Assignment 2:
Denoising, Contrast Enhancement, Edge Detection and Hough Transform

ED6001: Medical Image Analysis

Abhinav Azad | ED17B001

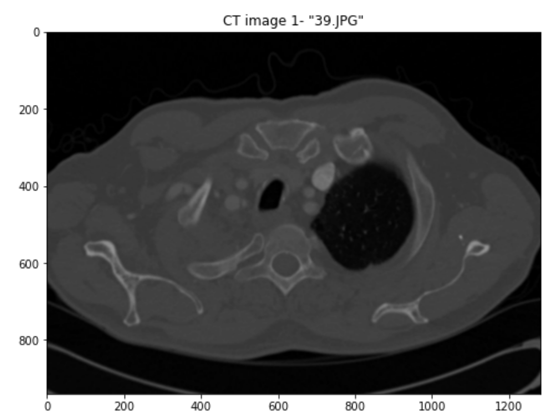
Table of Contents

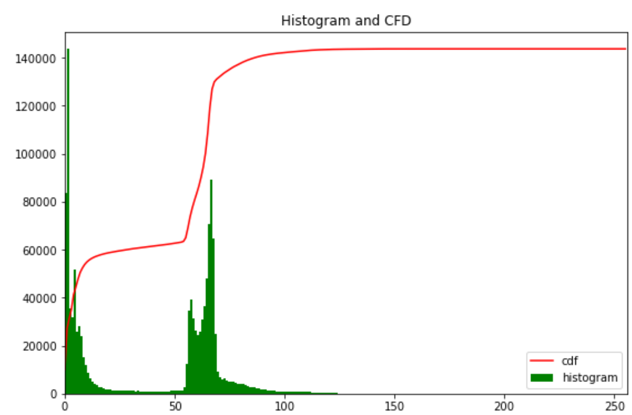
Guidelines

Codes written for this assignment can be found at:
https://github.com/abhiazadz/Edge_detection_Hough_transform.git

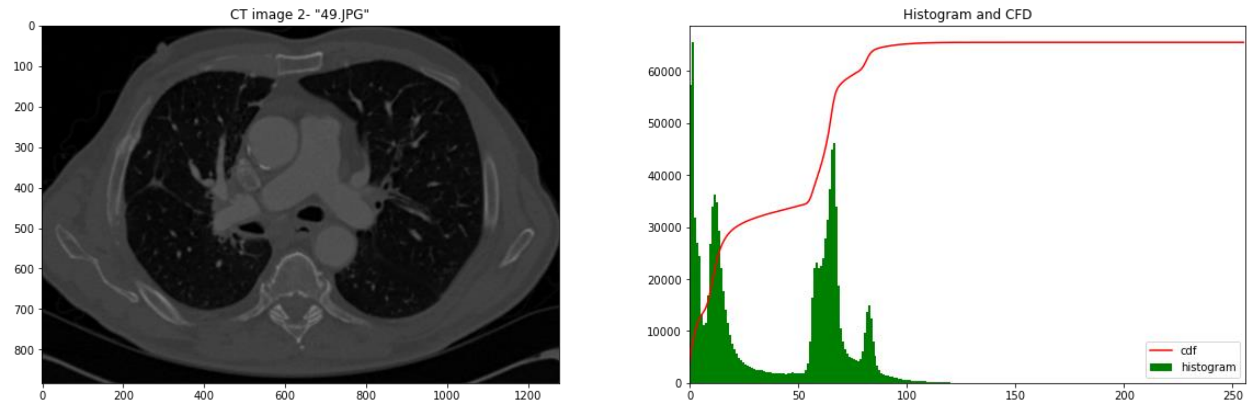
Images provided:

CT image1 – 39.JPG





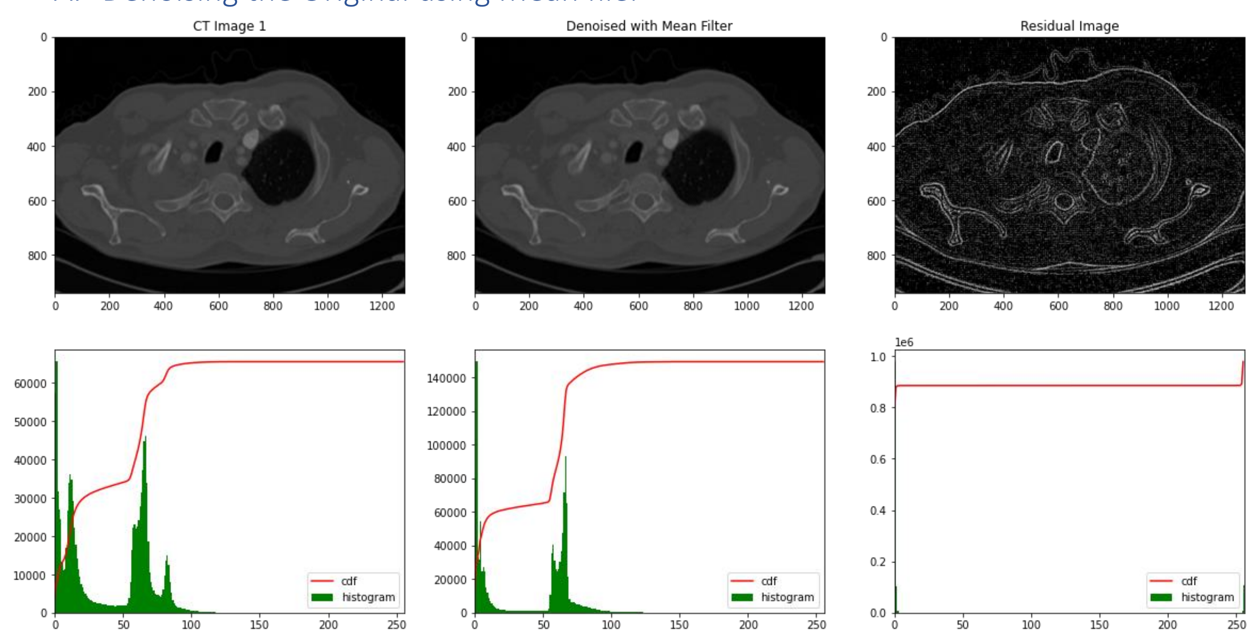
CT image2 – 49.JPG



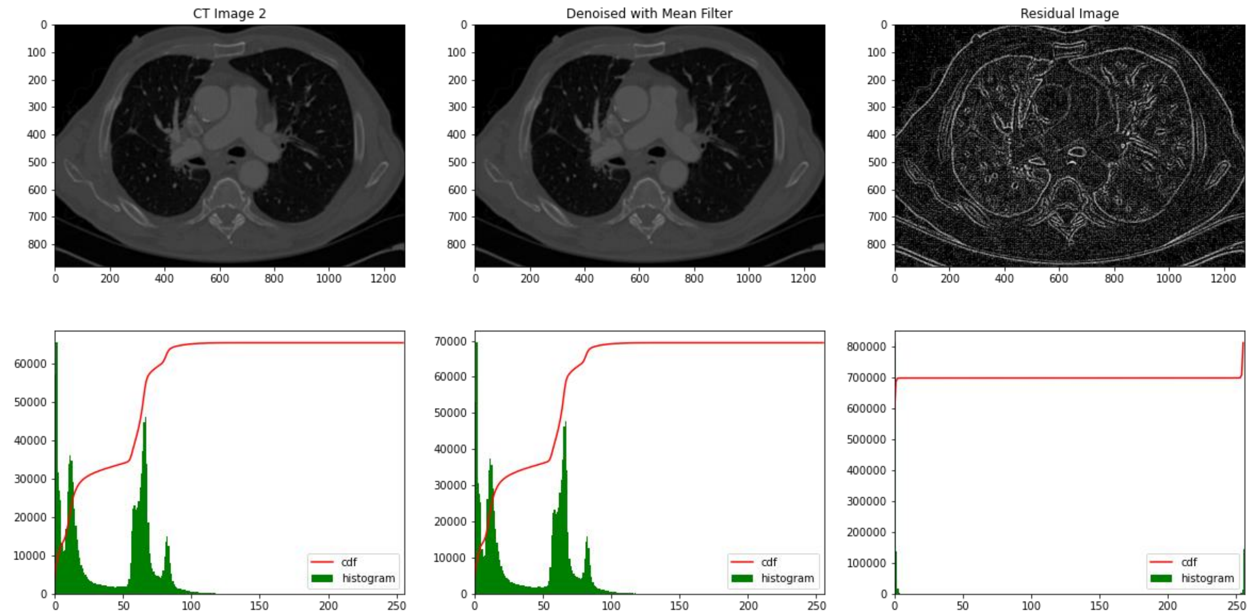
Part 1. Denoising

Denoise the image using appropriate filter.
Summarize your observations as:
a. What is the prominent noise identified in the image or introduced by yourself in the given image?
b. Type of filter used to denoise
c. Plot the residuals and analyze the performance of filter using the metric, PSNR

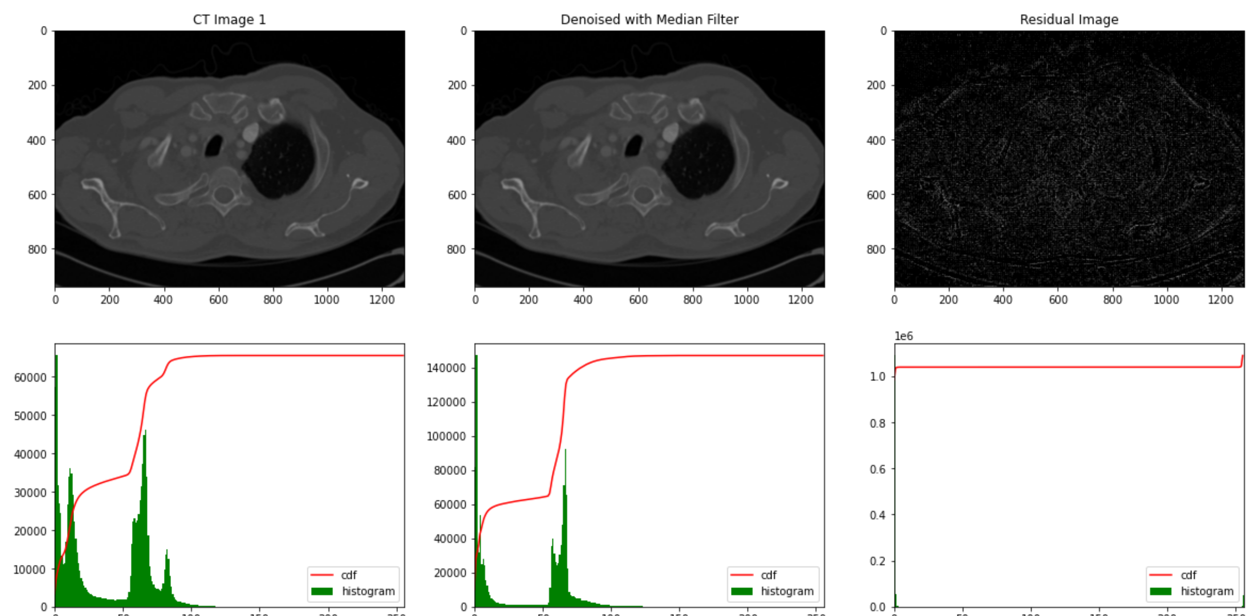
A. Denoising the Original using Mean filer



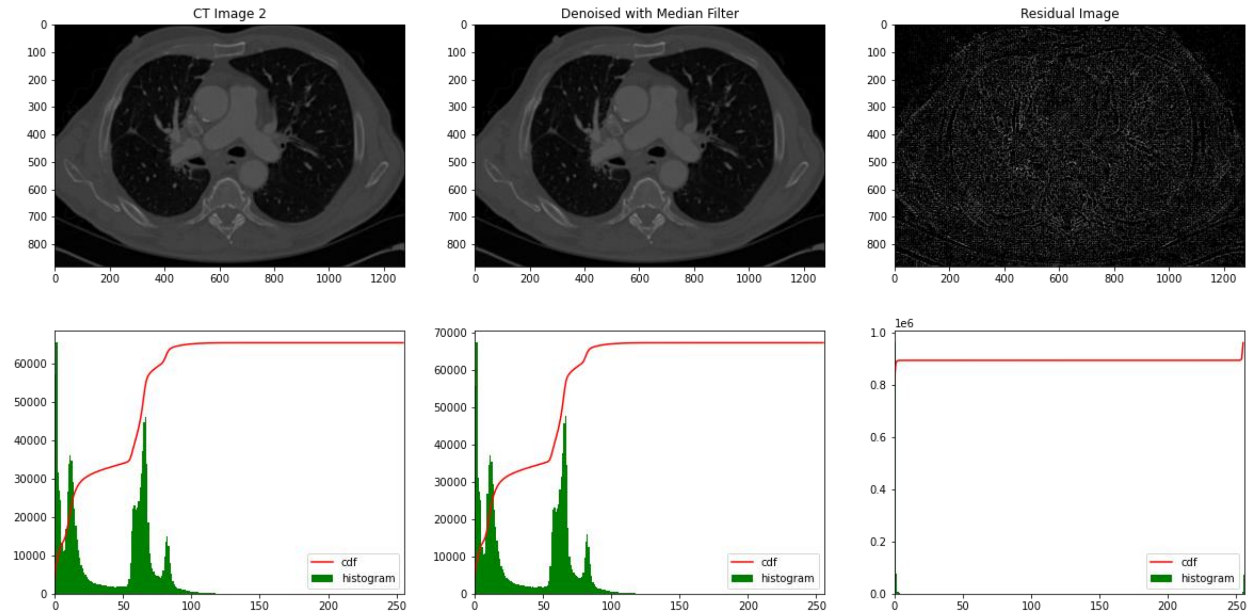
PSNR after Denoising with Mean Filter for CT Image 1: 54.17



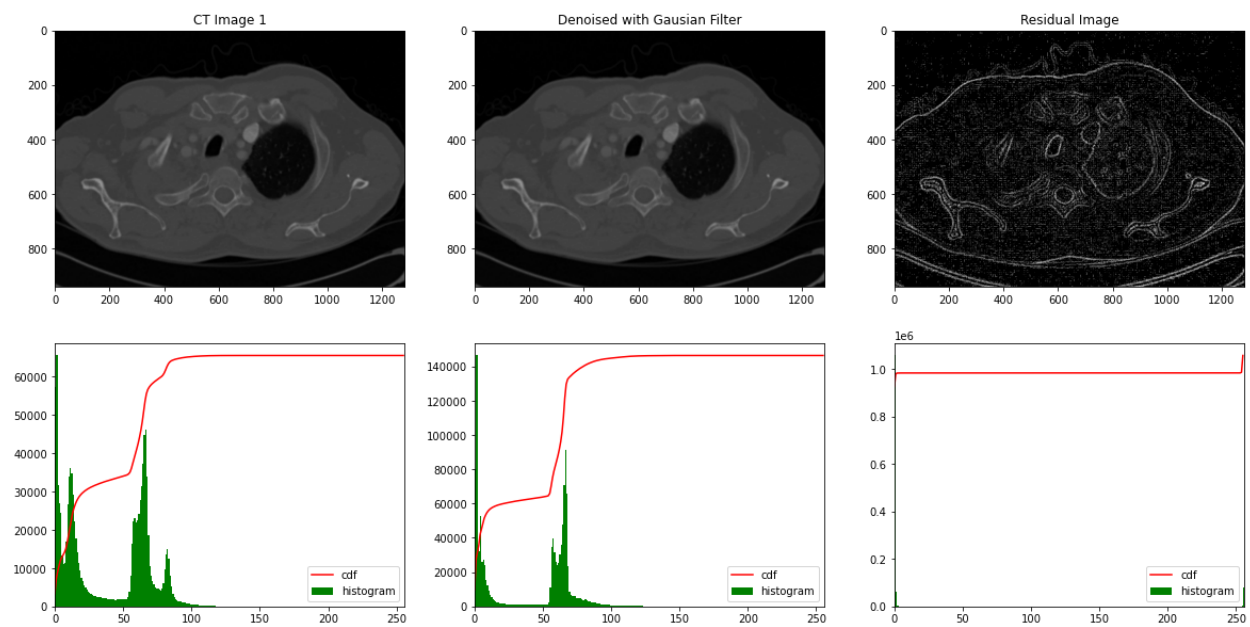
PSNR after Denoising with Mean Filter for CT Image 2: 52.18



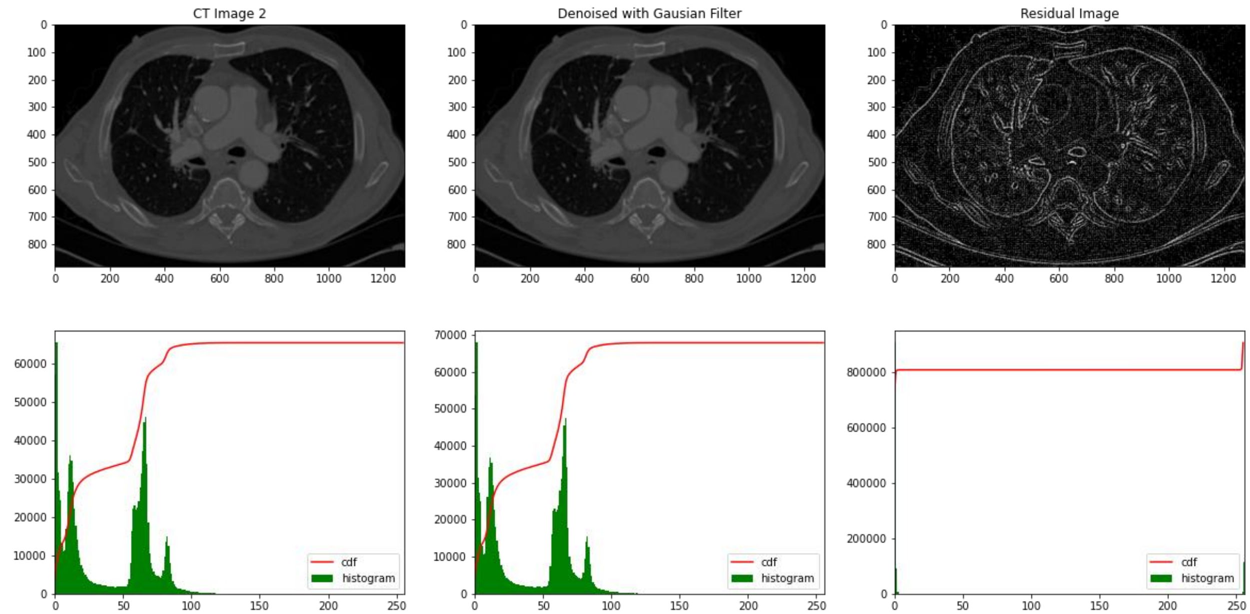
PSNR after Denoising with Median Filter CT Image 1: 57.04



PSNR after Denoising with Median Filter for CT Image 2: 54.78



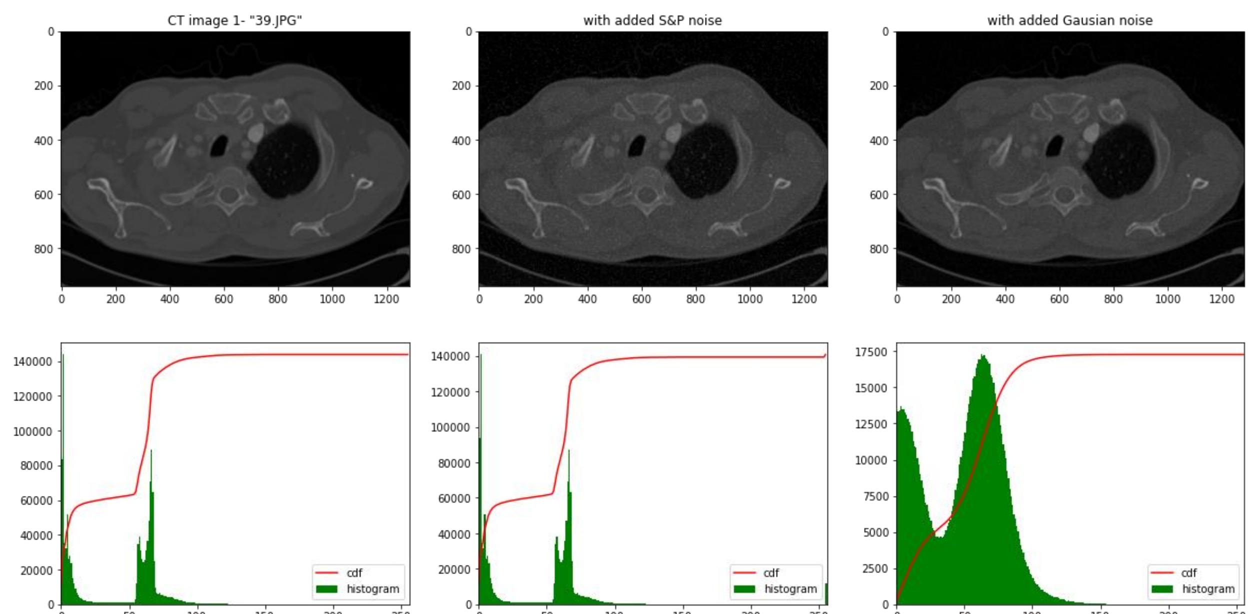
PSNR after Denoising with Gaussian Filter for CT Image 1: 56.52



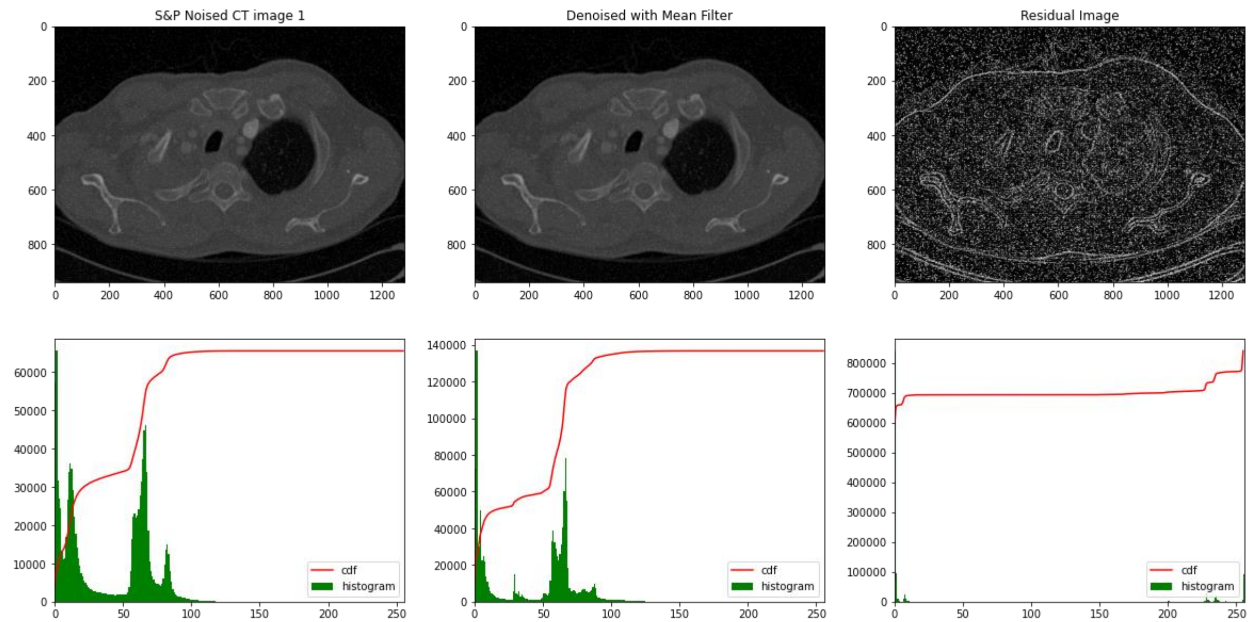
PSNR after Denoising with Gaussian Filter for CT Image 2: 54.38

B. Denoising after Added noise

Adding 2 kinds of noise to CT Image 1

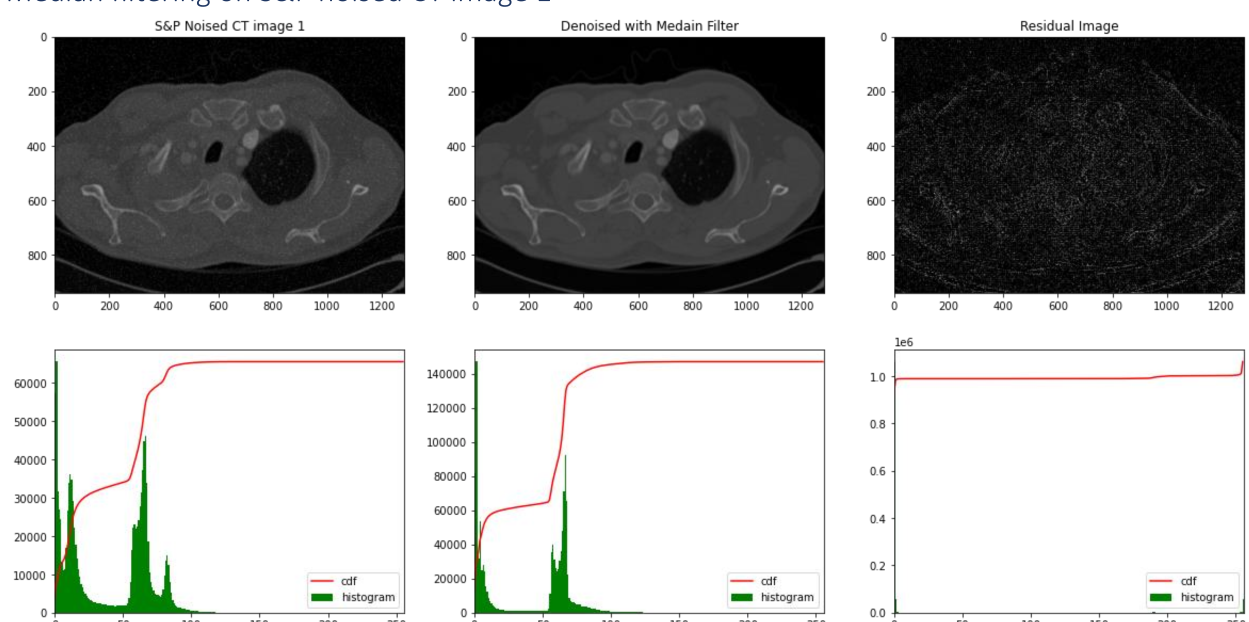


Mean filtering on S&P noised CT image 1



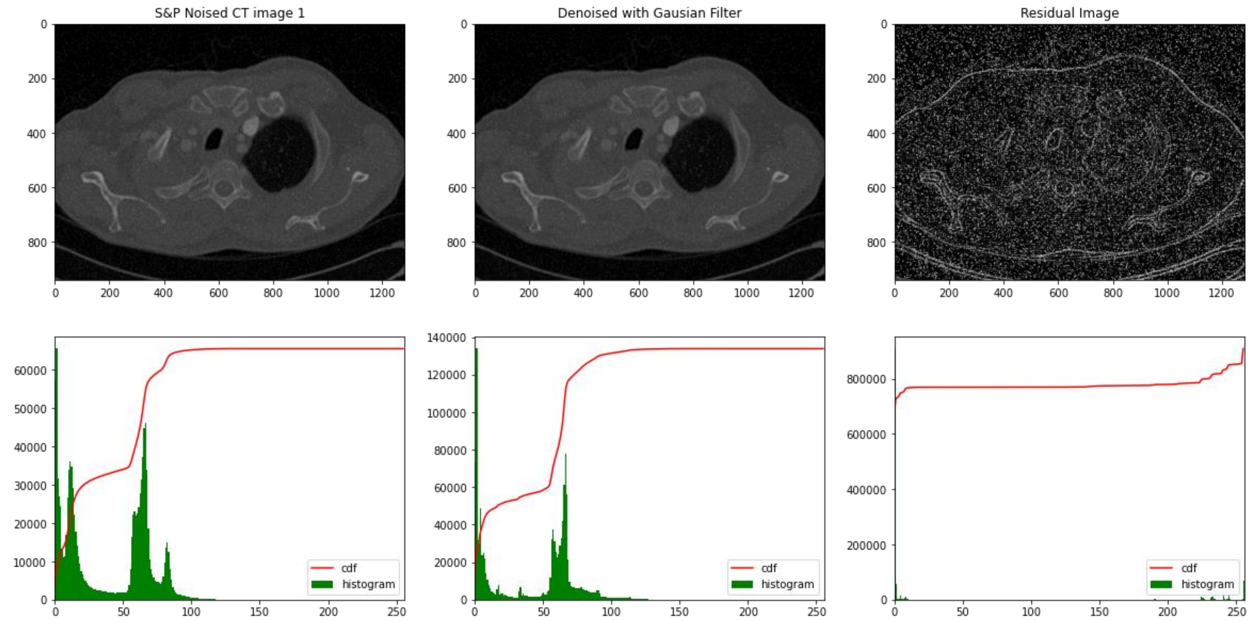
PSNR of S&P Noised CT image 1 w.r.t. to the original image: 18.09
PSNR after Denoising with Mean Filter w.r.t. to S&P Noised CT image 1: 18.67
PSNR of S&P Noised CT image 1 Denoised with Mean filter w.r.t. to original image: 27.15

Median filtering on S&P noised CT image 1



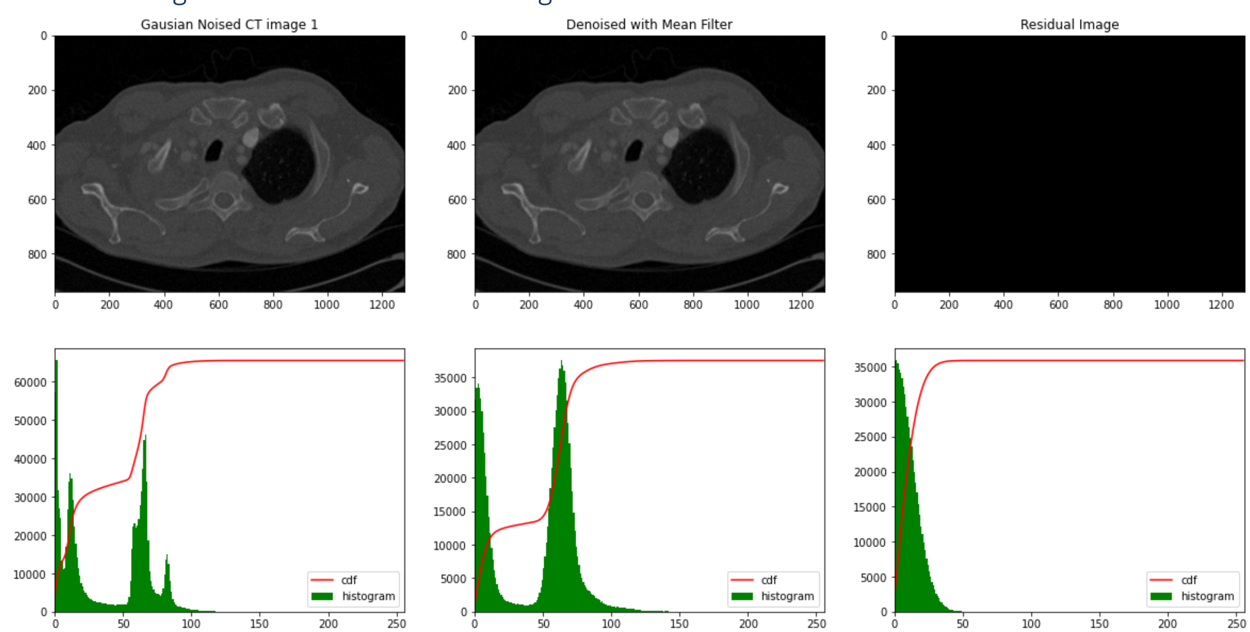
PSNR of S&P Noised CT image 1 w.r.t. to the original image: 18.09
PSNR after Denoising with Median Filter w.r.t. to S&P Noised CT image 1: 18.1
PSNR of S&P Noised CT image 1 Denoised with Median filter w.r.t. to original image: 55.0

Gaussian filtering on S&P noised CT image 1



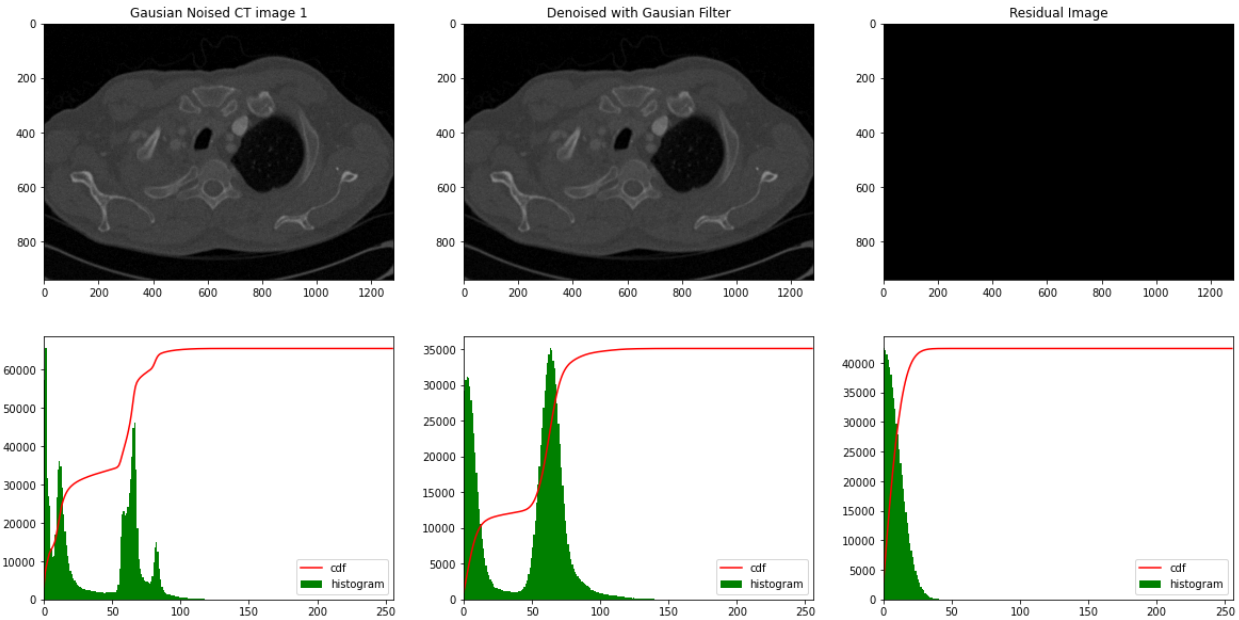
PSNR of S&P Noised CT image 1 w.r.t. to the original image: 18.09
PSNR after Denoising with Gaussian Filter w.r.t. to S&P Noised CT image 1: 20.09
PSNR of S&P Noised CT image 1 Denoised with Gaussian filter w.r.t. to original image: 26.23

Mean filtering on Gaussian noised CT image 1



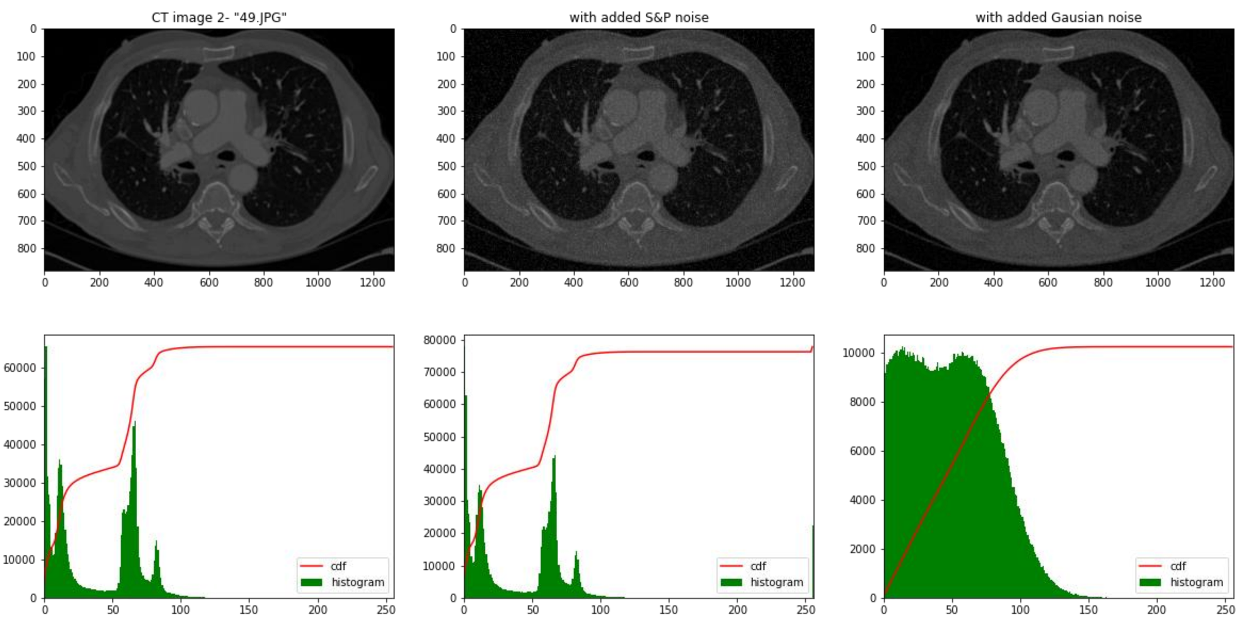
PSNR of Gaussian Noised CT image 1 w.r.t. to the original image: 25.11
PSNR after Denoising with Mean Filter w.r.t. to Gaussian Noised CT image 1: 25.62
PSNR of Gaussian Noised CT image 1 Denoised with Mean filter w.r.t. to original image: 34.6

Gaussian filtering on Gaussian noised CT image 1

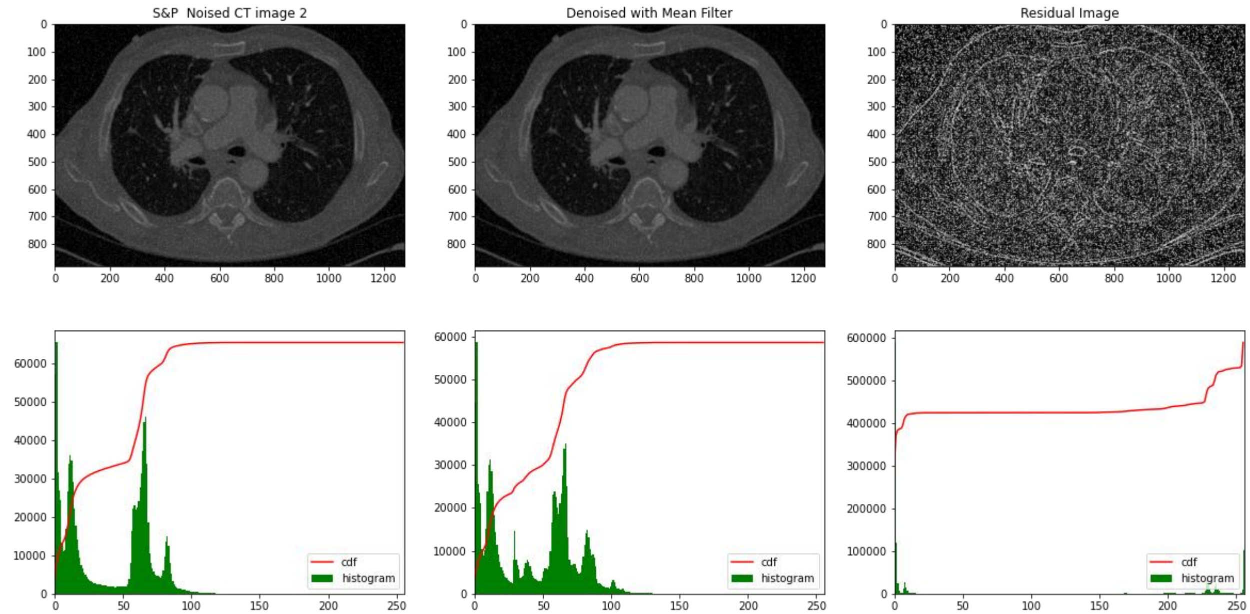


PSNR of Gaussian Noised CT image 1 w.r.t. to the original image: 25.11
PSNR after Denoising with Gaussian Filter w.r.t. to Gaussian Noised CT image 1: 27.04
PSNR of Gaussian Noised CT image 1 Denoised with Gaussian filter w.r.t. to original image: 33.61
05

Adding 2 kinds of noise to CT Image 2

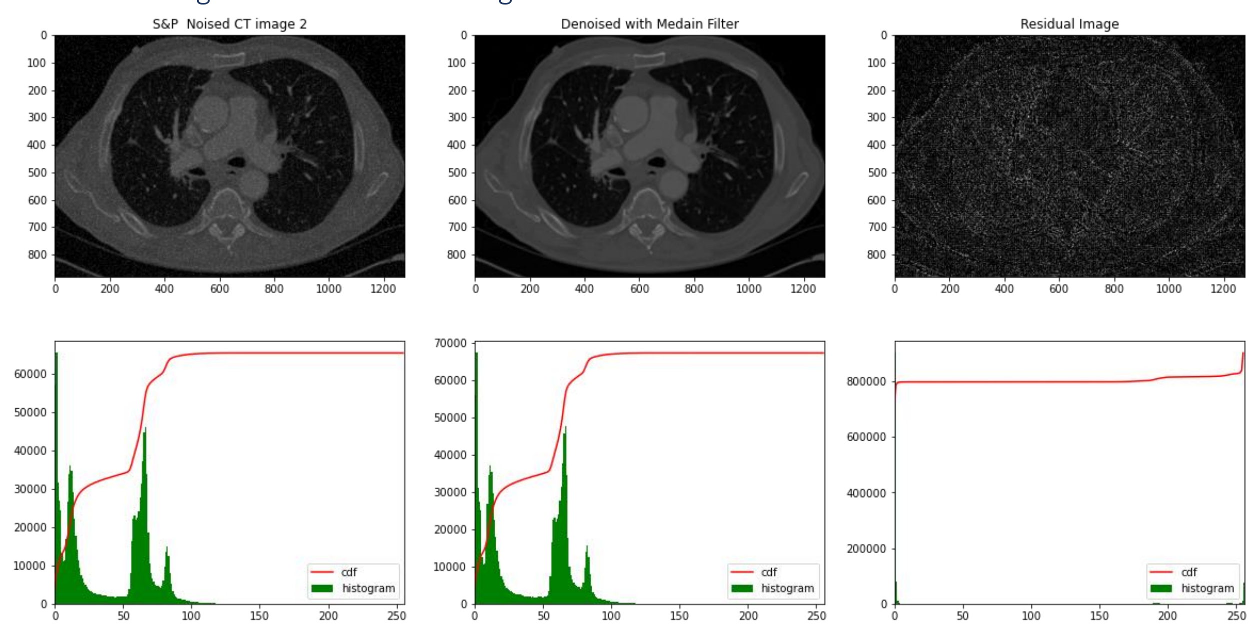


Mean filtering on S&P noised CT image 2



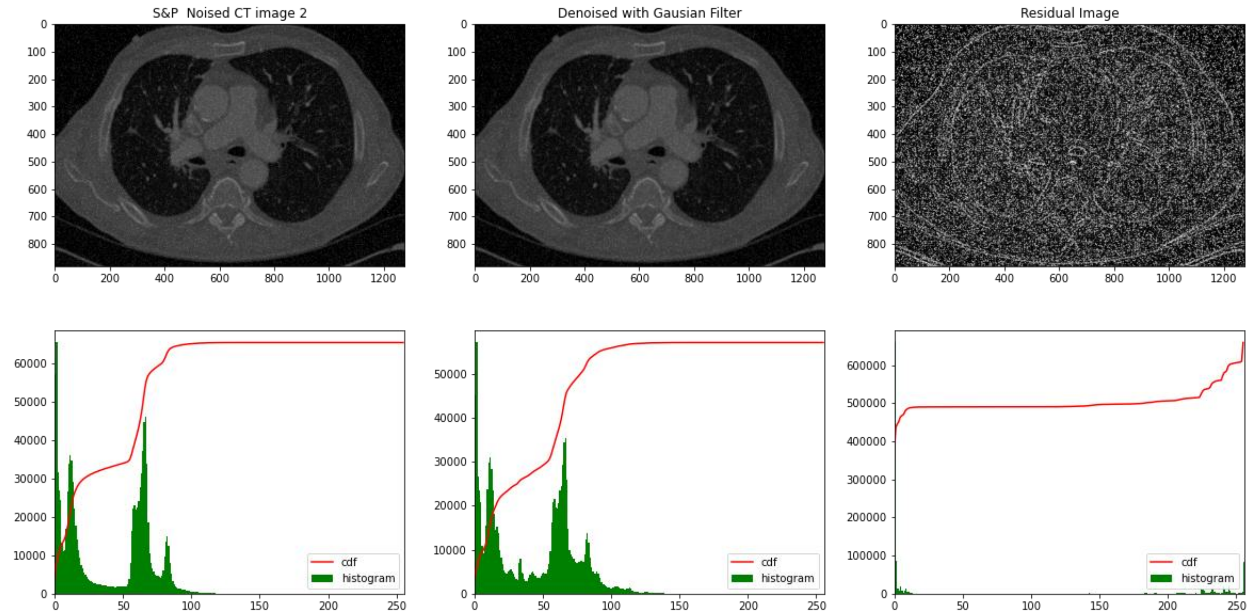
PSNR of S&P Noised CT image 2 w.r.t to the original image: 18.11
PSNR after Denoising with Mean Filter w.r.t to S&P Noised CT image 2: 18.68
PSNR of S&P Noised CT image 2 Denoised with Mean filter w.r.t to original image: 27.14

Medain filtering on S&P noised CT image 2



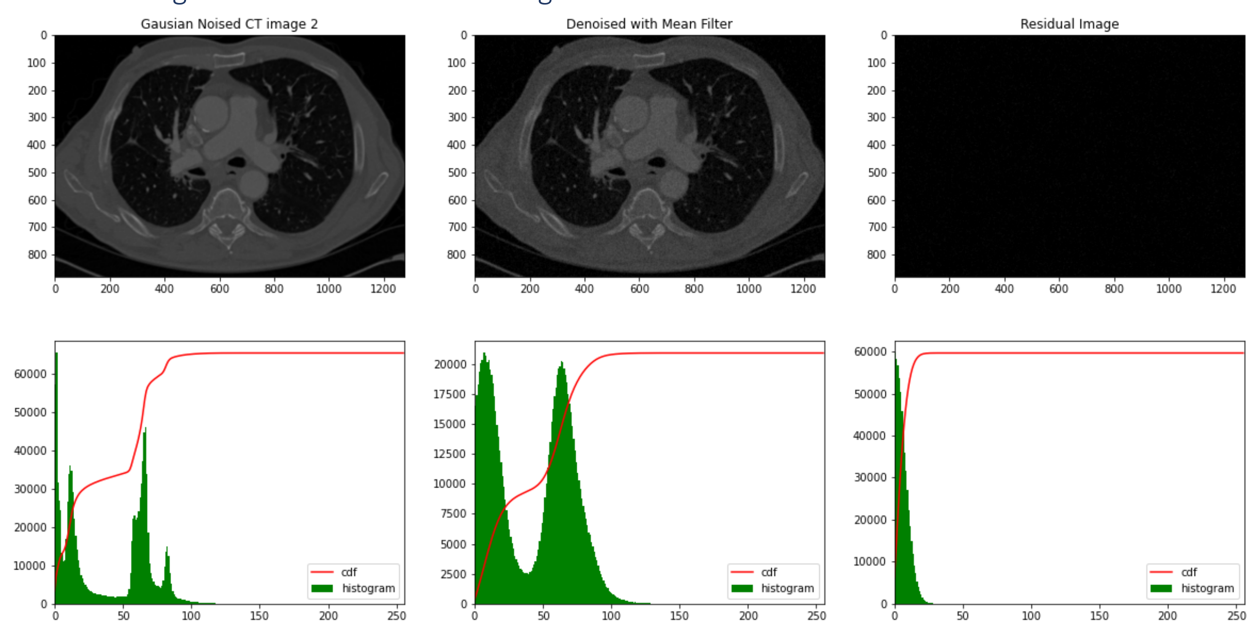
PSNR of S&P Noised CT image 2 w.r.t to the original image: 18.11
PSNR after Denoising with Median Filter w.r.t to S&P Noised CT image 2: 18.12
PSNR of S&P Noised CT image 2 Denoised with Median filter w.r.t to original image: 52.41

Gaussian filtering on S&P noised CT image 2



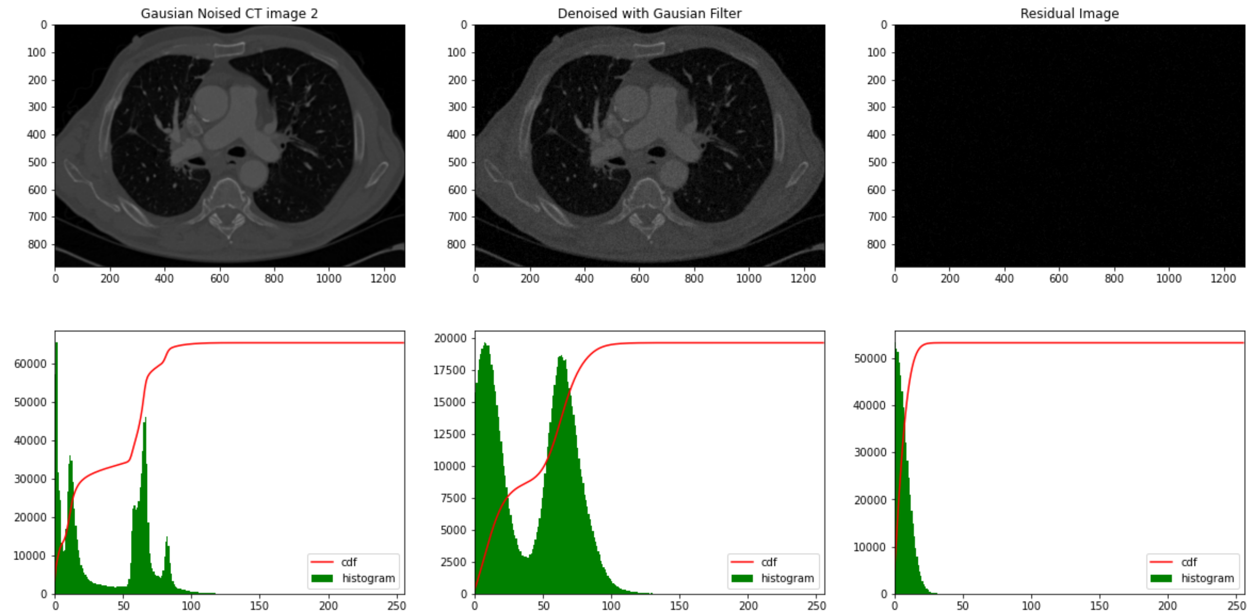
PSNR of S&P Noised CT image 2 w.r.t to the original image: 18.11
PSNR after Denoising with Gaussian Filter w.r.t to S&P Noised CT image 2: 20.1
PSNR of S&P Noised CT image 2 Denoised with Gaussian filter w.r.t to original image: 26.23

Mean filtering on **Gaussian** noised CT image 2



PSNR of Gaussian Noised CT image 2 w.r.t to the original image: 21.15
PSNR after Denoising with Mean Filter w.r.t to Gaussian Noised CT image 2: 21.66
PSNR of Gaussian Noised CT image 2 Denoised with Mean filter w.r.t to original image: 30.63

Gaussian filtering on **Gaussian** noised CT image 2



PSNR of Gaussian Noised CT image 2 w.r.t to the original image: 21.15
PSNR after Denoising with Gaussian Filter w.r.t to Gaussian Noised CT image 2: 23.08
PSNR of Gaussian Noised CT image 2 Denoised with Gaussian filter w.r.t to original image: 29.63

Part 1 – results:

PSNR of Denoised Original CT images with various filters

PSNRs after Filtering with:	CT Image 1	CT image 2
Mean filer	54.17	52.18
Gaussian Filter	56.52	54.38
Median Filter	57.04	**54.78**

PSNR after denoising of Artificially added noise

Noise type: S&P – CT image 1

Filter used to denoise S&P	Noised to Original	Noised to denoised	Denoised to Original
Mean filer	18.09	18.67	27.15
Median Filter	18.09	18.1	**55.0**
Gaussian Filter	18.09	20.09	26.23

Noise type: Gaussian distribution noise -CT image 1

Filter used to denoise Gaussian noise	Noised to Original	Noised to denoised	Denoised to Original
Mean filer	25.11	25.62	**34.6**
Gaussian Filter	25.11	27.04	33.61

Noise type: S&P – CT image 2

Filter used to denoise S&P	Noised to Original	Noised to denoised	Denoised to Original
Mean filer	18.11	18.68	27.14
Median Filter	18.11	18.12	**52.41**
Gaussian Filter	18.11	20.1	26.23

Noise type: Gaussian distribution noise -CT image 2

Filter used to denoise Gaussian noise	Noised to Original	Noised to denoised	Denoised to Original
Mean filer	21.15	21.66	**30.63**
Gaussian Filter	25.15	23.08	29.63

Part 1 - Observations

Denoising Originals:

1. Median filter has the highest PSNR for both the images, but Gaussian and Mean filters are also nearly effective.
2. The edges in the residual image is prominent for Gaussian and mean filters blurred images unlike the case of Median filtering.
3. As compared to Mean and Gaussian filters, Median filter preserves more edges in the images along with filtering the noise out.

Denoising of Artificially noised images:

4. Median filter works the best for S&P noise.
5. Mean filter works the best for Gaussian distribution noise.
6. Histograms of the original and filtered image do not vary much, the amount of noise is very less, and nothing can be surely said about the type of noise. However, since median filter works best, the noise may be Poisson distributed or salt and pepper.

Part 1 – conclusions

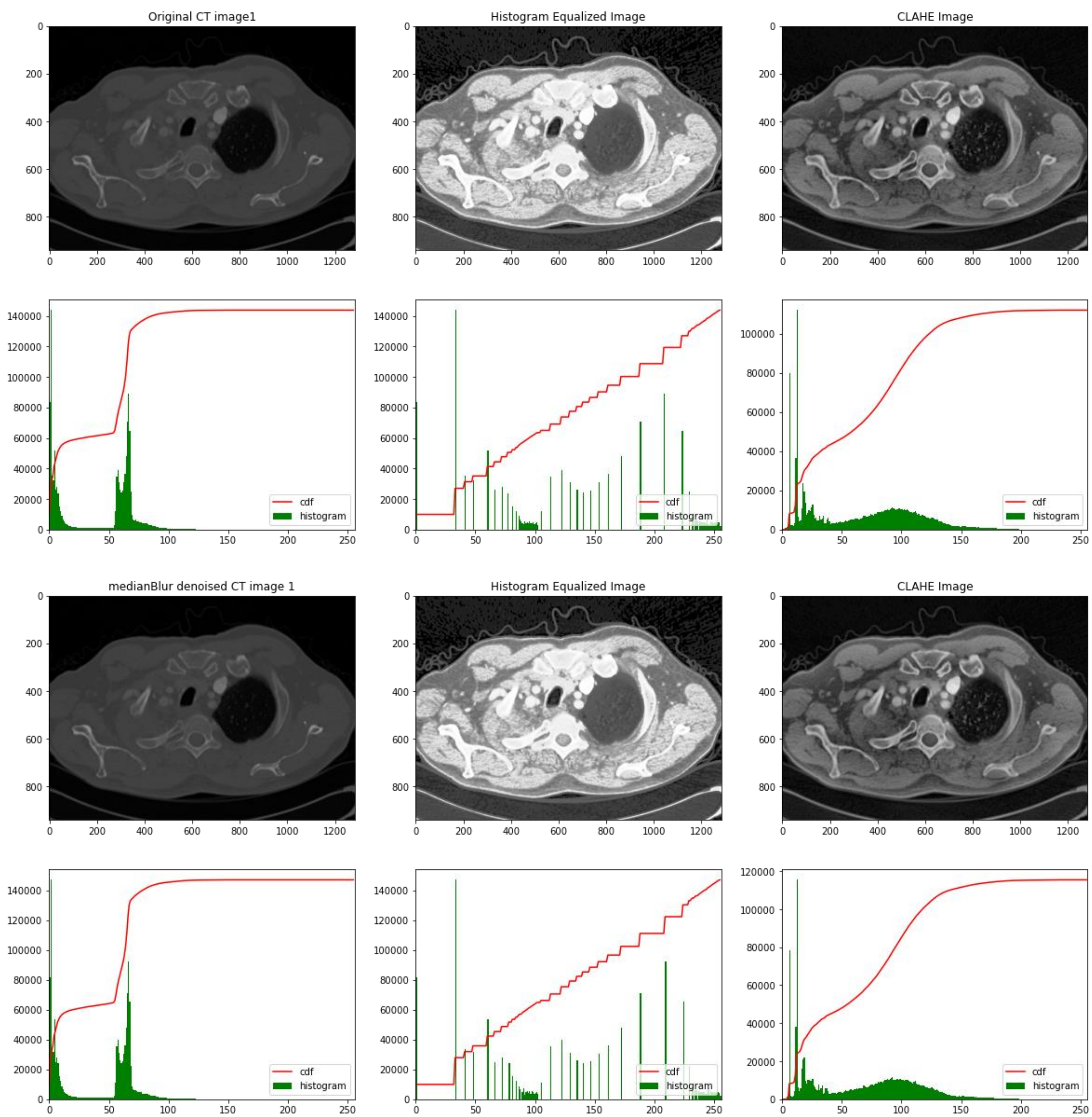
1. Noise in CT images is due to quantum noise inherent in photon detection and electronic noise. The number of photons reaching individual pixels on the detector follows a Poisson distribution.
2. The noise contains a broad range of frequency components. Linear filters like mean and Gaussian are effective because the high-frequency components can be reduced by the filter.
3. The smoothing effects of the blurring filters not only on noise elements but also on the edges of anatomical structures that consist of high-frequency signals. As a result, blurring is caused by unclear edges.
4. Since Median filter is a non-linear filter, it operates such that each target pixel is replaced by the median of pixels in the analysis window. The median filter can preserve edges, while still effectively reducing noise.
5. This explains why the median filter may show best results for the CT images.

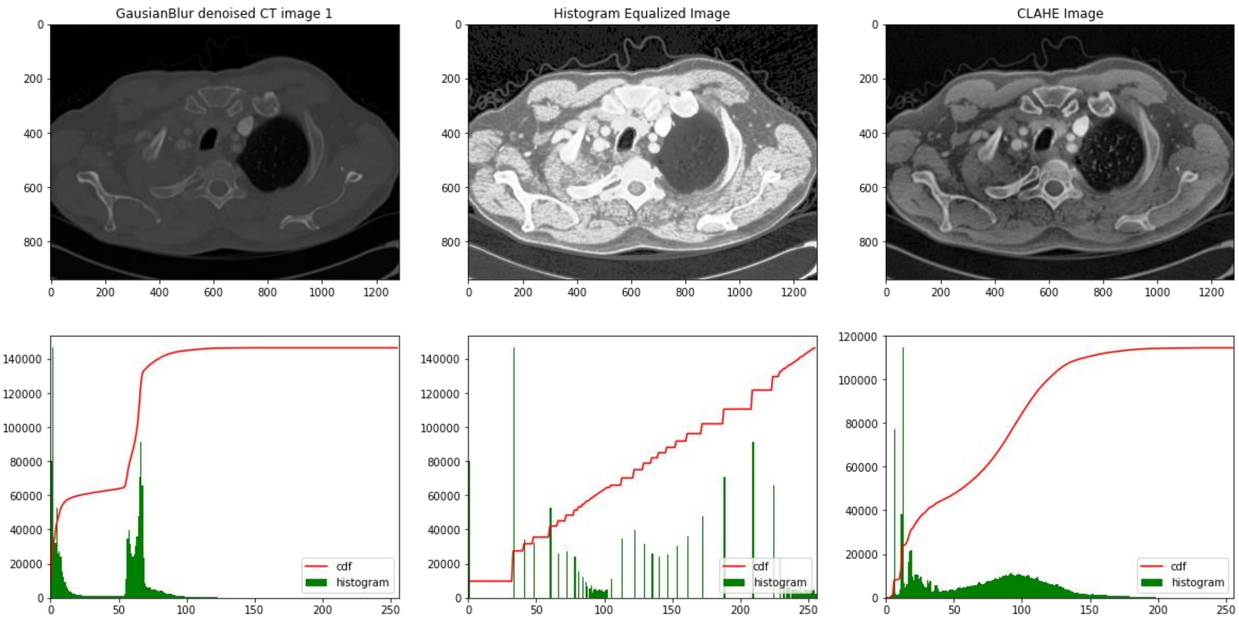
Part 2: Contrast enhancement and edge detection

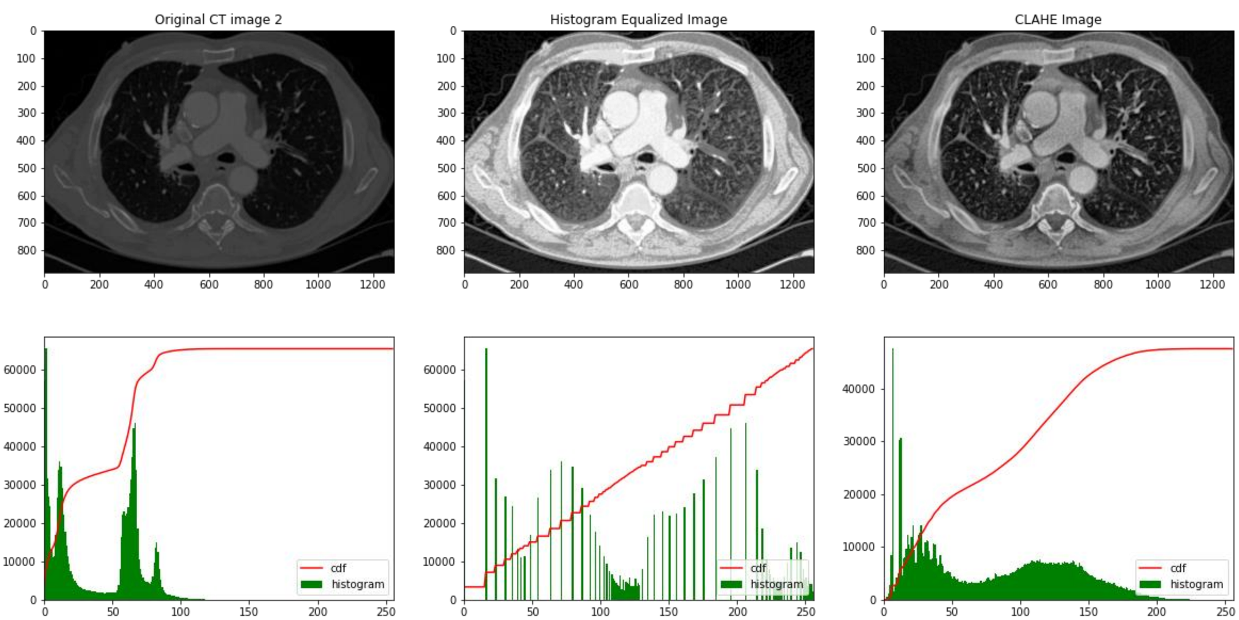
i. Enhance the contrast of the images. Visualize the input and the contrast enhanced image using histograms.

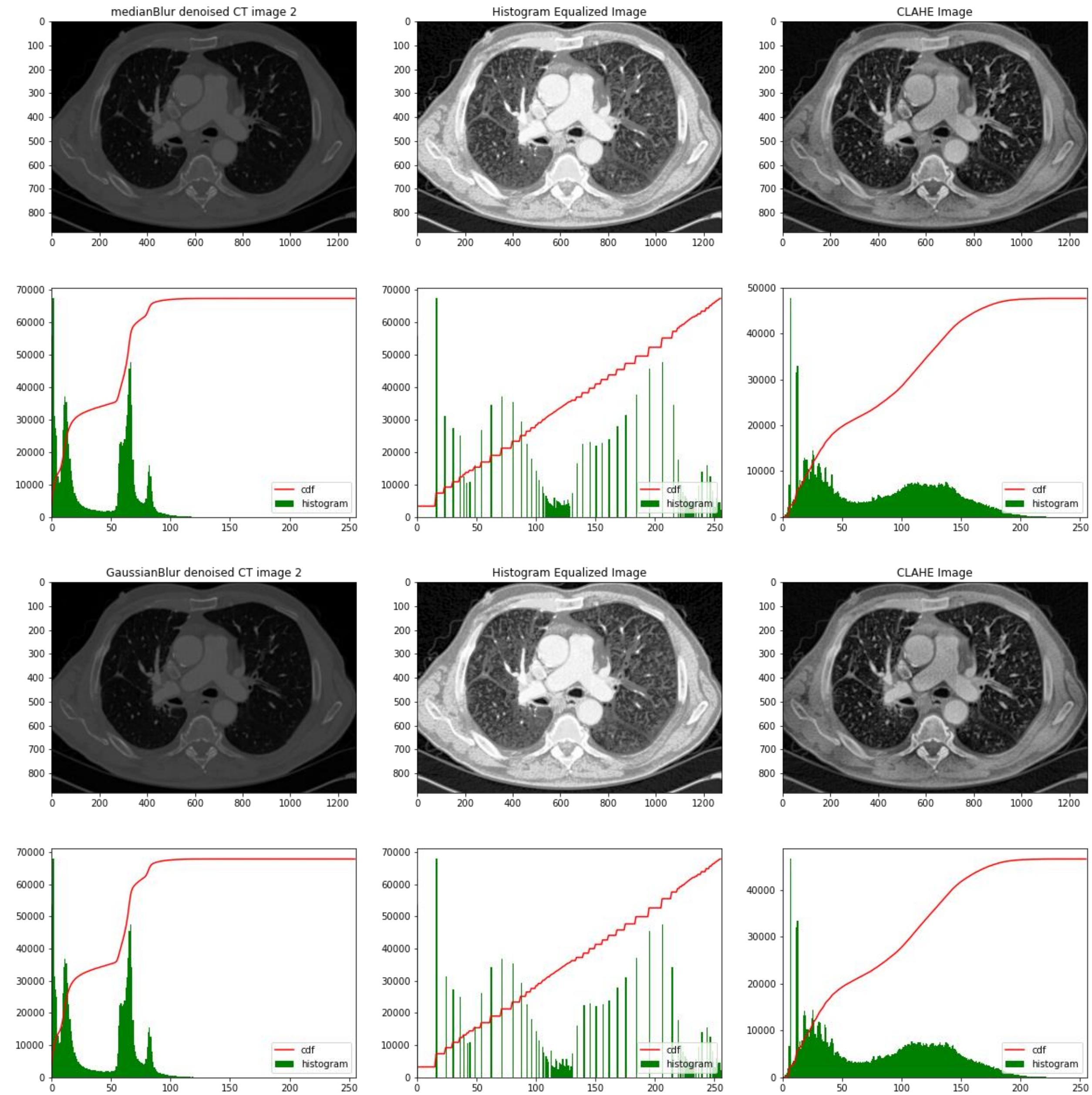
ii. After contrast enhancement, try to detect the edges of the sternum (crop the image such that it encompasses at least twice the size of the sternum) using various edge detectors and compare the performance of the edge detectors.

The given two images have very poor contrast, with thin peaks in the histograms at certain values. Histogram Equalization and Contrast Limit Adaptive Histogram Equalization (CLAHE) are both applied on the images after de-noising. Clip Limit = 3 and window size = 8 are used.




GausianBlur denoised CT image 1
Histogram Equalized Image
CLAHE Image
cdf
histogram
cdf
histogram
cdf
histogram


Original CT image 2
Histogram Equalized Image
CLAHE Image
cdf
histogram
cdf
histogram
cdf
histogram

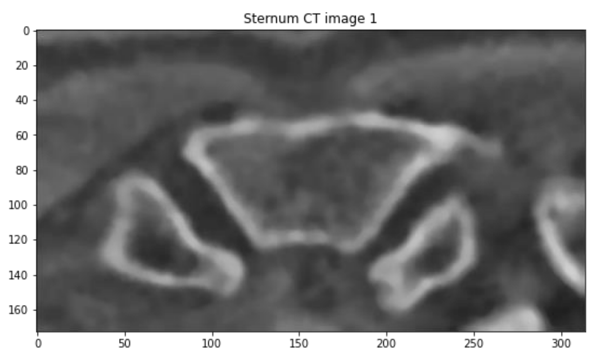

medianBlur denoised CT image 2
Histogram Equalized Image
CLAHE Image
cdf
histogram
cdf
histogram
cdf
histogram
GaussianBlur denoised CT image 2
Histogram Equalized Image
CLAHE Image
cdf
histogram
cdf
histogram
cdf
histogram

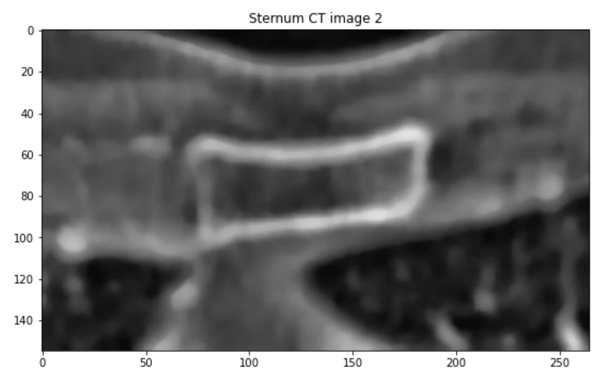
(i) Part 2-(i) – Observations with Contrast enhancement:

•For both the images, histogram equalization tries to achieve contrast enhancement globally by flattening the histogram. This leads to over-amplification of contrast and enhancement in contrast of noise signals as well.
•Histogram equalization produces undesirable effects like visible image gradient when applied to 8-bit images displayed with 8-bit gray-scale palette: it further reduces color depth (number of unique shades of gray) of the image.
•Histogram equalization works best when the background and foreground are both light or both dark, which is not the case with the given images which have two separate peaks in their histograms.
•CLAHE computes several histograms locally, enhancing the local contrast of regions having low contrast. It
avoids the over-amplification of contrast by clipping the histogram at a predefined value before computing
the CDF.

(ii) Part 2-(ii) Cropped images of the Sternum ROI and EDGE detection

The cropped images of Sternum from the two CT images are shown here.

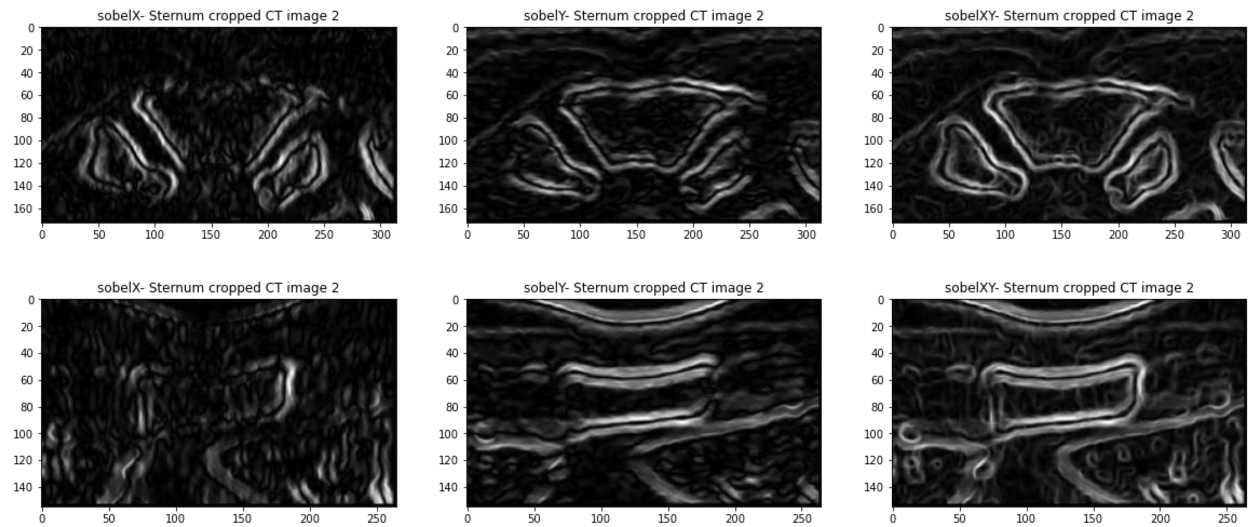




The CLAHE enhanced images are now used for edge detection of the sternum. The images are cropped out to get the sternum.

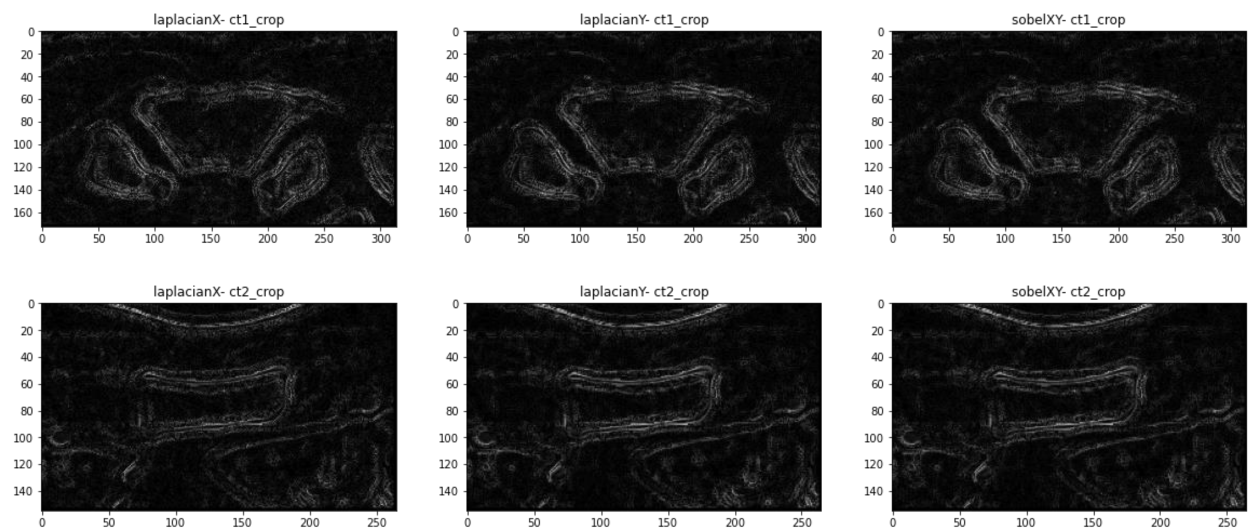
SOBEL EDGE DETECTOR

Sobel edge detector with dx = 2, dy = 1 and kernel size = 3 was applied on both the images.



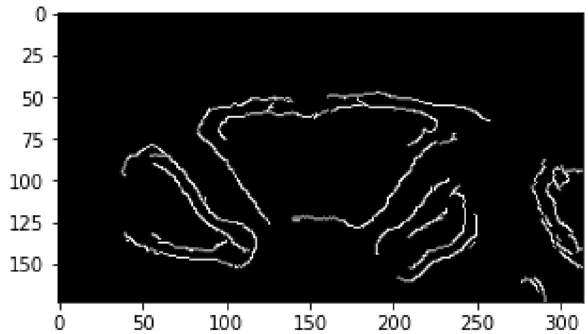
LAPLACIAN EDGE DETECTOR

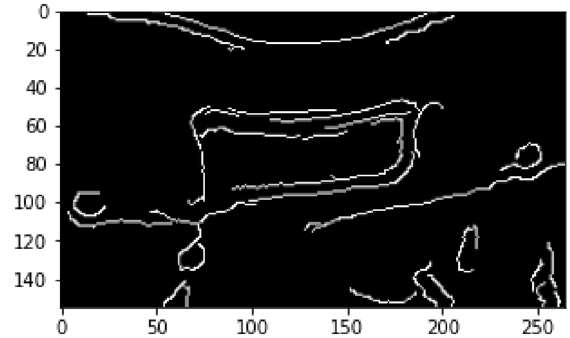
Laplacian edge detector with varying kernel sizes was applied on both images.



CANNY EDGE DETECTOR

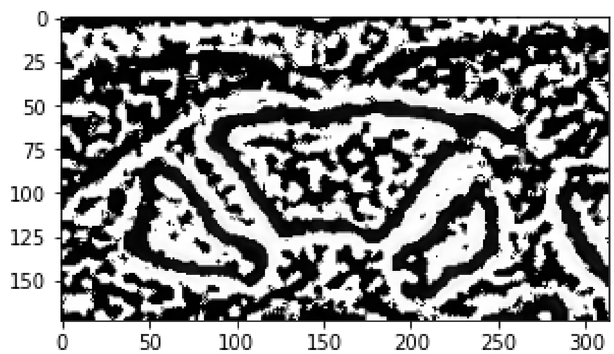
Canny edge detector was applied to image 1 with thresholds (50, 200) on image 1 and (130, 255) on image

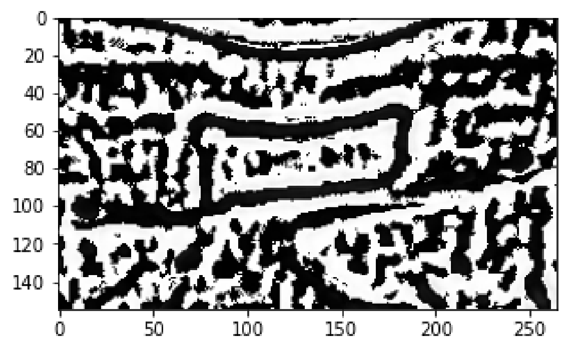




GAUSSIAN DIFFERENCE

In this, Gaussian blurring is applied to the same image with two different kernel sizes. This blurs the edge regions. The more blurred image is subtracted from the less blurred image to get the edges. For both the images, kernel size = 1 is used for less blurring.





Part 3: Application of Hough Transform

Use the best contrast enhanced edge detected image from part 2 and apply Hough transform to locate the sternum.

NOTE: Choose an appropriate value of 'm' (the blend parameter) of Lamé curve and thus the parameters space reduces to 2.

Hough Transform is a way to fit a parametric curve to an edge image. We first assume a certain shape and write its equation. We initiate an accumulator matrix representing the parameter space, with each cell initialized to zero. Then, the co-ordinates of each point on the edge in image space is substituted in the curve equation. The value at the parameter cell for which the equation is satisfied is incremented by 1. In the end, the cell(s) having the maximum number of votes correspond to the parameters which define the equation of the curve which best fits the edge.

In case of an edge having unclear boundaries in the edge image, as in Sobel/Laplacian Edge Detectors, there 10 may be multiple curves with the maximum vote. The edges determined by Canny Edge Detector have the most clear and sharp boundaries. Hence, it will ensure that a smaller number of unique curves are found out.

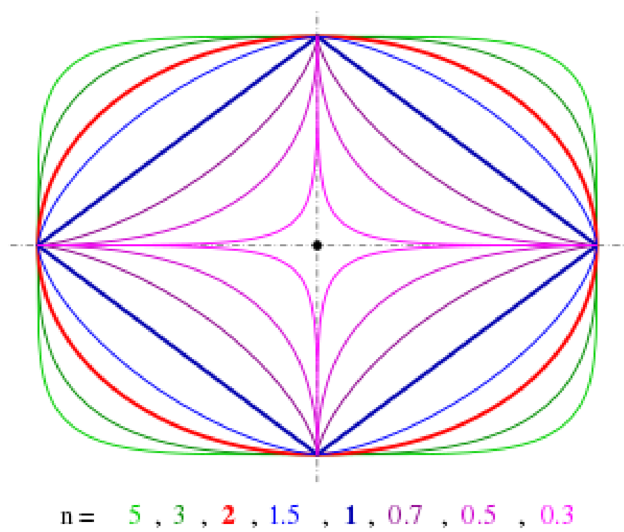
It is important to reduce the number of parameters as much as possible so as to reduce the dimensionality of parameter space and hence the computation cost.

Lamé curve

The Lamé curve after Gabriel Lamé also known as a superellipse, is a closed curve resembling the ellipse, retaining the geometric features of semi-major axis and semi-minor axis, and symmetry about them, but a different overall shape.

The equation for the curve is $|x/a|^m+|y/b|^m=1$

(m=n in the figure)



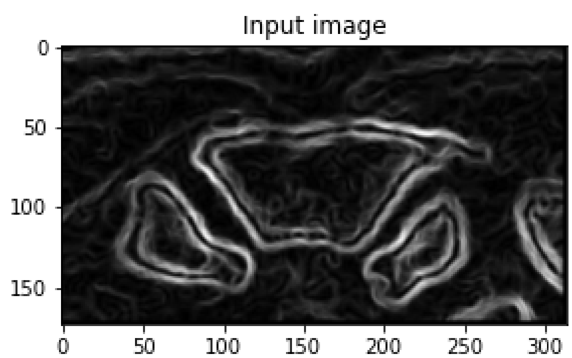
Python implementation results

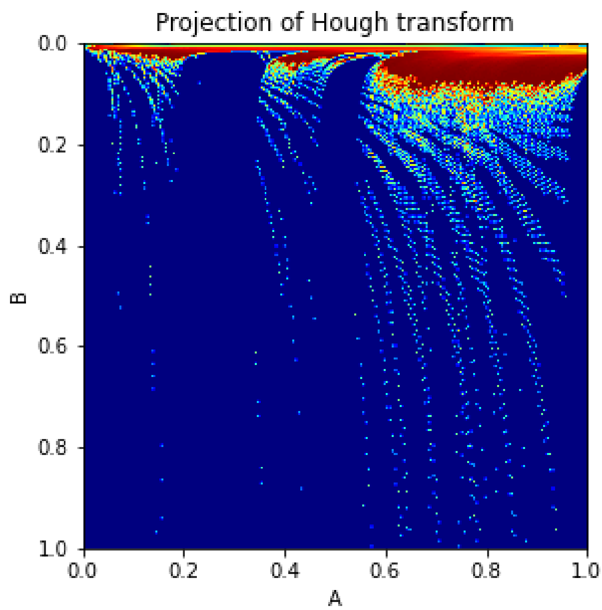
- ✓ Hough space parameters are minimized by selecting m based on a suitable assumption of the detectable shape. Here we selected **m=4.**
- ✓ A **threshold value of 125** was choses for the pixels in the Sobel edge image to be considered.
- ✓ Based on the above conditioning, Hough space parameters A and B were calculated.
- ✓ The final **A and B** are finally re-mapped to the image space to plot the lame curve overlaying in the sternum edges.

CT image 1 – Hough transform

Show below is the input Edge image of the (from the Sobel edge detector) and the 2D projection of its Hough transform space.

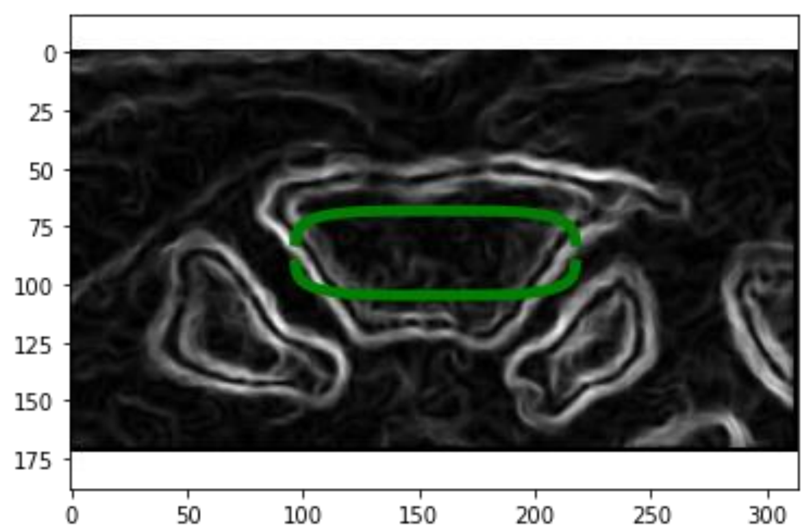
image shape: (173, 314)
shape of the accumulator: (360, 360)





Lame curve drawn over the image space corresponding to the highest voted point I the Hough space for CT image 1.

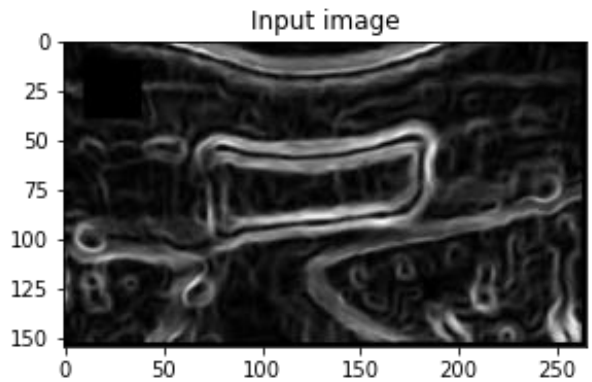
A: 60.55   B: 110842

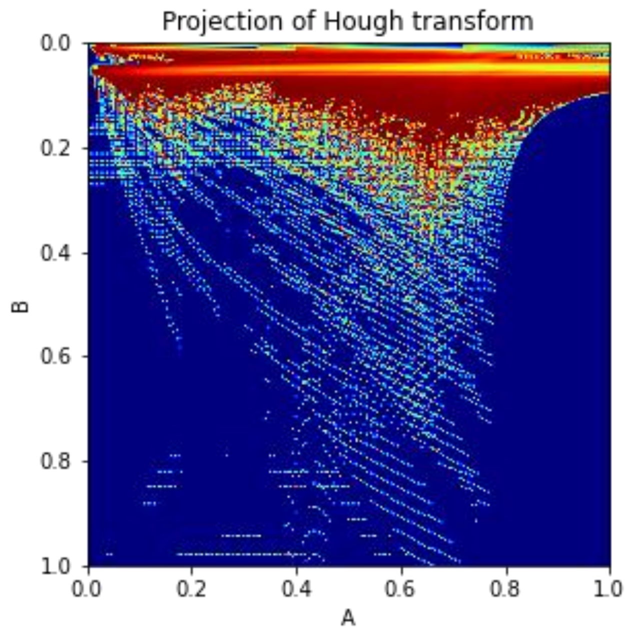


CT image 2 – Hough transform

Show below is the input Edge image of the (from the Sobel edge detector) and the 2D projection of its Hough transform space.

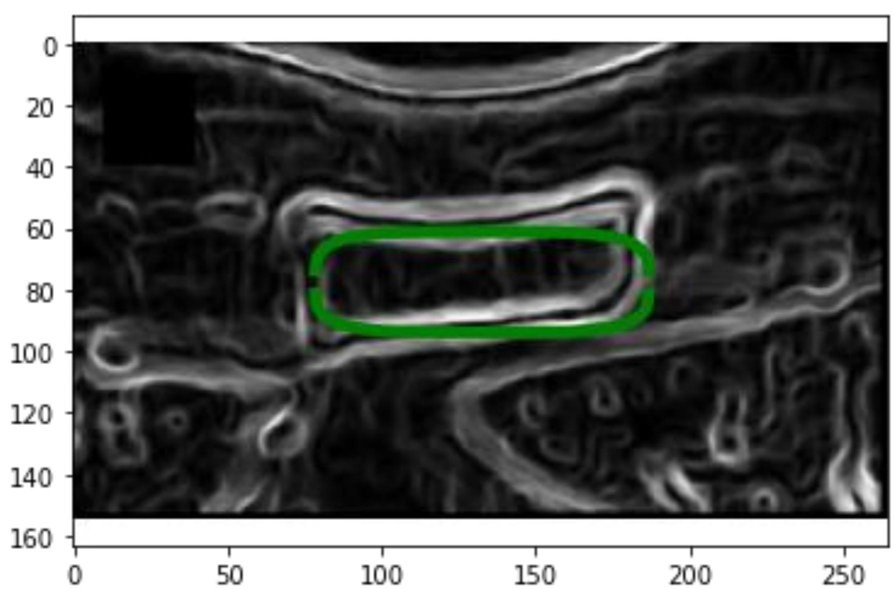
image shape: (155, 265)
shape of the accumulator: (308, 308)





Lame curve drawn over the image space corresponding to the highest voted point I the Hough space for CT image 2.

A: 54.25 B: 72345



References:

- Pattern recognition in medical imaging by means of the Hough transform of curves.
https://www.dima.unige.it/~perasso/files/Ispa_2013.pdf
- Lamé curve - https://en.wikipedia.org/wiki/Superellipse