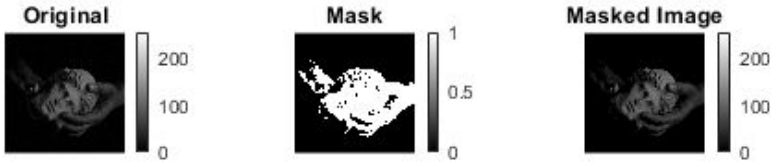
We are unable to publish the report due to memory out error while publishing. We are able to run the entire(execute) code and produce all the required images. Hence we are providing a custom report.

Contents:

%% MyMainScript
%% Part-(a) Foreground Mask using Thresholding
%% Part-(b) Linear Contrast Stretching
%% Part-(c) Histogram Equalization
%% Part-(d) Histogram Matching
%% Part-(e) Contrast-Limited Adaptive Histogram Equalization (CLAHE)
%% Part-(e - (i)) Original and CLAHE
%% Part-(e - (ii)) Original and CLAHE (Larger Window)
%% Part-(e - (ii)) Original and CLAHE (Smaller Window)
%% Part-(e - (iii)) Original and CLAHE (Half the Threshold)

```
%% MyMainScript
%   The main script for Question 2
tic;
```

```
%% Part-(a) Foreground Mask using Thresholding
%
threshold = 15;
[~,~] = myForegroundMask(threshold);
```



%% Part-(b) Linear Contrast Stretching

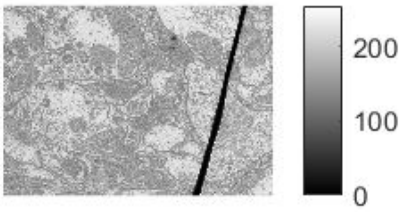
% Contrast stretching isn't effective on the image '../data/church.png'
% because, our contranst stretching is linear. In linear contrast
% stretching, one maps the range of intensities in each of the red, blue
% and green channels to the complete range [0,255]. If it so happens that
% even one pixel of the channel has a 0 intensity and one pixel has 255
% intensity, there will be no contrast stretching happening. This can be
% disadvantageous when we have very few pixel at the extremes of the
% intensity range - as is the case with the church image.

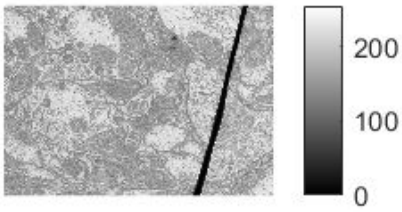
```
[image] = imread('../data/barbara.png');
[~] = myLinearContrastStretching(image);
[image] = imread('../data/TEM.png');
[~] = myLinearContrastStretching(image);
[image] = imread('../data/canyon.png');
[~] = myLinearContrastStretching(image);
[image] = imread('../data/church.png');
[~] = myLinearContrastStretching(image);
[image] = imread('../data/chestXray.png');
[~] = myLinearContrastStretching(image);
statue_img = imread('../data/statue.png');
```

```
threshold = 15;
mask = (statue_img>threshold);
masked_img = statue_img.*uint8(mask);
[~] = myLinearContrastStretching(masked_img);
```

200
100
0

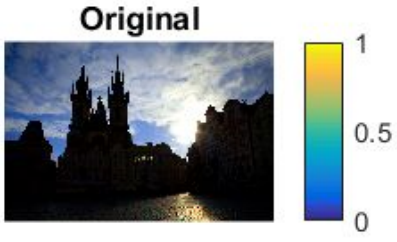
200
100
0


200
100
0


200
100
0

Original
1
0.5
0

Linear Contrast Stretched
1
0.5
0


Original
1
0.5
0

Linear Contrast Stretched
1
0.5
0

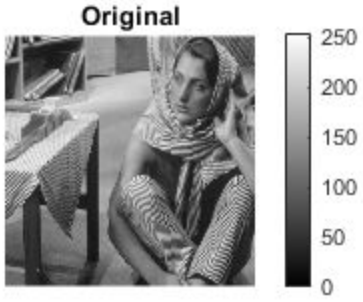
200
100
0

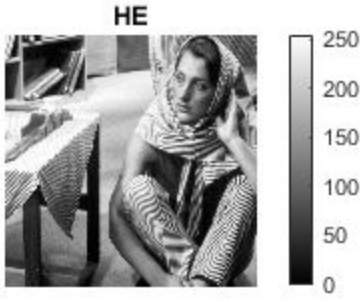
200
100
0

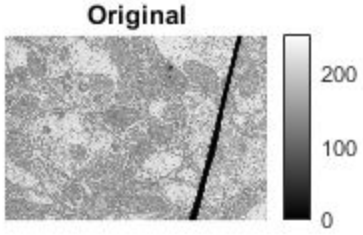
200
100
0

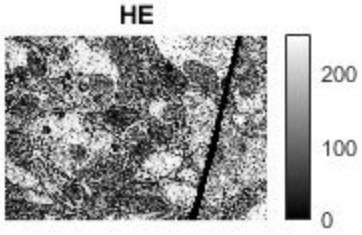
200
100
0

```
%% Part-(c) Histogram Equalization
%   The histogram equalization method used here works better than the
%   linear contrast stretching for the '../data/church.png' image because
%   the histogram equalization method can also perfrom non-linear contrast
%   stretching suitable for the image.
[image] = imread('../data/barbara.png');
[~] = myHE(image);
[image] = imread('../data/TEM.png');
[~] = myHE(image);
[image] = imread('../data/canyon.png');
[~] = myHE(image);
[image] = imread('../data/church.png');
[~] = myHE(image);
[image] = imread('../data/chestXray.png');
[~] = myHE(image);
statue_img = imread('../data/statue.png');
threshold = 15;
mask = (statue_img>threshold);
masked_img = statue_img.*uint8(mask);
[~] = myHE(masked_img);
```


Original
250
200
150
100
50
0


HE
250
200
150
100
50
0


Original
200
100
0

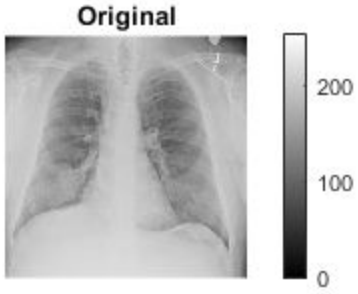

HE
200
100
0

Original
1
0.5
0

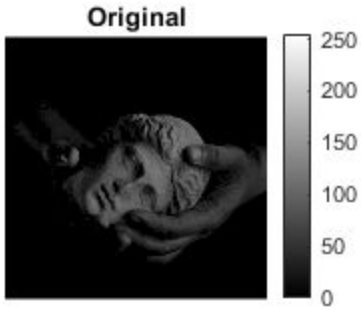
HE
1
0.5
0

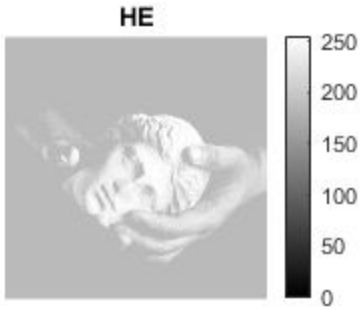
Original
1
0.5
0

HE
1
0.5
0

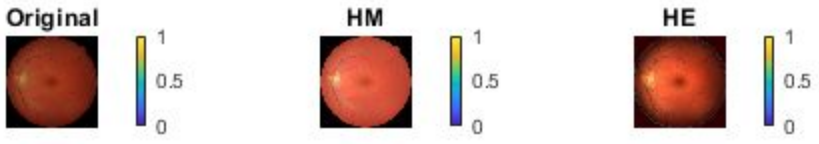

Original
200
100
0

HE
200
100
0


Original
250
200
150
100
50
0

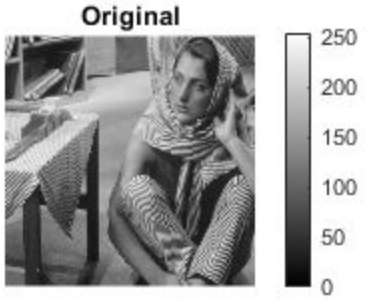

HE
250
200
150
100
50
0

%% Part-(d) Histogram Matching
%
matched_image = myHM();

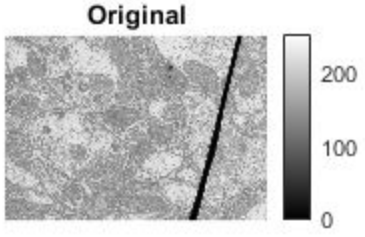


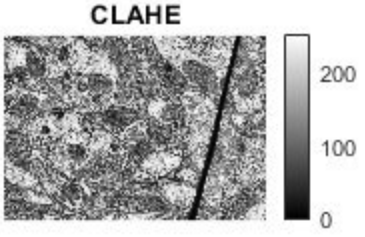
%% Part-(e) Contrast-Limited Adaptive Histogram Equalization (CLAHE)

%% Part-(e - (i)) Original and CLAHE
[image] = imread('../data/barbara.png');
[~] = myCLAHE(image, 225, 0.4);
[image] = imread('../data/TEM.png');
[~] = myCLAHE(image, 120, 0.5);
[image] = imread('../data/canyon.png');
[~] = myCLAHE(image, 130, 0.7);
[image] = imread('../data/chestXray.png');
[~] = myCLAHE(image, 180, 0.8);


Original
250
200
150
100
50
0

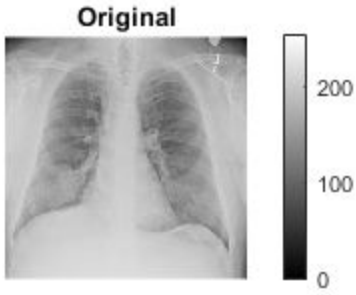
CLAHE
250
200
150
100
50
0


Original
200
100
0


CLAHE
200
100
0

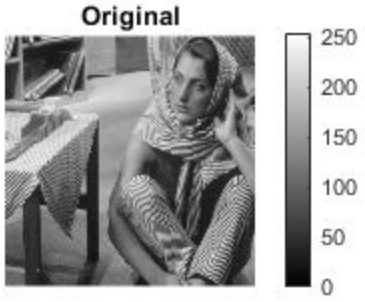
Original
1
0.5
0

CLAHE
1
0.5
0

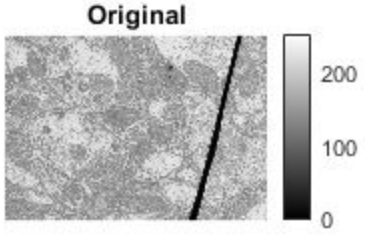

Original
200
100
0

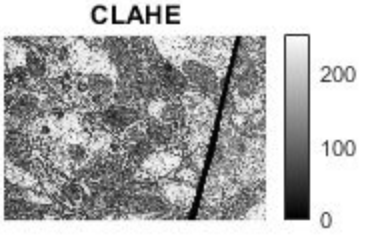
CLAHE
200
100
0

```
%% Part-(e - (ii)) Original and CLAHE (Larger Window)
%
[image] = imread('../data/barbara.png');
[~] = myCLAHE(image, 501, 0.4);
[image] = imread('../data/TEM.png');
[~] = myCLAHE(image, 501, 0.5);
[image] = imread('../data/canyon.png');
[~] = myCLAHE(image, 301, 0.7);
[image] = imread('../data/chestXray.png');
[~] = myCLAHE(image, 501, 0.8);
```


Original
250
200
150
100
50
0

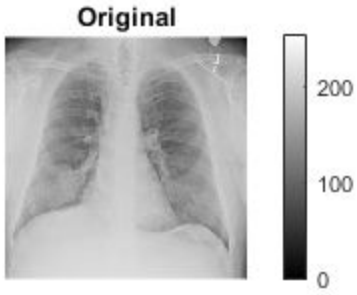
CLAHE
250
200
150
100
50
0

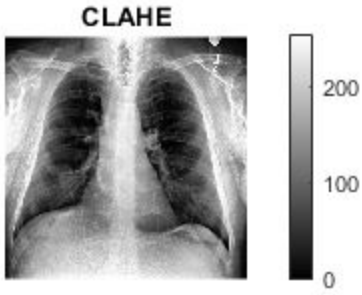

Original
200
100
0


CLAHE
200
100
0

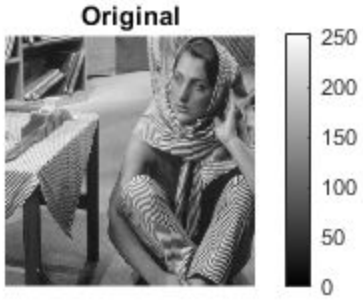
Original
1
0.5
0

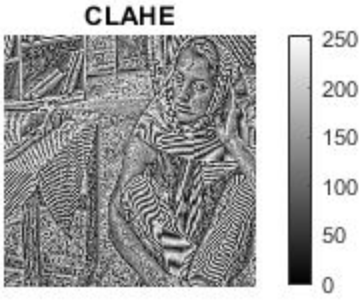
CLAHE
1
0.5
0

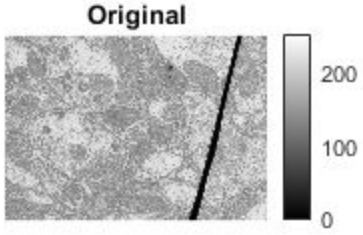

Original
200
100
0

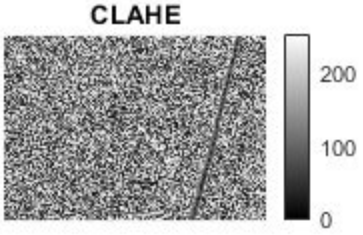

CLAHE
200
100
0

```
%% Part-(e - (ii)) Original and CLAHE (Smaller Window)
%
[image] = imread('../data/barbara.png');
[~] = myCLAHE(image, 10, 0.4);
[image] = imread('../data/TEM.png');
[~] = myCLAHE(image, 10, 0.5);
[image] = imread('../data/canyon.png');
[~] = myCLAHE(image, 10, 0.7);
[image] = imread('../data/chestXray.png');
[~] = myCLAHE(image, 10, 0.8);
```


Original
250
200
150
100
50
0

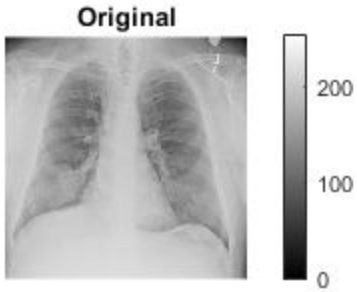

CLAHE
250
200
150
100
50
0

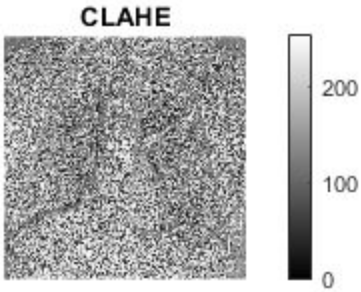

Original
200
100
0


CLAHE
200
100
0

Original
1
0.5
0

CLAHE
1
0.5
0

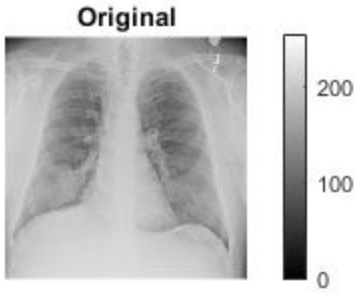

Original
200
100
0

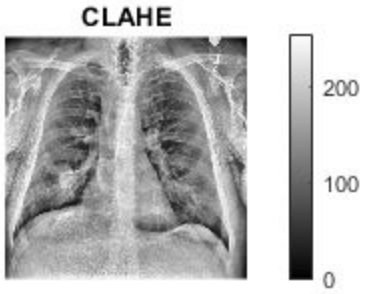

CLAHE
200
100
0

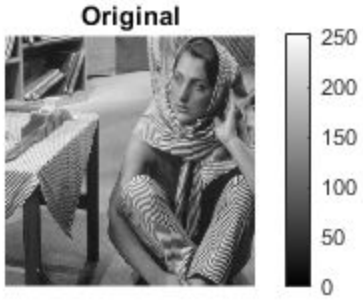
```
%% Part-(e - (iii)) Original and CLAHE (Half the Threshold)
%
[image] = imread('../data/barbara.png');
[~] = myCLAHE(image, 225, 0.2);
[image] = imread('../data/TEM.png');
[~] = myCLAHE(image, 120, 0.25);
[image] = imread('../data/canyon.png');
[~] = myCLAHE(image, 130, 0.35);
[image] = imread('../data/chestXray.png');
[~] = myCLAHE(image, 180, 0.4);
%%
toc
```

Original
1
0.5
0

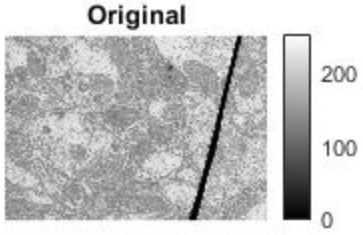
CLAHE
1
0.5
0

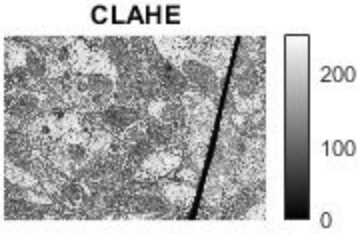

Original
200
100
0


CLAHE
200
100
0


Original
250
200
150
100
50
0

CLAHE
250
200
150
100
50
0


Original
200
100
0


CLAHE
200
100
0