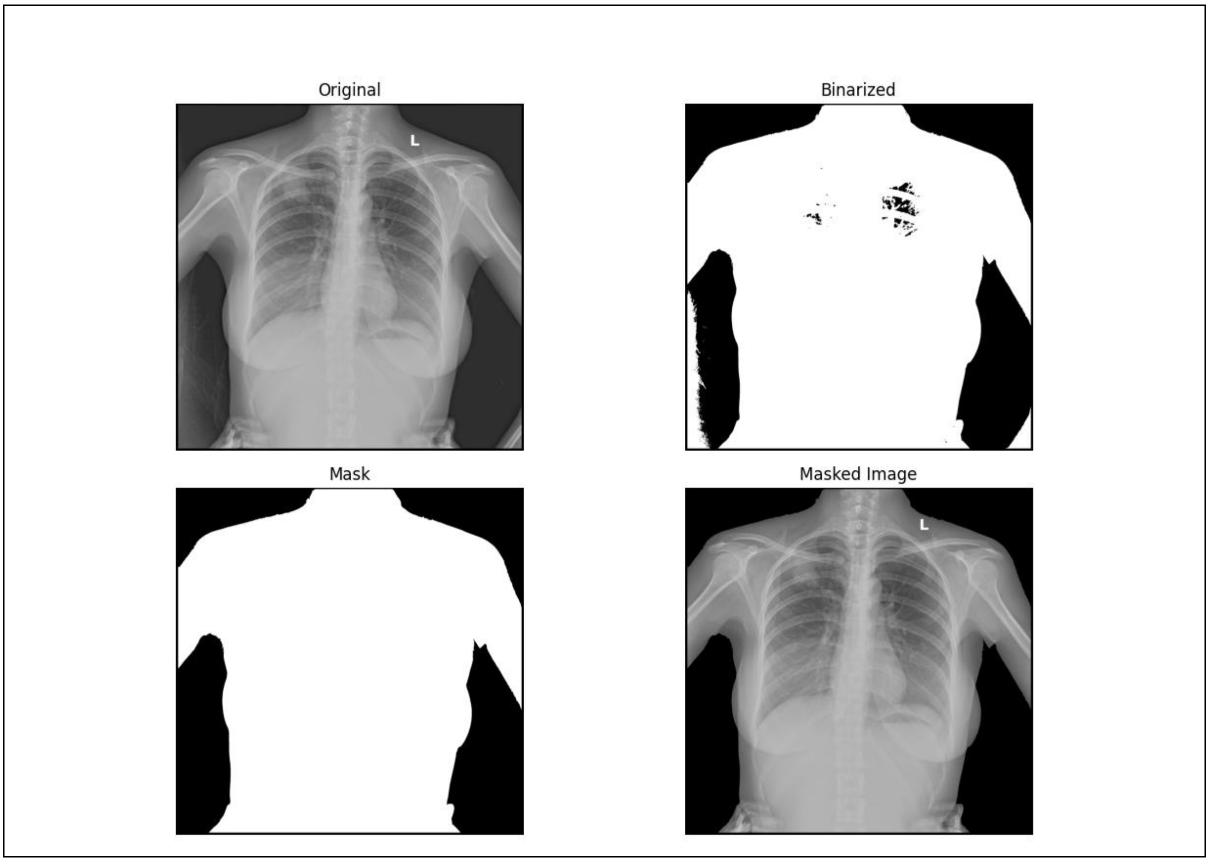
You are expected to submit your work as a ***Python script file on DLE/Moodle*** (no emails accepted) with the following filename after you have been marked in the lab ***(marking in-lab components for these DT labs will be in the last 30mins of any lab session):***

DT-4104-Yourname-lab3.py

TASK-1: Load the DICOM file and apply thresholding to remove background noise *(5 points)*

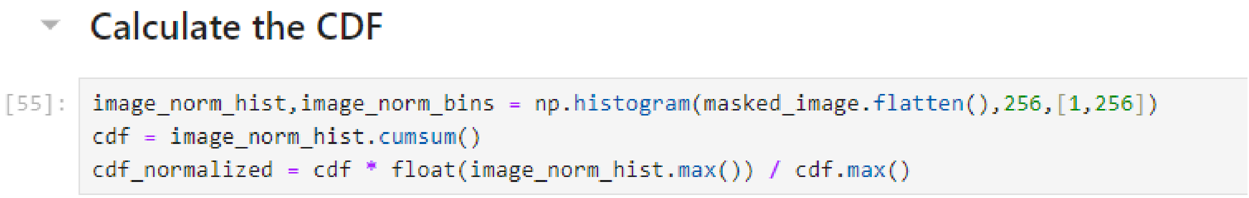
Complete the following tasks:

-1.1: Load a DICOM image from the X-ray folder and print the shape, min pixel value, max pixel value, and the datatype of the image. **(1)**
-1.2: Scale the image to a range of 0-255, datatype to uint8, and display the image. **(1)**
-1.3: Apply thresholding from cv2 on the scaled image by adjusting the clipping limit and removing the background noise. **(2)**
-1.4: Visualize the results (Input Image, Scaled Image, Binarized image, Masked Image) **(1)**

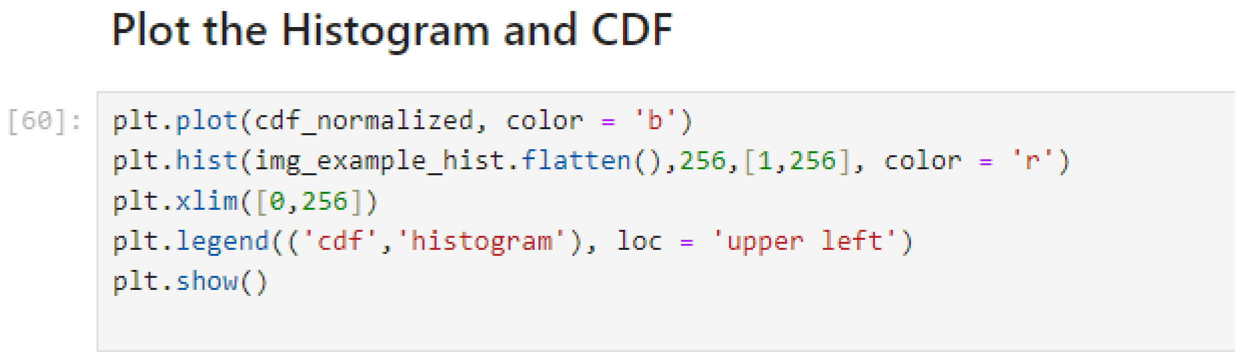


TASK-2: Image enhancement through histogram equalization *(10 points)*

Step-1: Calculate the CDF for the processed image and plot the intensity histogram for the processed image.

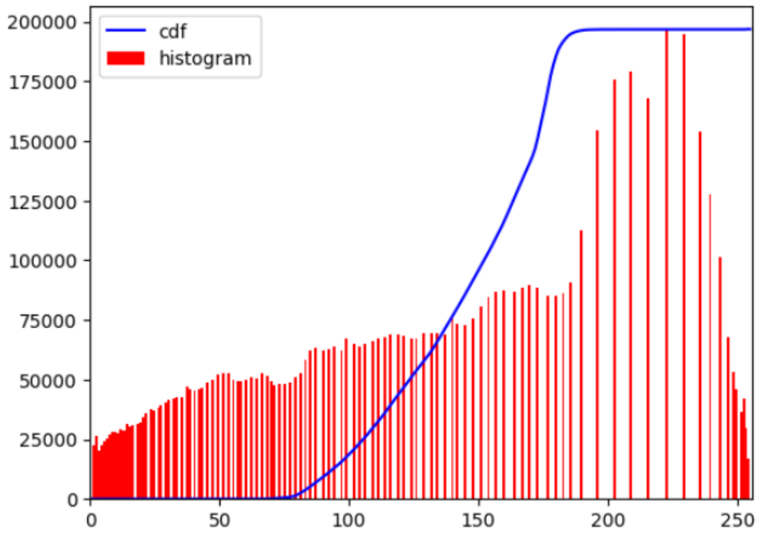

Calculate the CDF

```
[55]: image_norm_hist,image_norm_bins = np.histogram(masked_image.flatten(),256,[1,256])
      cdf = image_norm_hist.cumsum()
      cdf_normalized = cdf * float(image_norm_hist.max()) / cdf.max()
```

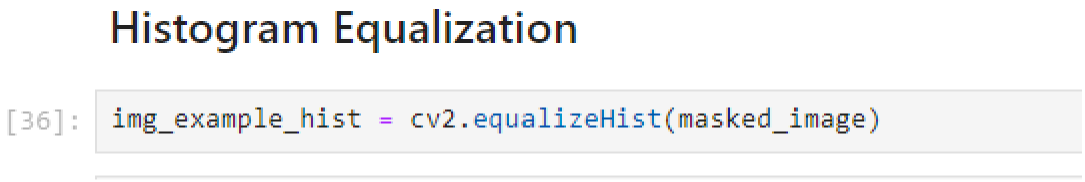


Plot the Histogram and CDF

```
[60]: plt.plot(cdf_normalized, color = 'b')
      plt.hist(img_example_hist.flatten(),256,[1,256], color = 'r')
      plt.xlim([0,256])
      plt.legend(('cdf','histogram'), loc = 'upper left')
      plt.show()
```



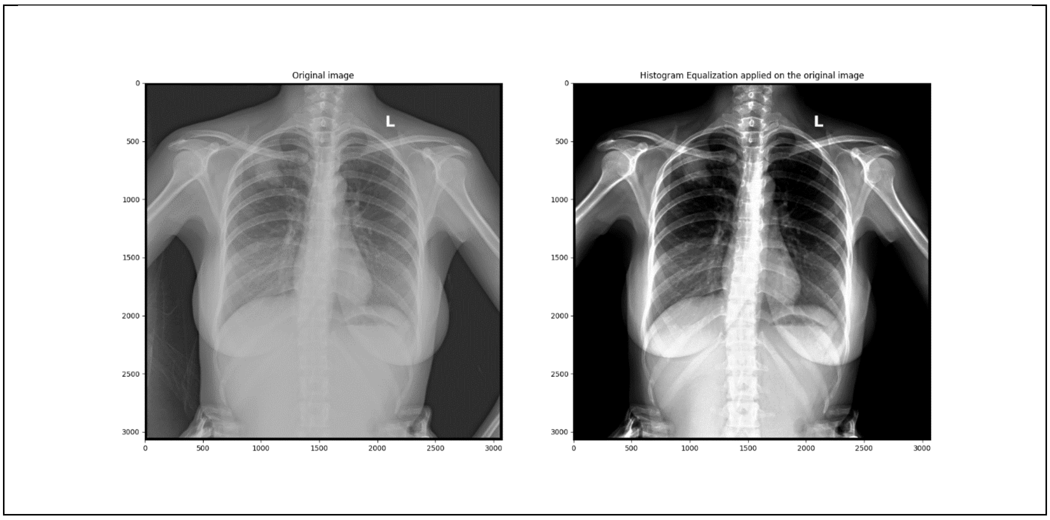
Step-2: Apply Histogram Equalization on the processed image.



Histogram Equalization

```
[36]: img_example_hist = cv2.equalizeHist(masked_image)
```

Step-3: Visualize the original image and the enhanced image.



Step-4: Calculate the CDF for the enhanced image and plot the intensity histogram for the processed image.

Complete the following task:

- **2:** Perform the above exercise from noise removal to image enhancement on the Mammogram image and compare the results/histograms for the original image and the enhanced image (5 pts)

Home Assignment: Try other image enhancement techniques using the open-cv library on Mammogram images and visualize the results with required histograms

(10 pts).

Complete the following tasks:

-1: Load a DICOM image from the Mammogram folder, normalize the image, and remove the background noise using thresholding (2.5 pts)
-2: Apply any image enhancement technique using open-cv library (2.5 pts)
-3: Compare the visualize the results with histogram equalized image (2.5 pts)
-4: Compare the histograms and CDFs and plot them (2.5 pts)
