Report for part 1

NOTE: For coloured images RGB channels were processed separately and then their resulting output images were concatenated

Q2) a) Linear Contrast Stretching

Algorithm Used:

Min and max Intensities with non zero frequencies were found using the histogram array of the input image and they were extrapolated to [0,255] {Min ->0; Max-> 255}. Other intensities were linearly called accordingly.

Pseudo Code:

Function [linearStretched] = myLinearContrastStretching(src)

minSrc = double(min(min(src)));

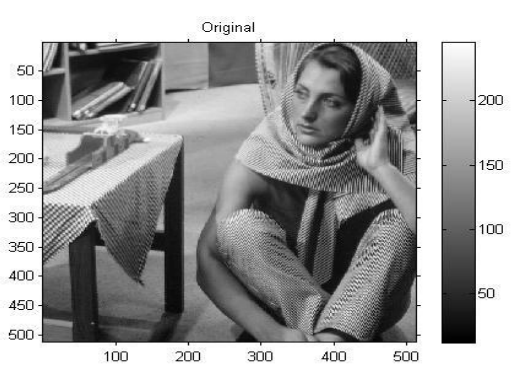
maxSrc = double(max(max(src)));

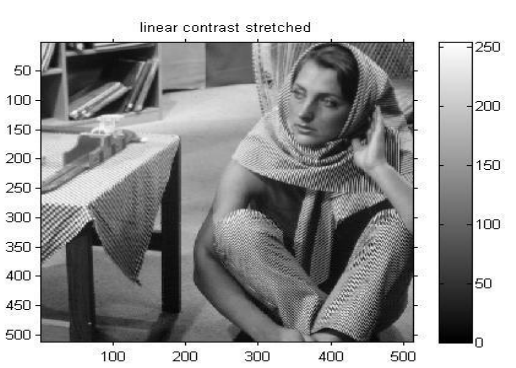
[m, n] = size (src);

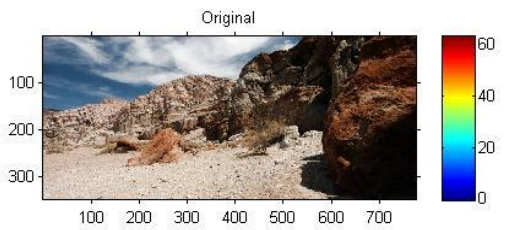
linearStretched = zeros(m,n);

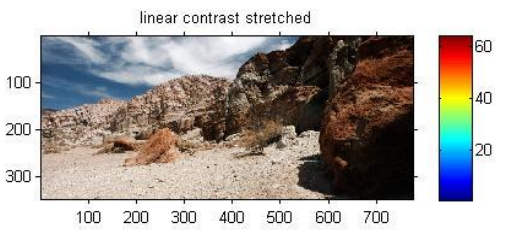
linearStretched = 255*(src – minSrc)/(maxSrc – minSrc);

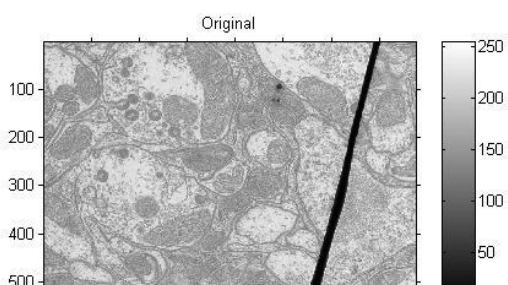
Result:

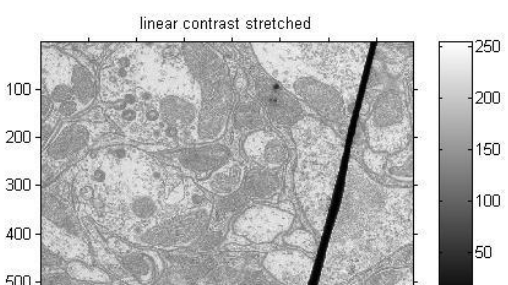





Original
100
200
300
100 200 300 400 500 600 700
60
40
20
0


linear contrast stretched
100
200
300
100 200 300 400 500 600 700
60
40
20


Original
100
200
300
400
500
100 200 300 400 500 600 700
250
200
150
100
50
0


linear contrast stretched
100
200
300
400
500
100 200 300 400 500 600 700
250
200
150
100
50
0

Q2) b) Histogram Equalization (HE)

Algorithm Used:

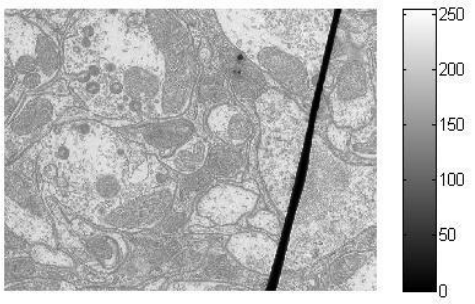
CDF of input image was found out and scaled to [0,255] and later intensity of the output pixels were assigned depending upon value in CDF function of intensity of corresponding pixel on input image.

Result:

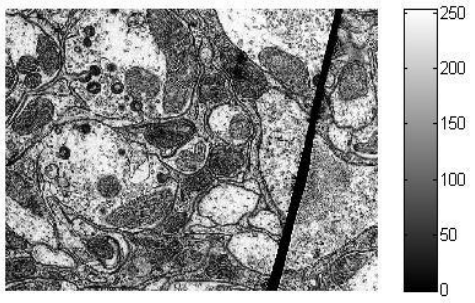
Original

histogram equalised

Original



histogram equalised



Q2) c) Adaptive Histogram Equalization (AHE)

Algorithm used:

Histogram equalisation were carried out on NxN window around every pixel to find the value of intensity of the center pixel.

N values were tuned for the best results. NL-filter was used to slide the window over input image.

Findings:

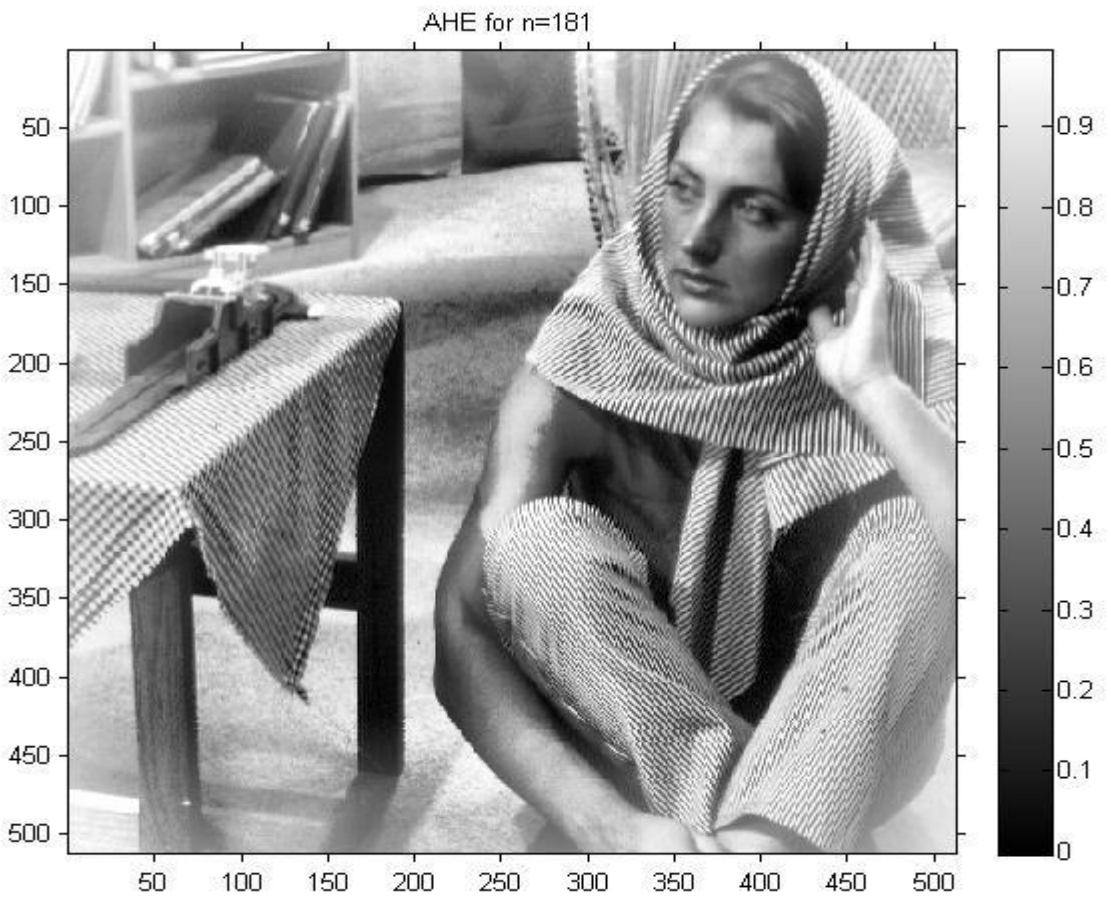
N values were arbitrarily varied from 20 to 200 to get the best result

In case of **“barabara.png”** the best results were found for **N=181**, for N > 181 we observe low contrast enhancement and lesser noise, while for N<181 we observe more contrast enhancement and more noise.

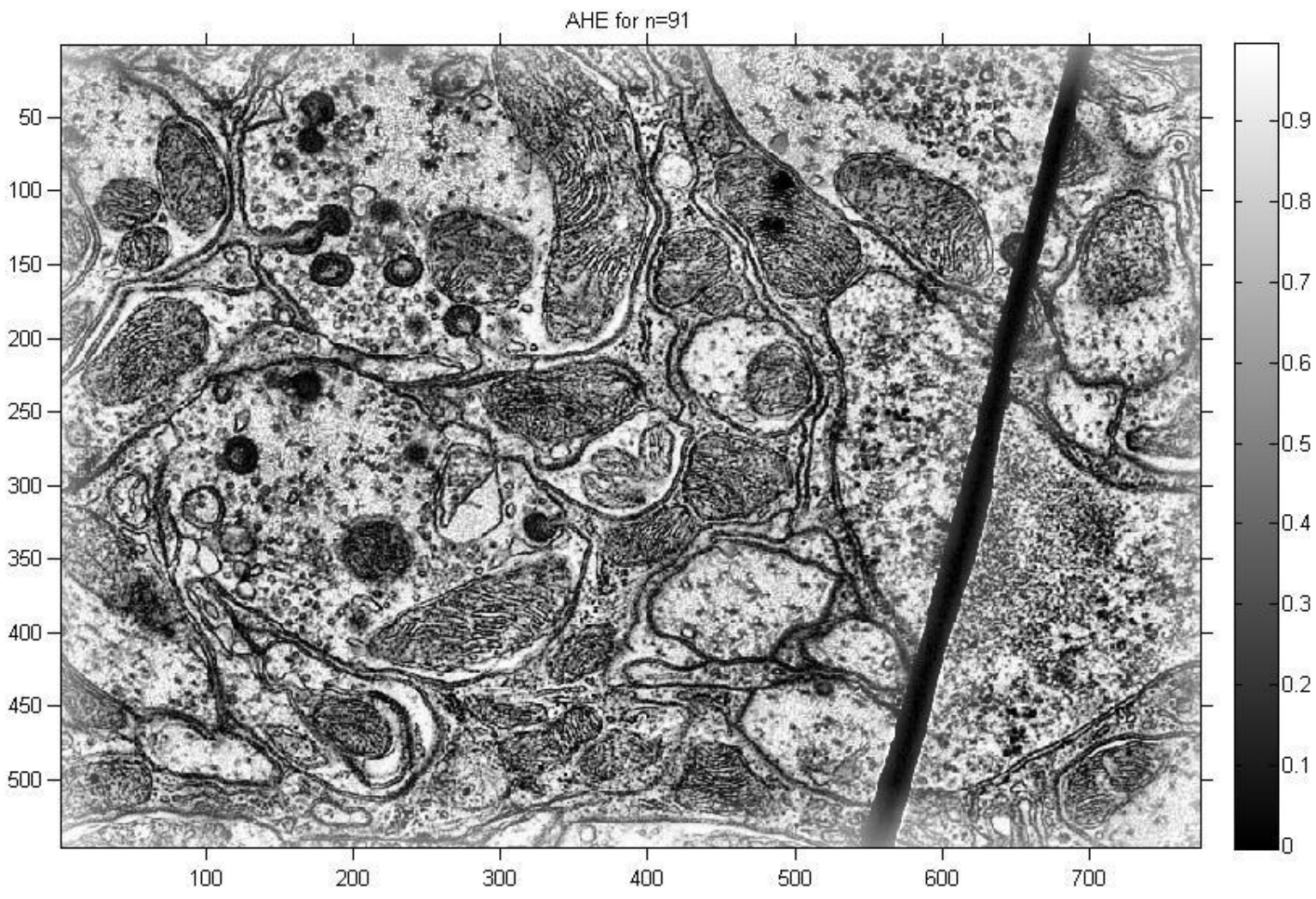
In case of **“TEM.png”** the best results were found for N=91, for N > 91 we observe low contrast enhancement and lesser noise, while for N<91 we observe more contrast enhancement and more noise.

In case of “**canyon.png**” the best results were found for N=161, for N > 161 we observe low contrast enhancement and lesser noise, while for N<161 we observe more contrast enhancement and more noise.

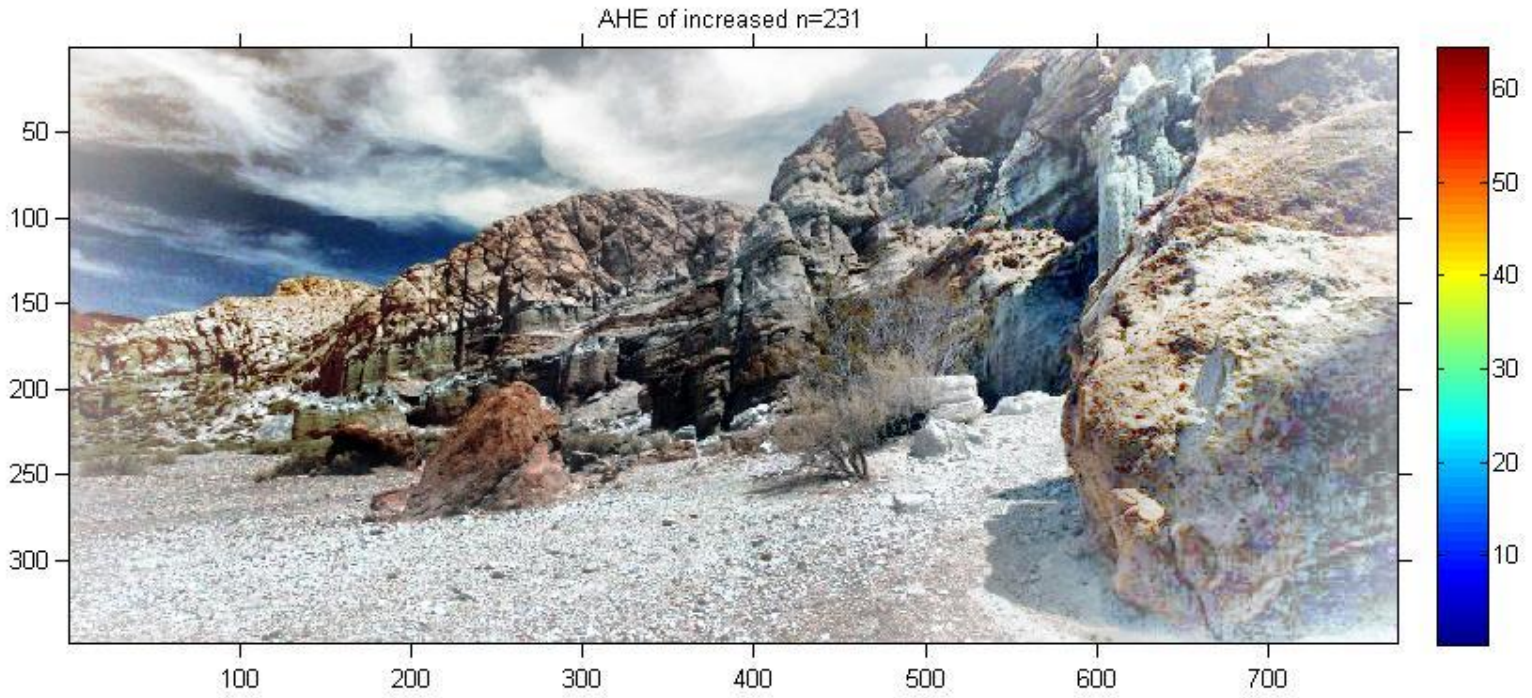
Ideal N:

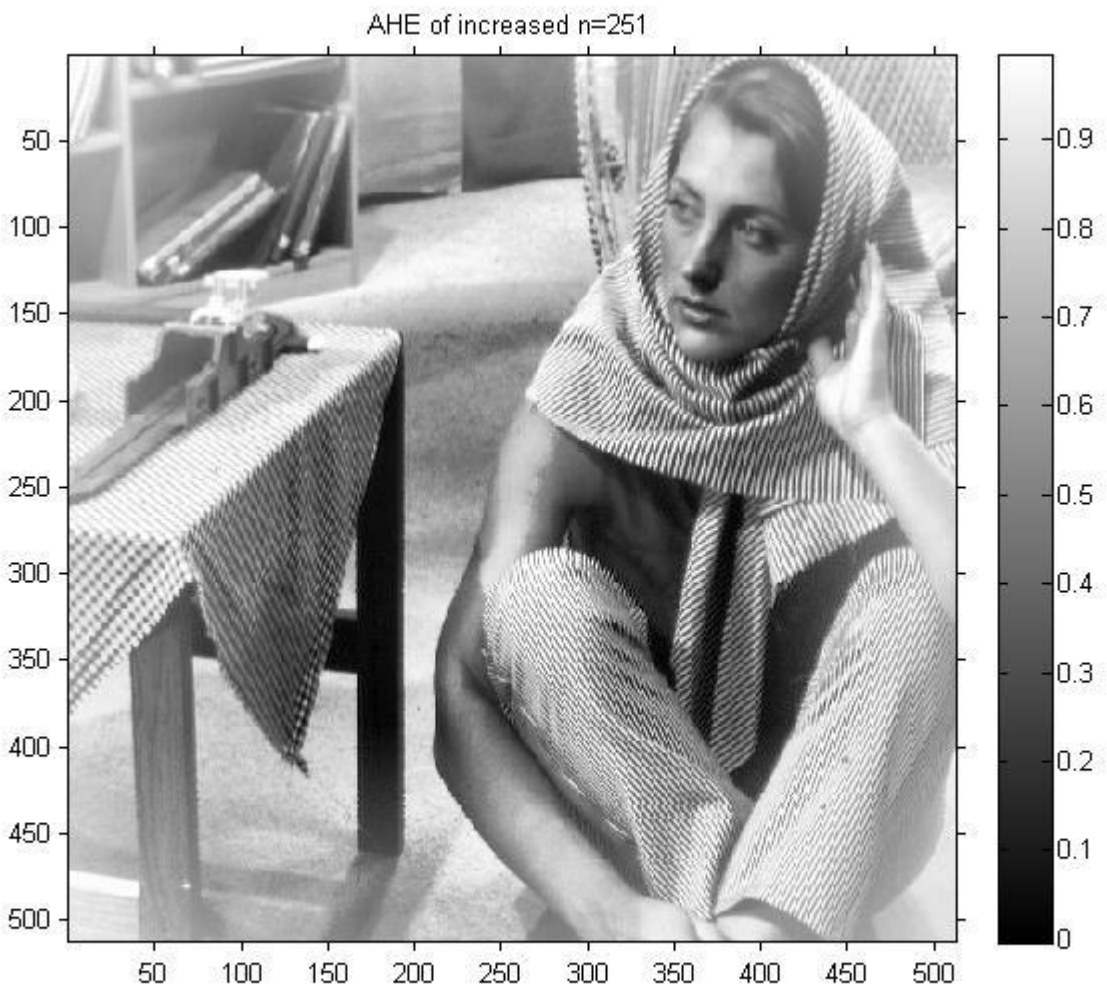


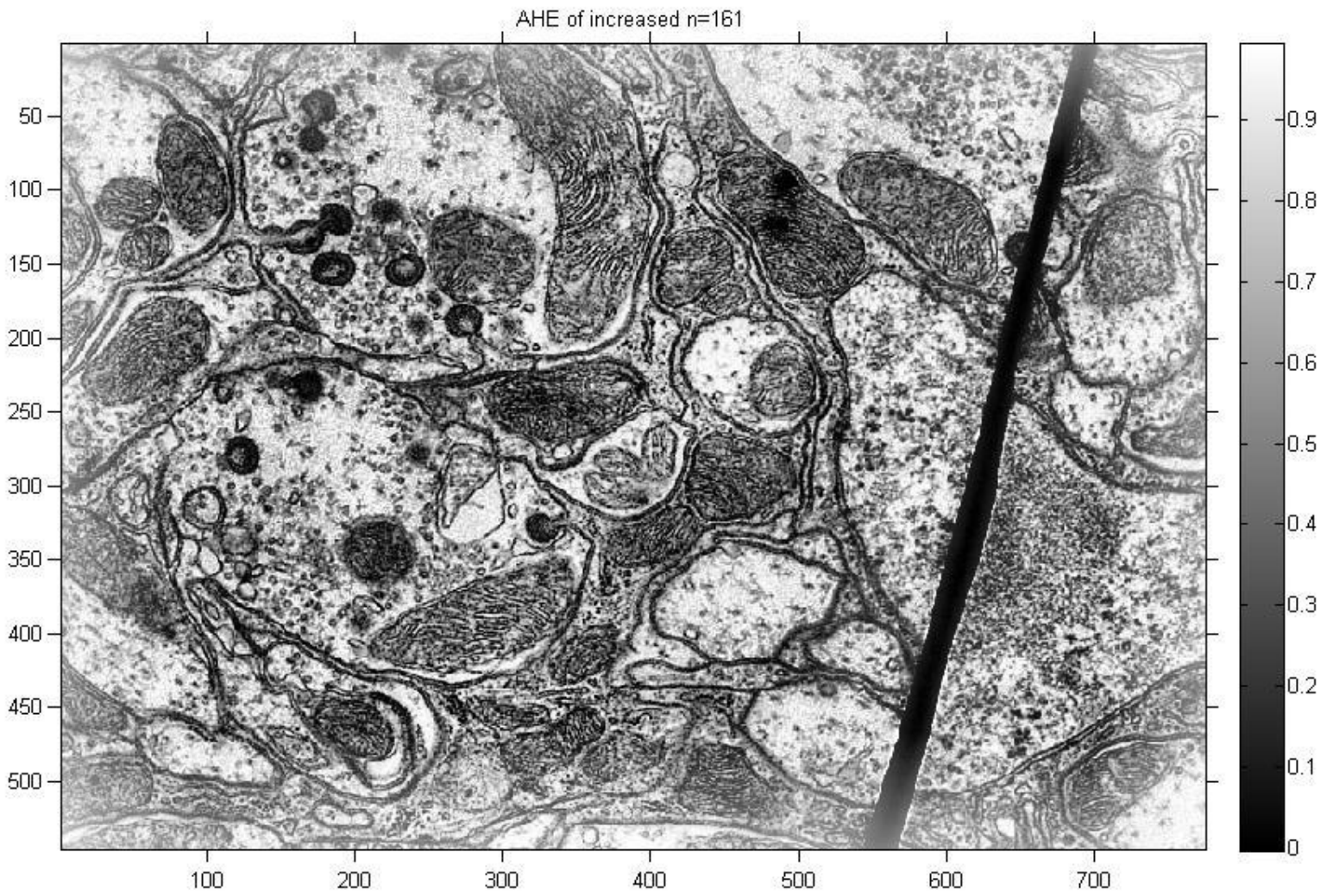

AHE of n=161
60
50
40
30
20
10


AHE for n=91
50
100
150
200
250
300
350
400
450
500
100
200
300
400
500
600
700
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0

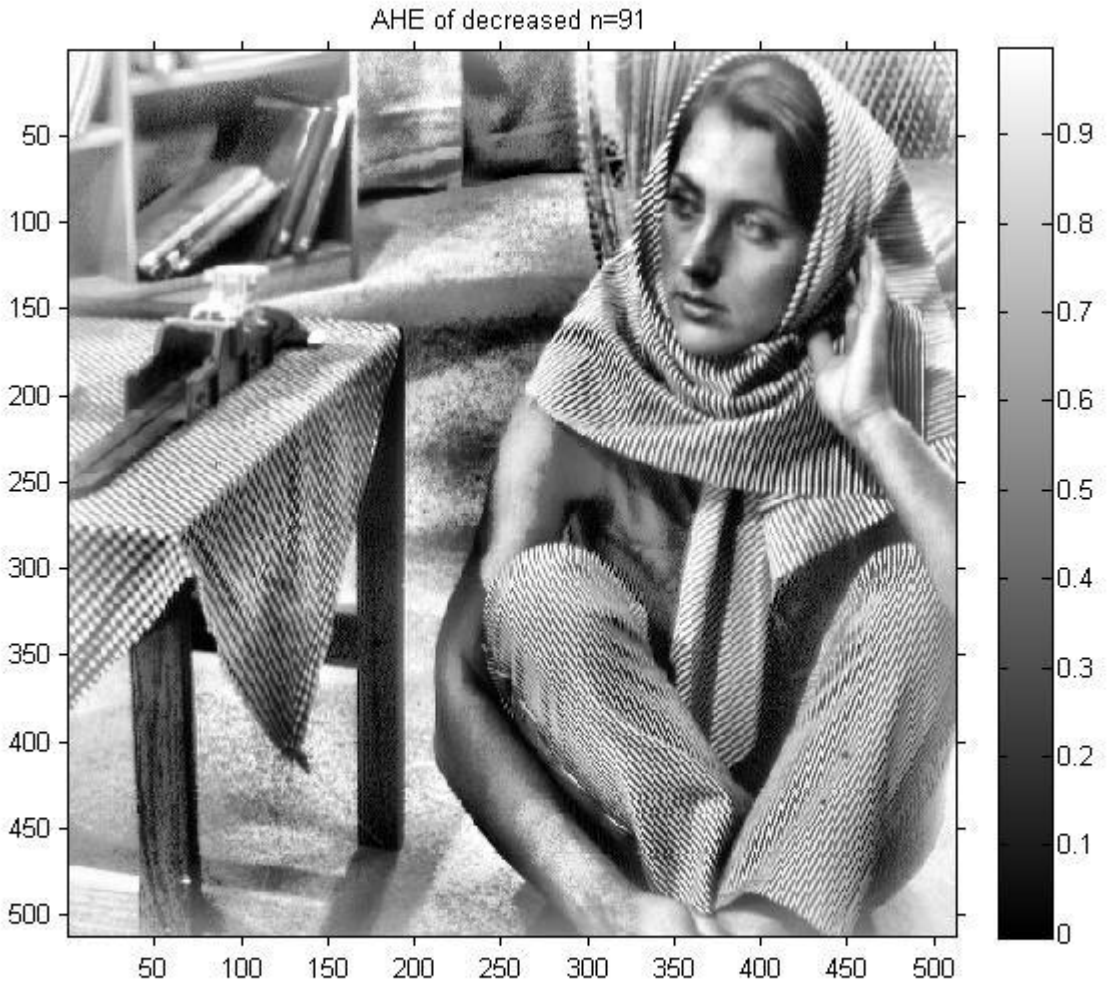
For Increased N:

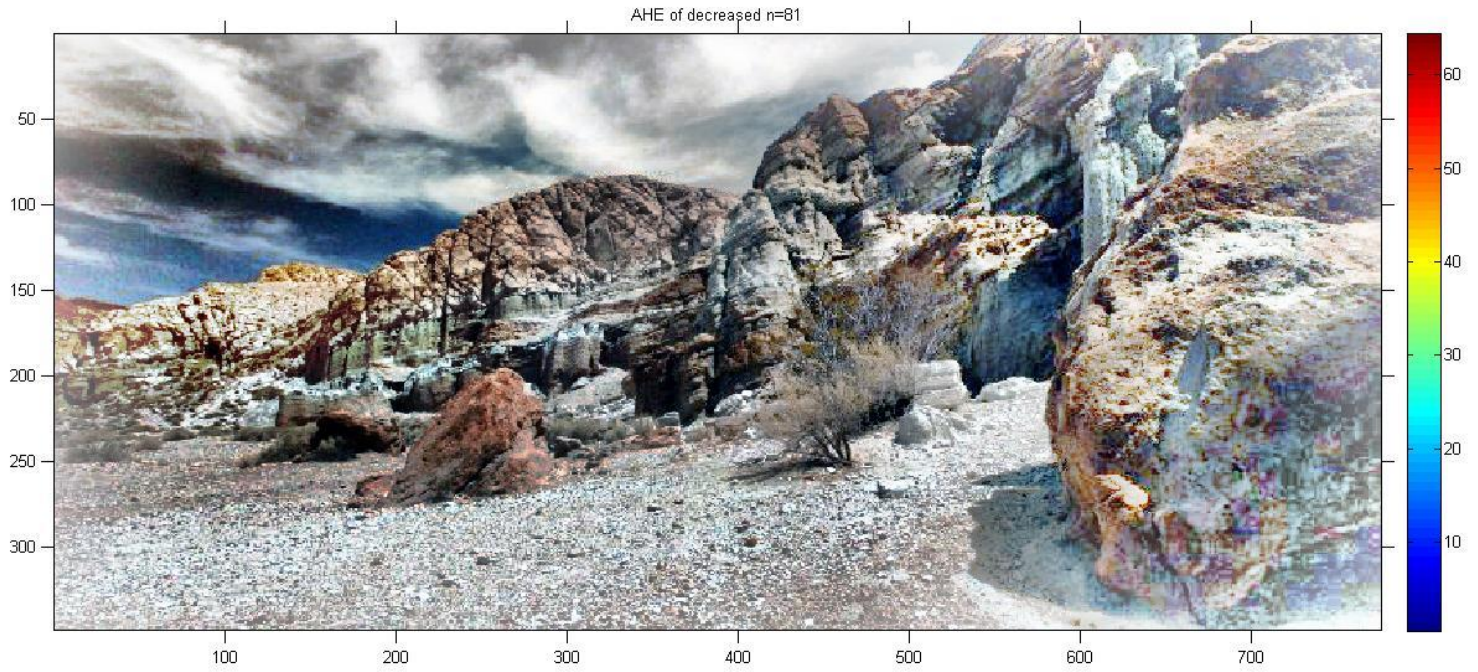


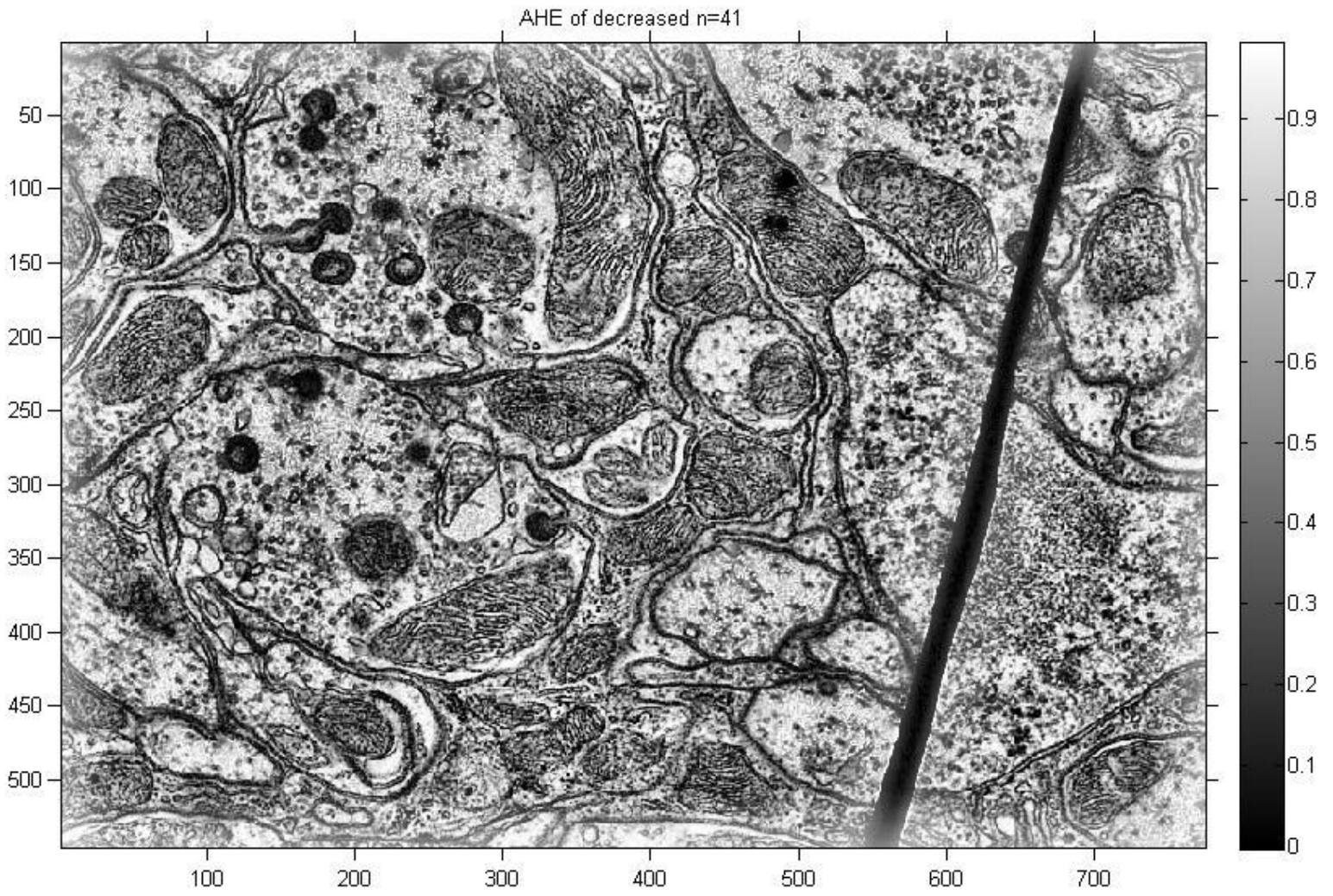




For Decreased N:







Q2) d) Contrast-Limited Adaptive Histogram Equalization (CLAHE)

Algorithm Used:

Histogram equalisation were carried out on NxN window. For the intensities having normalised frequency beyond certain threshold, extra mass was chopped off and distributed evenly among all the pixels. The value of the centre pixel in output window was assigned to the corresponding pixel. Nlfilter was used to slide the window over input image

Findings:

In case of white shaded image it was observed that the threshold values were on a lower side as compared to dark shade images.

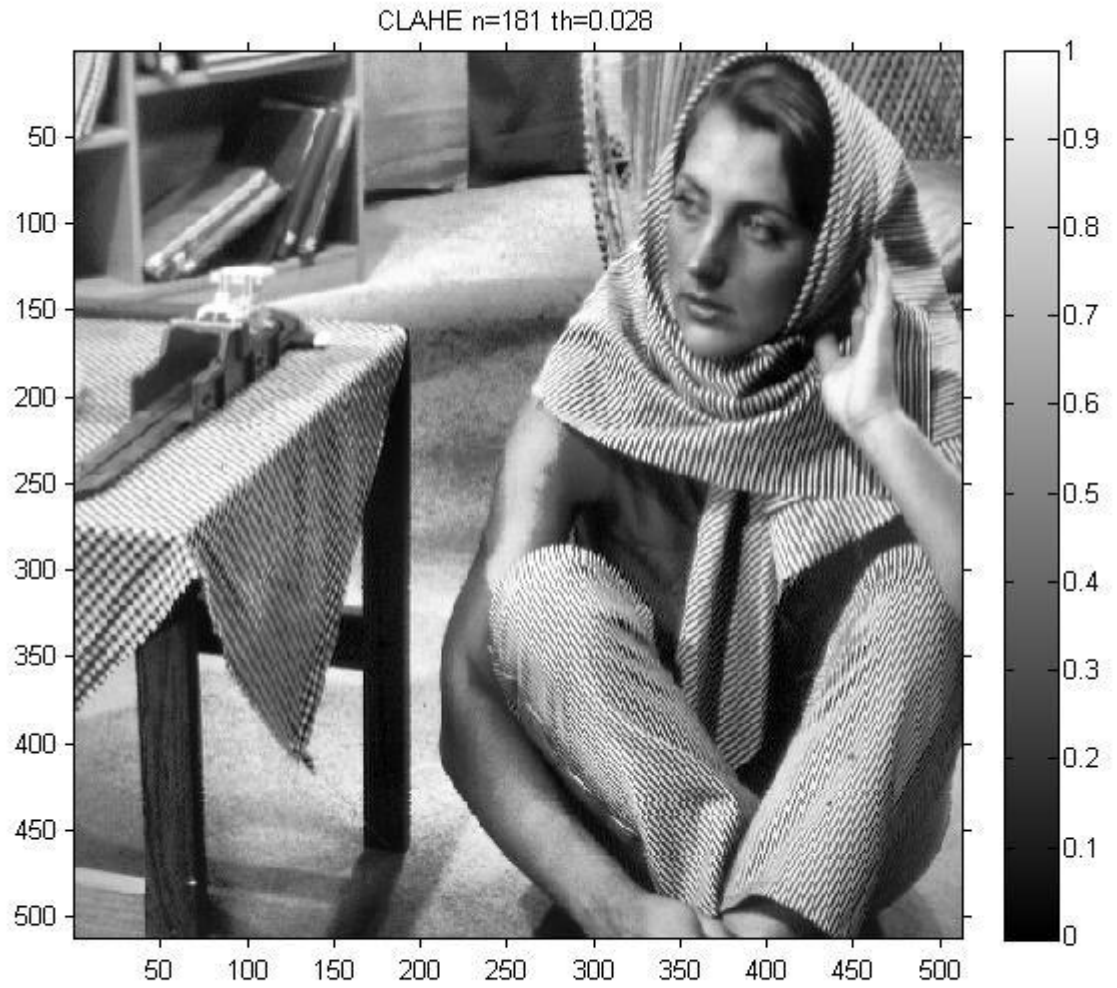
Hence we used the value of threshold = 0.018 for **"barabara.png"** and threshold = 0.028 for **"TEM.png"**. For the coloured "**canyon.png**" we observed that the best result was obtained for threshold = 0.035.

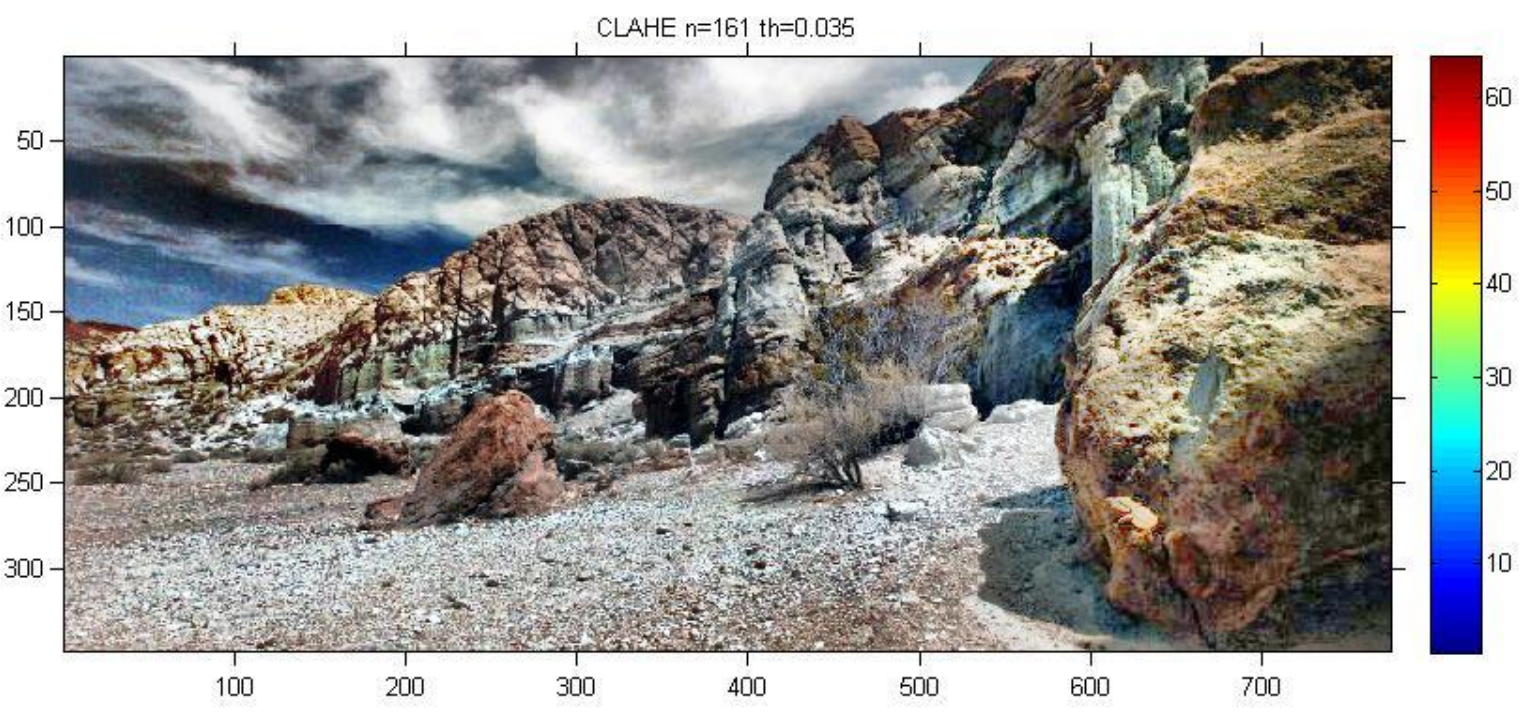
Comments:

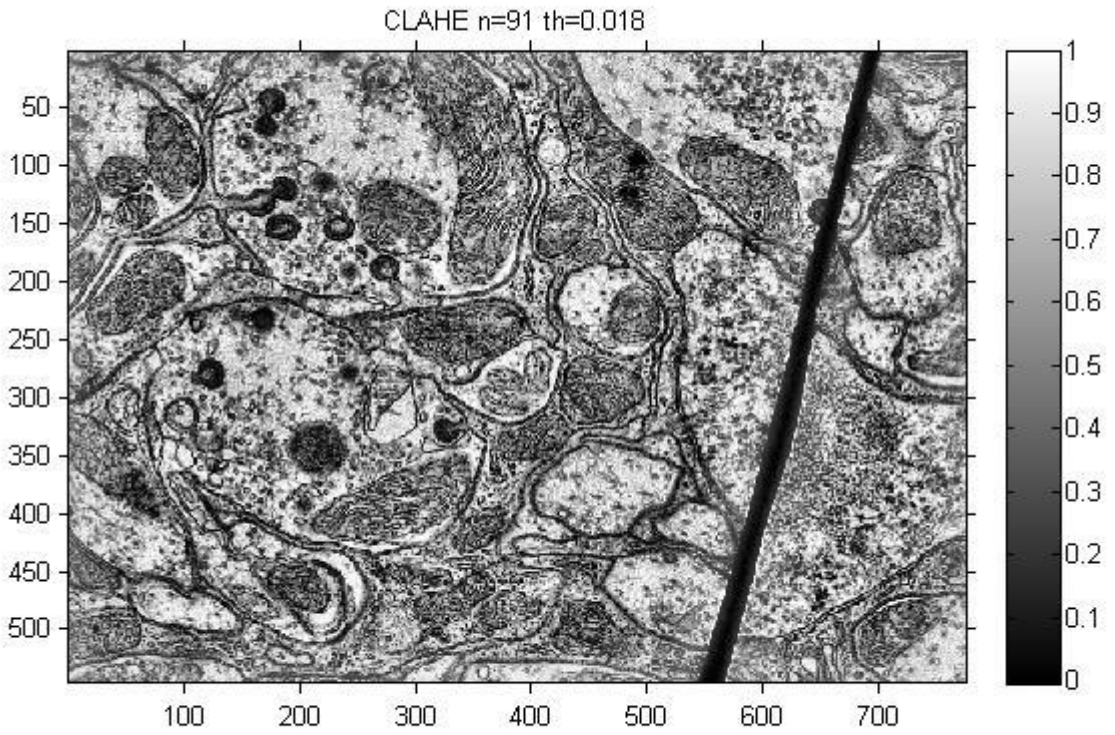
Sharp intensity peaks which were present in AHE are removed in CLAHE

Results:

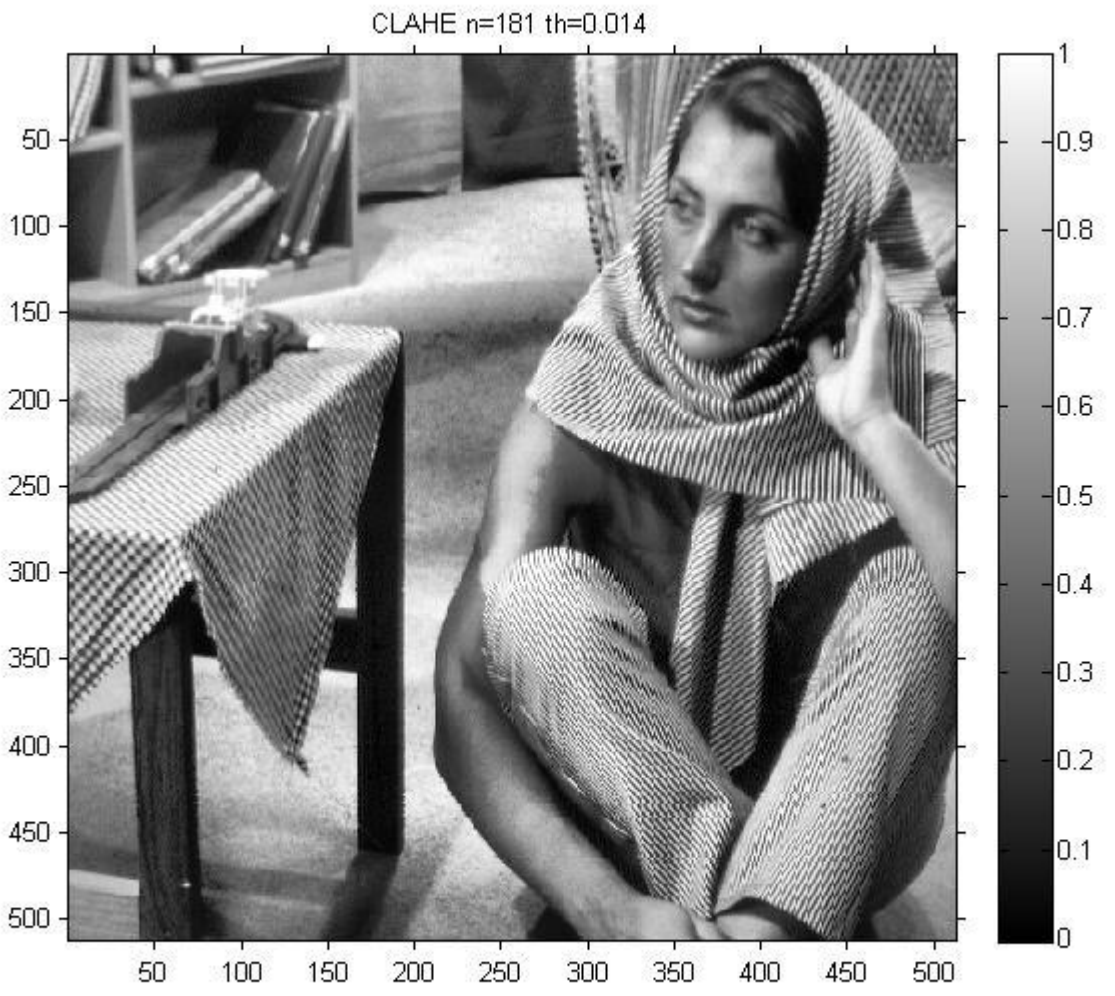
For Ideal N and Threshold Parameter:

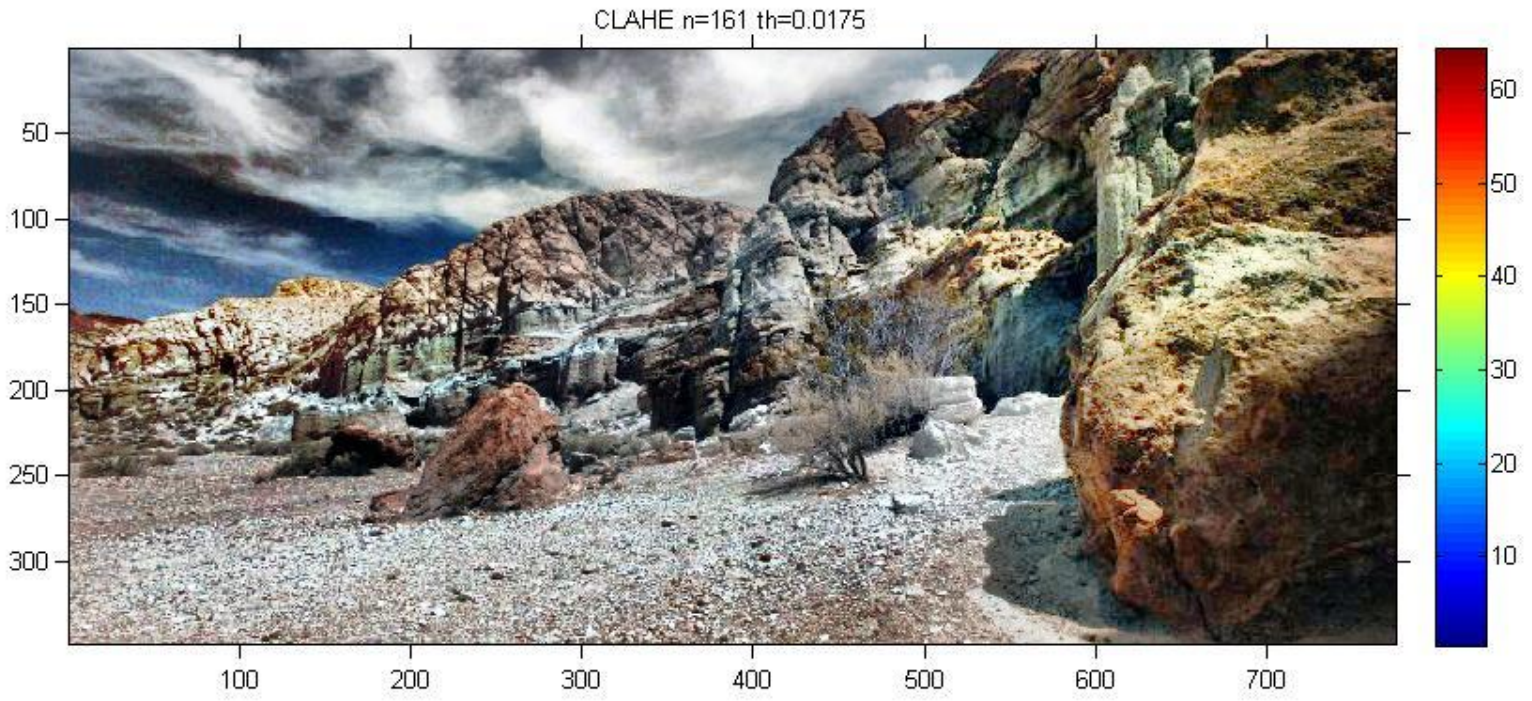


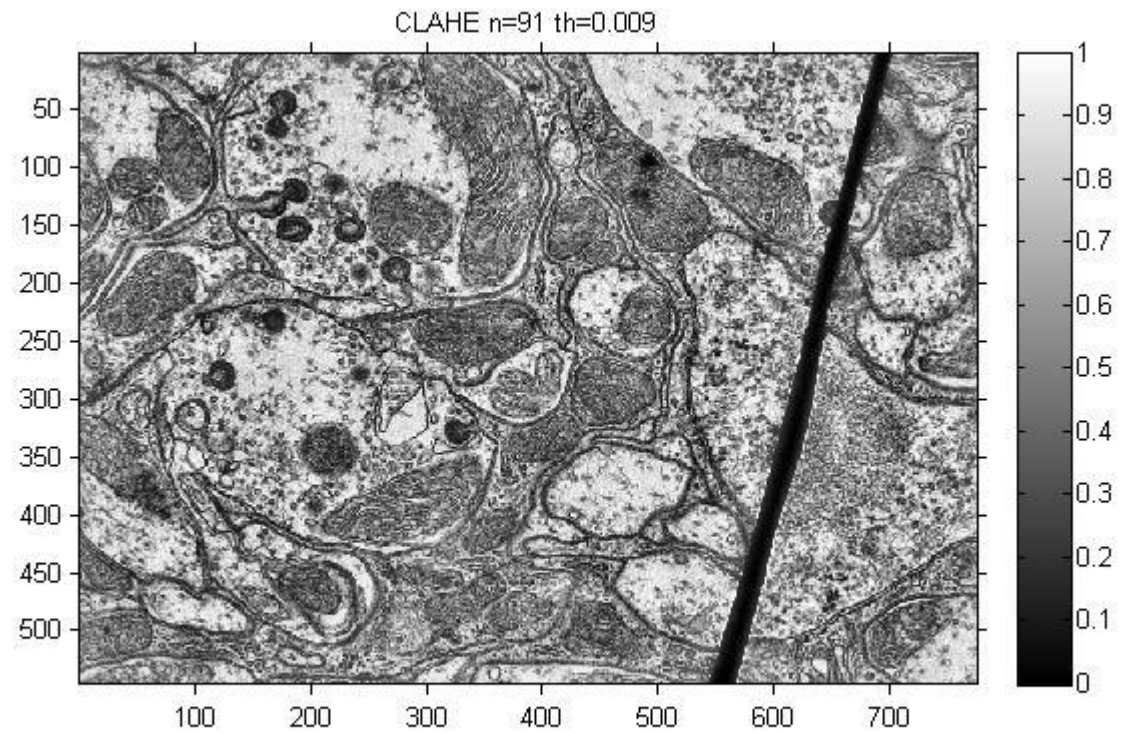




For Threshold parameter = (Threshold ideal)/2




CLAHE n=161 th=0.0175
50
100
150
200
250
300
100
200
300
400
500
600
700
60
50
40
30
20
10


CLAHE n=91 th=0.009
50
100
150
200
250
300
350
400
450
500
100
200
300
400
500
600
700
1
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0