CS 663: Assignment 1 Question 2

Yash Sanjeev - 180070068
Shubham Kar - 180070058
Rishav Ranjan - 180070045
Garaga V V S Krishna Vamsi - 180070020

September 1, 2020

Contents

1 Foreground Mask

To get a good mask, we first applied Gaussian filter to the original image and then applied thresholding to it.

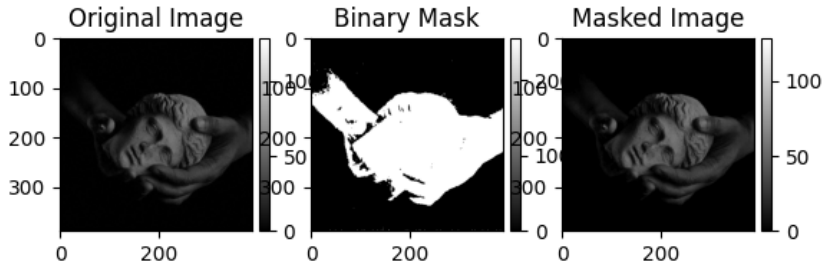

Figure 1: Original image, Binary mask and Masked image

2 Linear Contrast Stretching

Formulae for the linear function is provided below and the graph for the function in the domain is attached too:
if x is lying between 0 and 1 , then

$$f(x) = \begin{cases} \frac{x}{5}, & 0 \leq x < 0.3 \\ \frac{(x-0.3)*0.88}{0.4} + 0.06, & 0.3 \leq x \leq 0.7 \\ 1 - \frac{1-x}{5}, & 0.7 < x \leq 1 \end{cases}$$

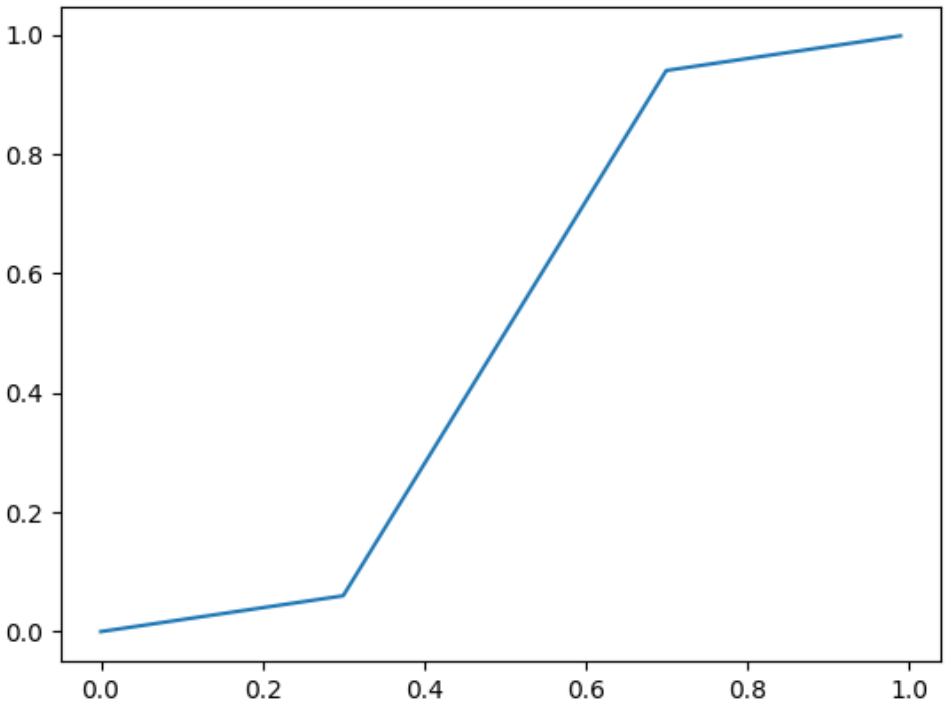

Figure 2: Linear Function

We performed Linear Contrast Stretching in the following images, and the output after applying Contrast Stretching are attached below.

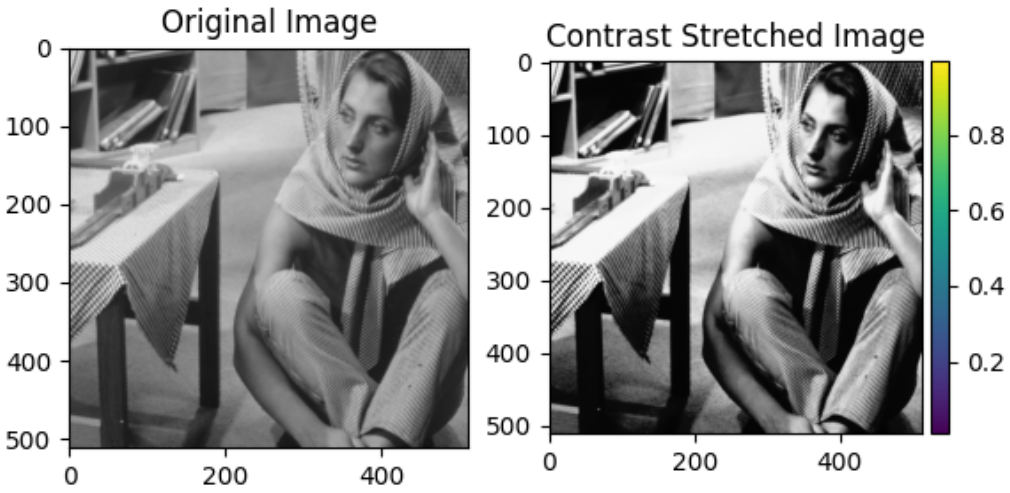

Figure 3: Original image and Contrast Enhanced Image

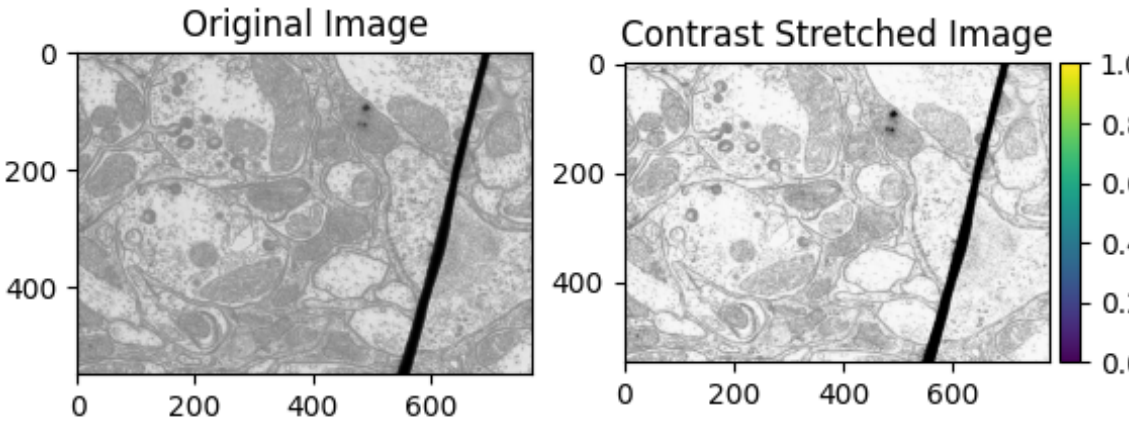

Figure 4: Original image and Contrast Enhanced Image

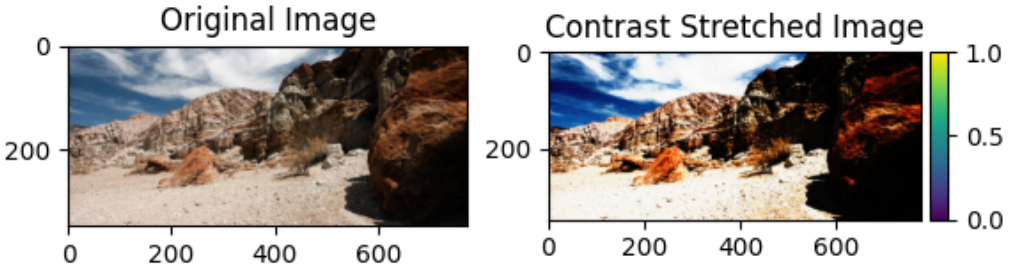

Figure 5: Original image and Contrast Enhanced Image

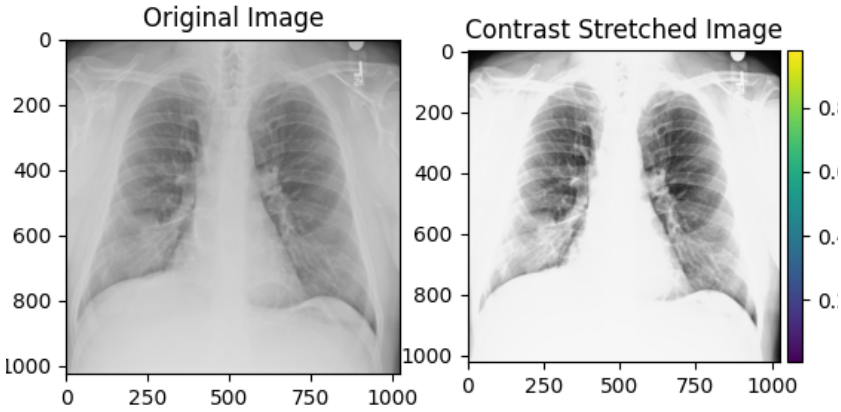

Figure 6: Original image and Contrast Enhanced Image

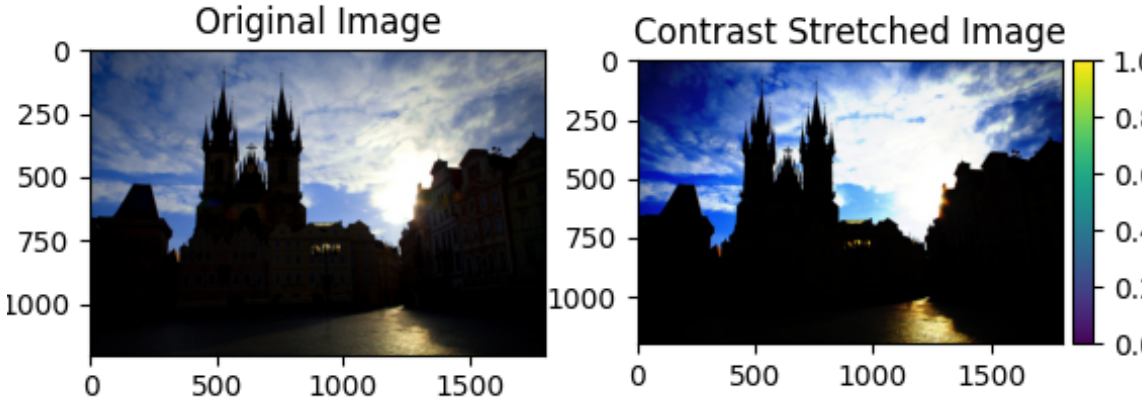

Figure 7: Original image and Contrast Enhanced Image

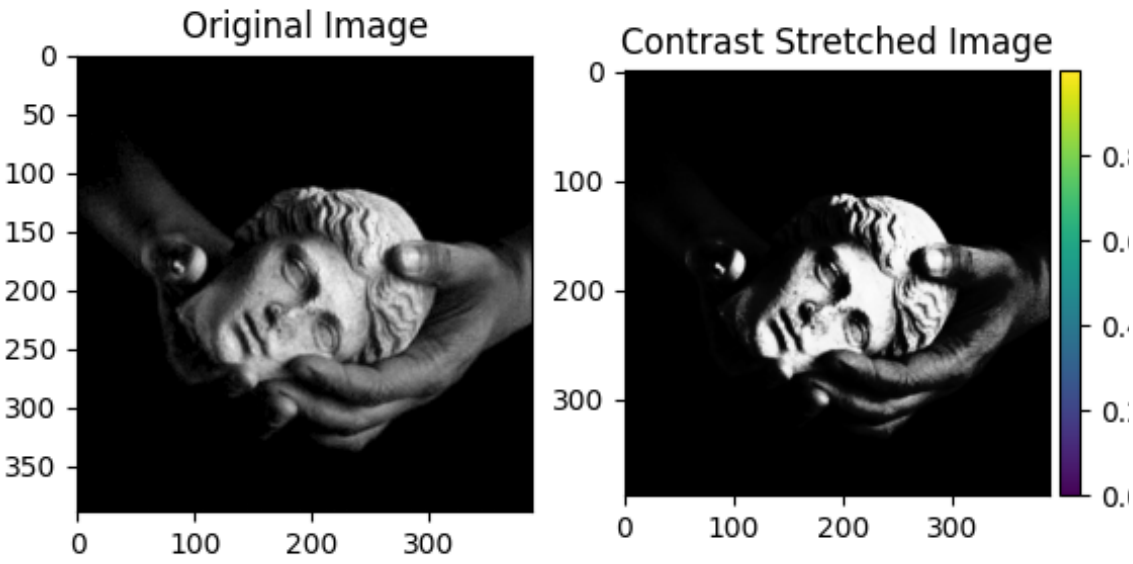

Figure 8: Original Image and Contrast Enhanced Image

Observation: When we apply Linear Contrast Stretching in Image 5, it is not quite effective. The reason is that most of the pixels of the image are dark. Hence if we see the linear function graph, the values at the corner brings values more closer and we see good effect of stretching in the middle region. But, since there are not many pixels having values in the middle region, stretching is not so effective here.

3 Histogram Equalization

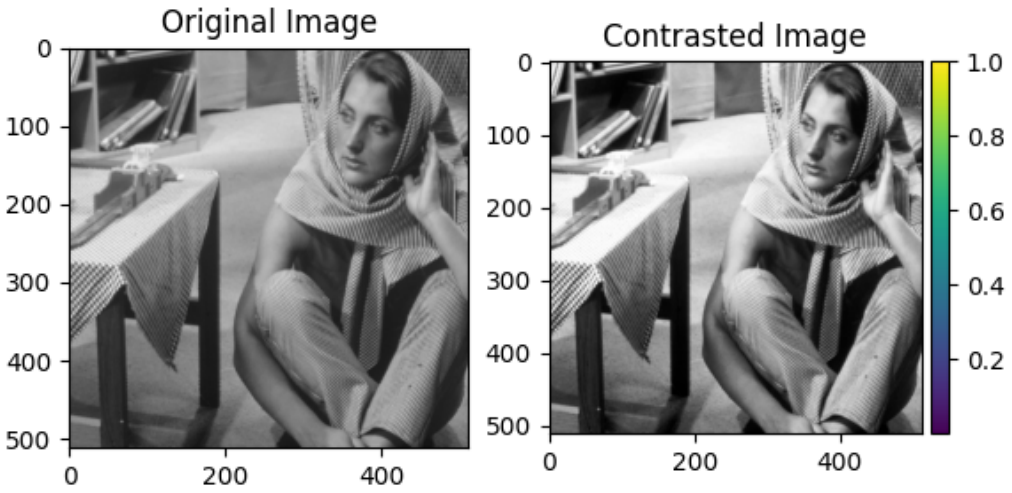

Figure 9: Original image and Contrast Enhanced Image by HE

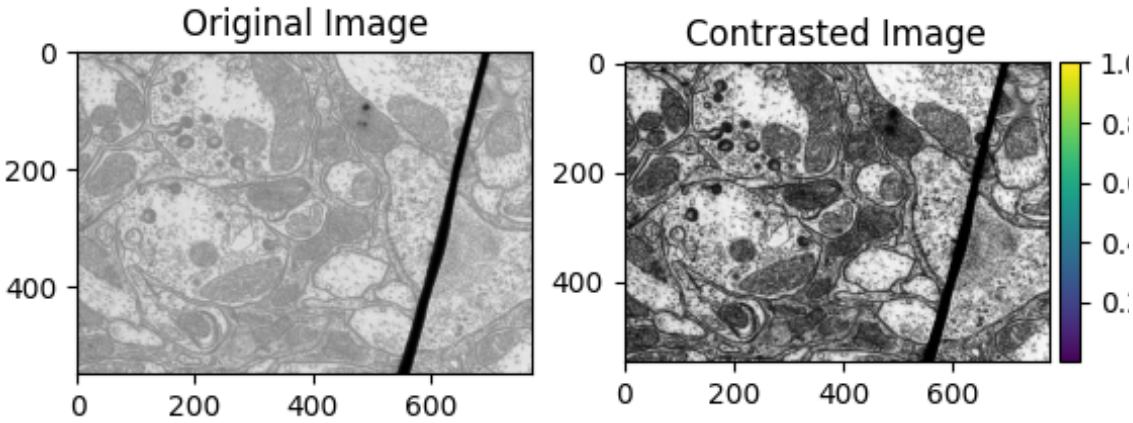

Figure 10: Original image and Contrast Enhanced Image by HE

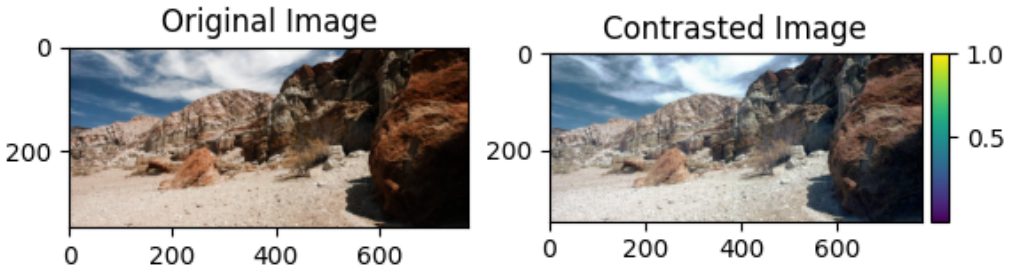

Figure 11: Original image and Contrast Enhanced Image by HE

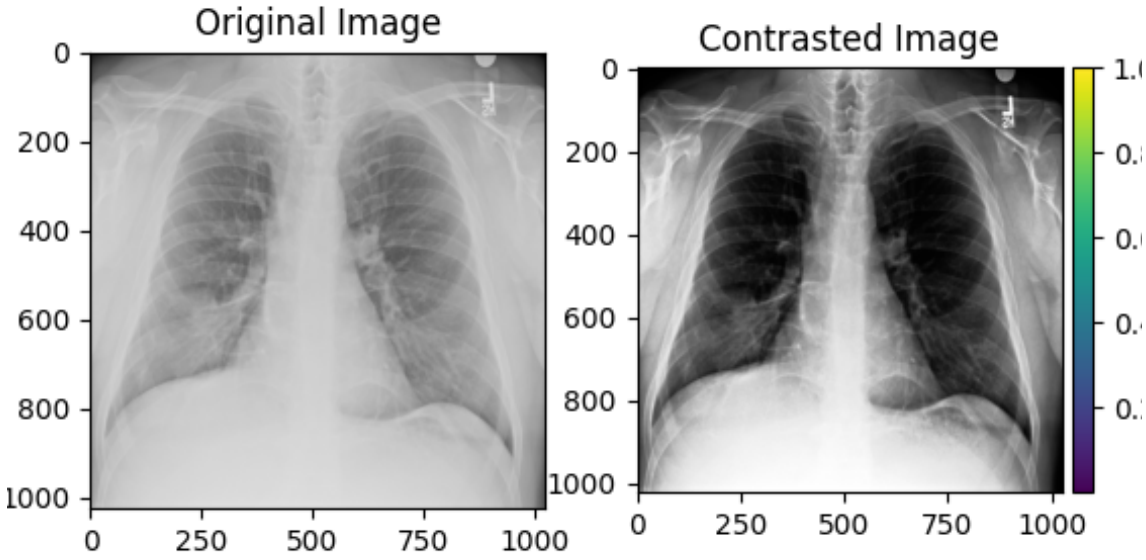

Figure 12: Original image and Contrast Enhanced Image by HE

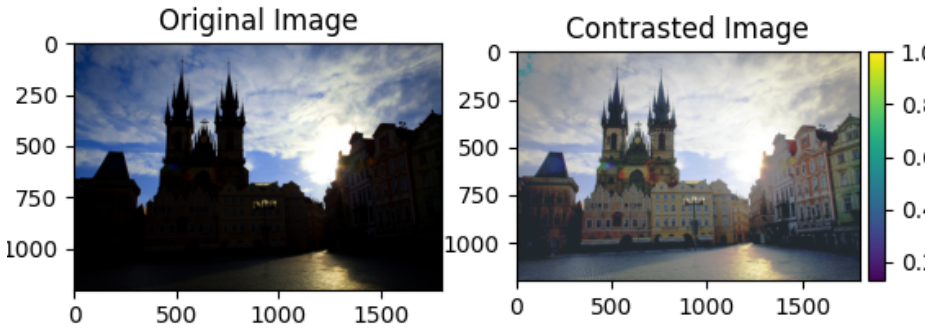

Figure 13: Original image and Contrast Enhanced Image by HE

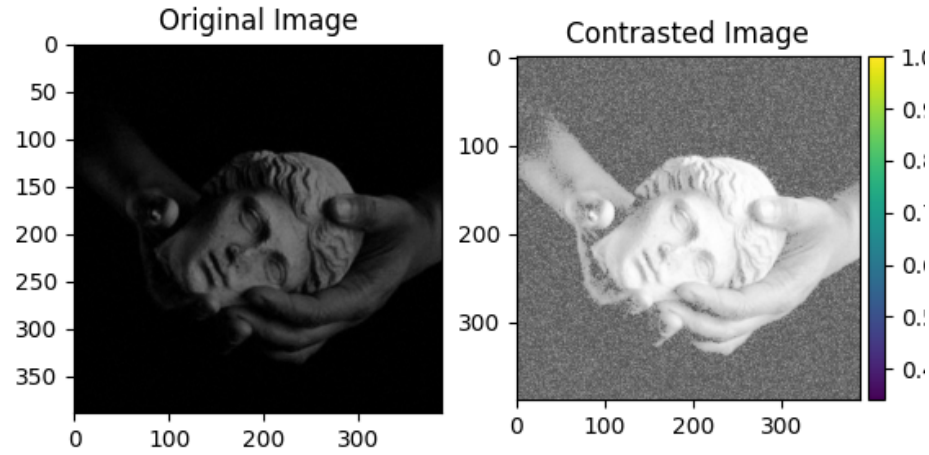

Figure 14: Original Image and Contrast Enhanced Image by HE

Observation: The contrast we achieve because of histogram equalization is far more better than the one we achieved because of linear contrast stretching. The image looks much appealing now. So, Histogram Equalization is obviously better than Linear Contrast Stretching.

4 Histogram Matching

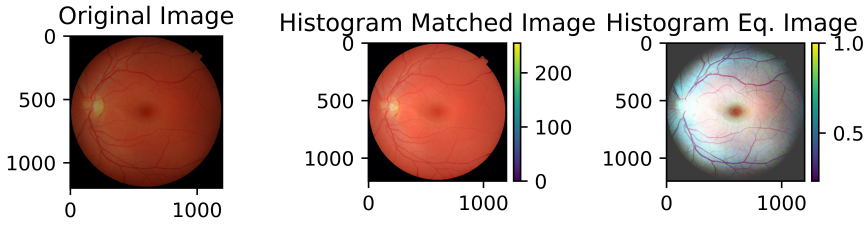

Figure 15: Original image, Histogram-Matched image and the Histogram-Equalised image

Observation: Histogram Matching focuses on details that were not so much glorified in the first image, but was distinctly glorified in the reference image.

5 Contrast-Limited Adaptive Histogram Equalization (CLAHE)

We implemented CLAHE using two Algorithms:

- **Exact CLAHE:** In this algorithm, first we pad the original image with a value '300' which is not in the range of 0 to 255 so that we can ignore these during the histogram calculation. Now, for every pixel in the original image, we consider a window of some optimal size(different for each picture) with the current pixel as center. Now we calculate the probability distribution threshold it and hence perform histogram equalisation for that window and then mapping the equalised value of the center to the output at that pixel. This algorithm takes very long time to execute in Python and hence we use Threading and hence this can be only run on Ubuntu.
- **Approximate CLAHE using Bi-Linear interpolation:** Here, we divide the whole image into m×n blocks of some window size (can be fine-tuned) and we perform the histogram equalisation same as above for each of these blocks and store the value at the center for each of these blocks. Now we perform Bi-Linear Interpolation to find the values of output image at other points. This is computationally faster than before but the approximation becomes worse as the window size increases. Also for the boundary elements, we just did a 1-D Linear interpolation of the 2 closest centers and for the four corners, we just took the value of the closest center. These approximations are clearly visible in the output images for larger windows.

Procedure

- Perform CLAHE and manually tune the window size and the histogram-threshold parameter to get the best contrasted output image.
- Redo the CLAHE with (i) Larger window size than optimal and (ii) Smaller window size than optimal.
- Redo the CLAHE with optimal window size and half threshold parameter
- This whole procedure is done for Barbara, TEM, Canyon and Chest X-ray images

Results
Below are the results for barbara.png for both the algorithms
1)Exact CLAHE

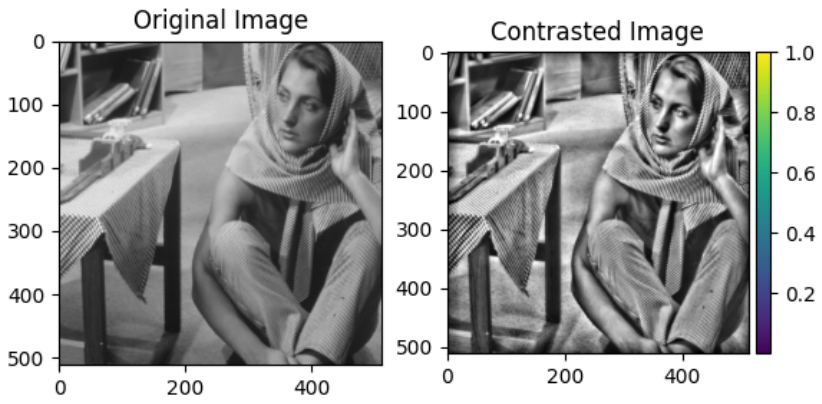

Figure 16: Optimal Parameters

Figure 17: Large window size

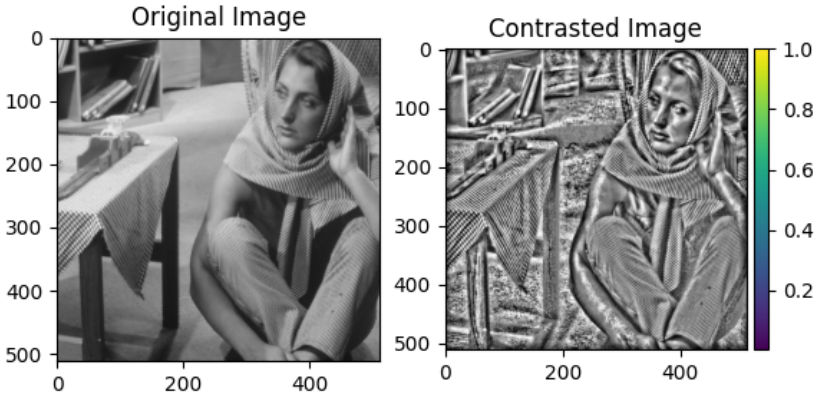

Figure 18: Small window size

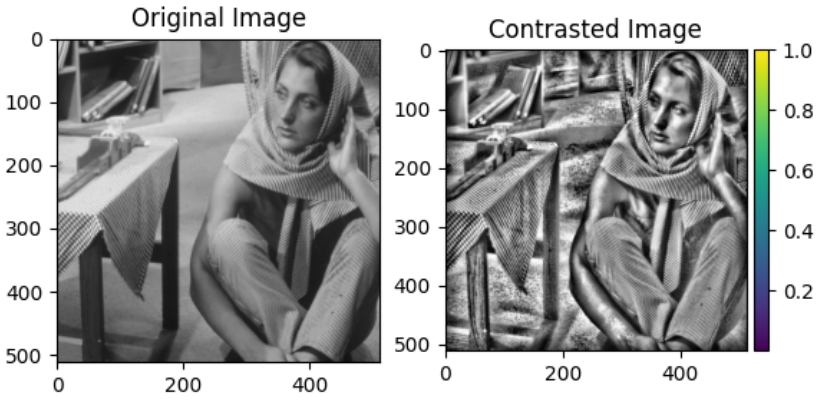

Figure 19: High threshold parameter

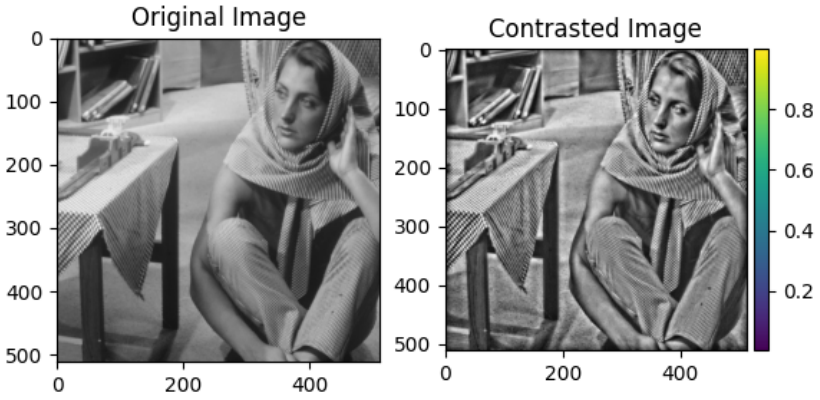

Figure 20: Low threshold parameter

2)CLAHE using Bilinear Interpolation

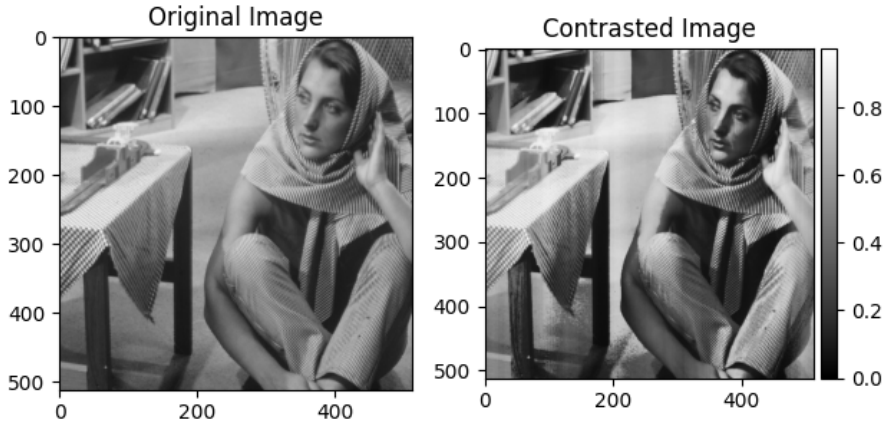

Figure 21: Optimal Parameters

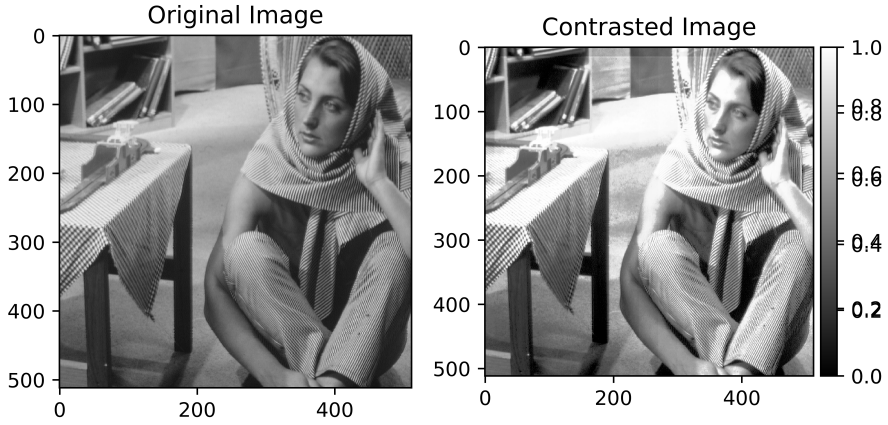

Figure 22: Large window size

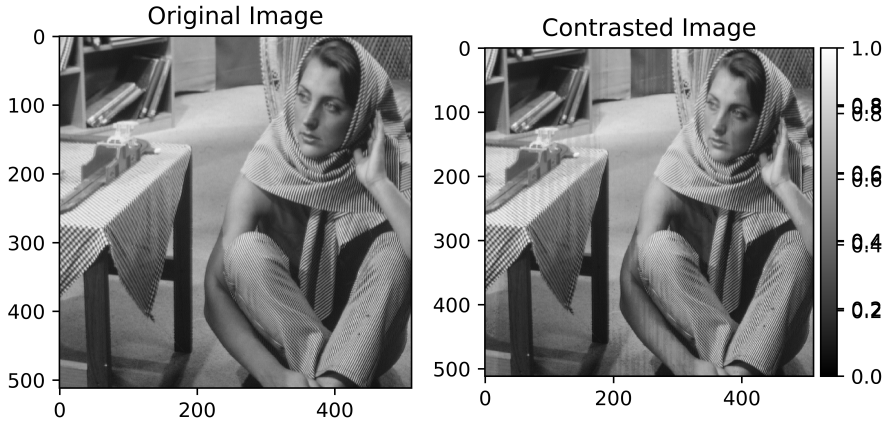

Figure 23: Small window size

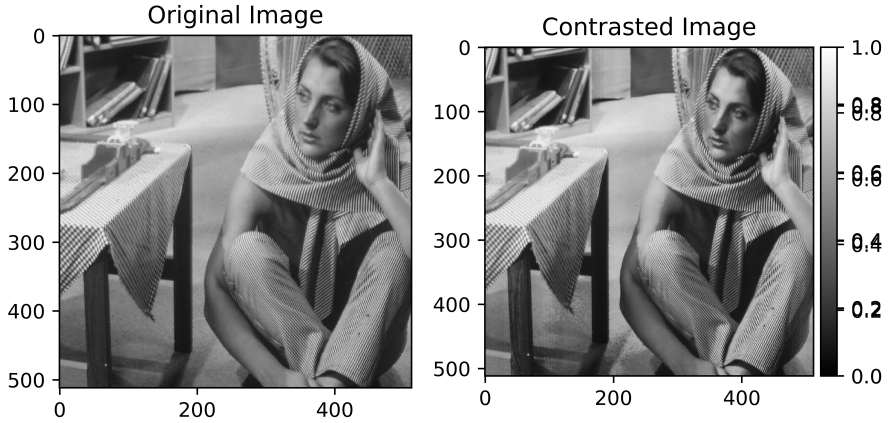

Figure 24: Half the threshold parameter

Observations

- **Large window size:** There will be low contrast improvement but very less noise amplification and hence almost approaches histogram equalisation
- **Small Window size:** There will be High contrast improvement but also very high noise amplification as the noise values are stretched at all windows throughout the image.
- **High Threshold Parameter:** No clipping happens and hence it reduces simple AHE and hence noise amplification is visible.
- **Low Threshold Parameter:** Histogram becomes uniform at all the regions and it leads to little or no contrast improvement. We can also see that the overall image becomes more dark.