Advanced Image Processing
Assignment #2

Shikhar Vashishth

M.Tech CSA - 13374

Problem 1

Part 1:

Kernel Size	**Standard deviation**	**Mean Sq. Error**
3	0.10	99.934713
3	**1.00**	**89.614492**
3	2.00	110.292918
3	4.00	115.660166
3	8.00	117.003916
7	0.10	99.934713
7	1.00	114.808492
7	2.00	217.848307
7	4.00	266.146444
7	8.00	279.693726
11	0.10	99.934713
11	1.00	114.857747
11	2.00	233.674296
11	4.00	326.976624
11	8.00	361.110718

The best low pass Gaussian filter was of **size 3** with **standard deviation 1.0**.

Observations:

1. From the observed MSE values we can see that as we increase the size of the Gaussian filter and the value of the standard deviation, the blurness in the image increases which in turn increases the MSE error.

2. The mean square error with sigma = .1 remains almost the same for all the kernel sizes because, the kernels generated in all these cases have middle entry as 1 and all other entries are of order $\approx 10^{-40}$ which give us back the same noisy image. That is the reason why Gaussian filters with low variance (0.1) are not optimal. On the other hand, if we increase the size of the filter and its variance too much then the resulting image becomes too blurred and thus, more away from the desired image. Therefore, a filter with moderate size and variance gave us the best results.

Figure 1: Choosing optimal low pass Gaussian filter

Part 2:

Variance of high pass of the original image y_1:

$$\sigma_{y_1}^2 = \frac{1}{MN}\sum_{j=1}^{N}\sum_{i=1}^{M} y_1^2(i,j) = 148.1371$$

Variance of noise in the high pass image z_1:

$$\sigma_{z_1}^2 = (1 - w(0,0))^2\sigma_{z_1}^2 + \sum_{l,k[(l,k)\neq(0,0)]} (w(k,l))^2\sigma_{z_1}^2 = 71.7244$$

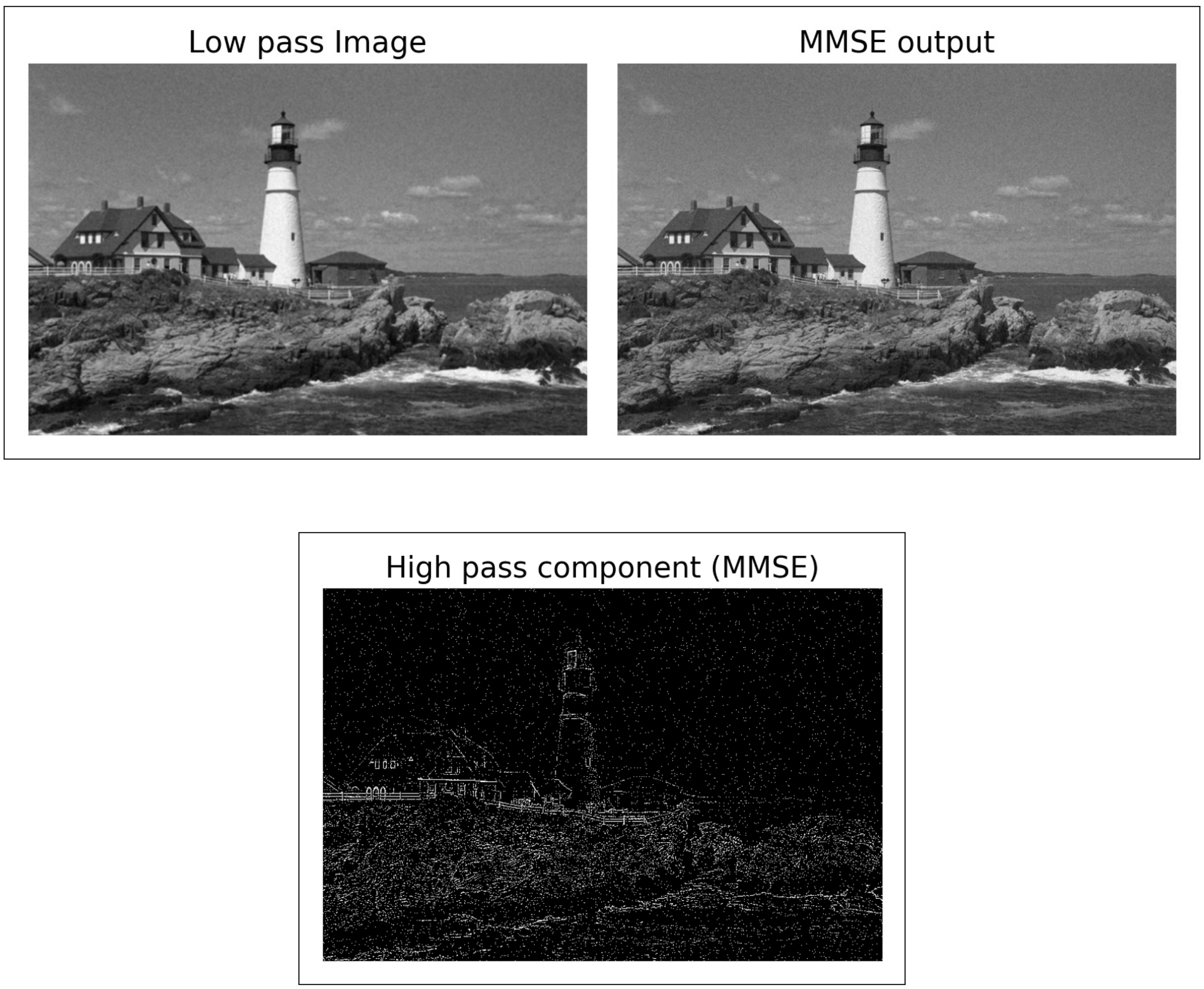

Figure 2: Applying MMSE filter

The value of MSE of low pass component obtained after applying gaussian filter was **89.6144** and after applying MMSE filter ($\hat{x} = \mu_y + \hat{x_1}$)the value of MSE reduced to **57.9407**.

Observations:

1. We can clearly see that the mean squared error has reduced on adding the high frequency component, $\hat{x_1}$ to the low pass component, μ_y. Thus MMSE has made the output more closer to the original image.

2. Although the low pass output of the gaussian filter and MMSE output are visually not easily differentiable but on careful analysis one can see that MMSE output is more sharper than the low pass image.

3. The figure 2 shows the high pass component of the image which was getting thrown away with the noise. It helps us understand why the MMSE method is able to give us better results.

Part 3: SureShrink

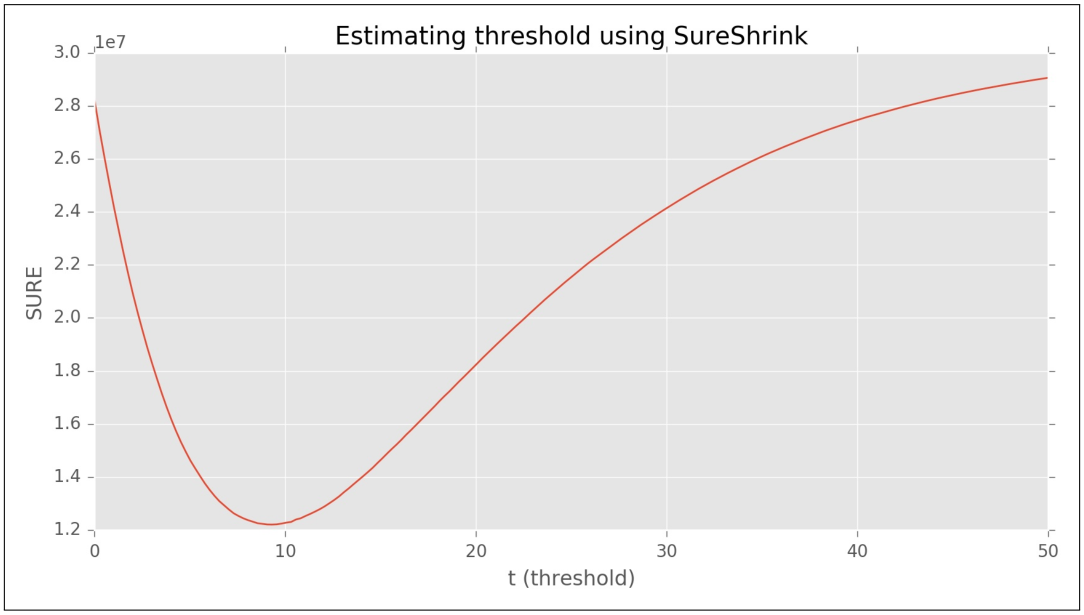

Figure 3: Comparing SURE for different values of 't'

The minimum value of SURE is obtained at threshold(t) = **9.2964**.

Threshold (t)	**MSE**
7.45	51.9138
7.98	50.9634
8.51	50.2244
9.03	49.6786
9.30 (*SureShrink*)	**49.4724**
10.09	49.0986
10.61 (*Best*)	**49.0355**
11.14	49.1028
11.66	49.2875
12.72	49.9651

Table 1: MSE of shrinkage estimator for different thresholds

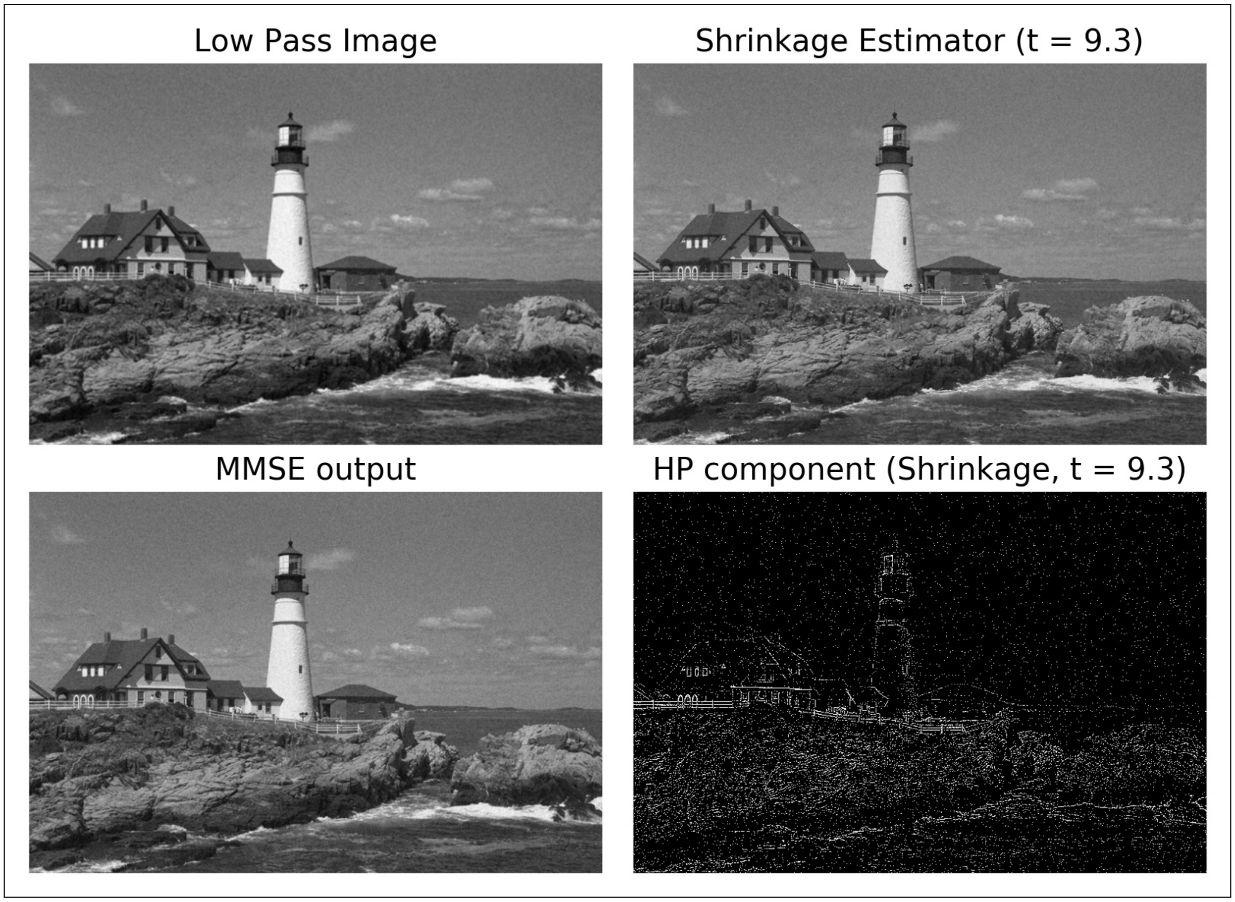

Figure 4: Comparing Shrink Estimator output with low pass output

MSE for Shrinkage estimator: **49.4724**
MSE for MMSE output: **57.9407**
MSE for Low pass Image: **89.6144**

Observations:

1. The SURE value attains minimum at threshold = 9.2964 and start rising and become constant after threshold = 80.

2. The Table 1 verifies the optimality of the threshold given by *SureShrink* method. We can see that although the best value of threshold (with minimum MSE) is around 10.61 but *SureShrink* gives us threshold value as 9.2964 which is quite close to the optimum value.

3. Using the threshold, obtained from SureShrink method, in Shrink Estimator gives us a considerable decrease in MSE. This method gives us better results as compared to MMSE method and gaussian low pass filter.

Part 4: Two scale SureShrink

Selecting Gaussian kernel for second scale

Kernel Size	Standard deviation	Mean Sq. Error
17	**8.00**	**442.986165**
17	12.00	460.042155
17	16.00	466.232666
21	8.00	478.722046
21	12.00	506.104411
21	16.00	516.328451
23	8.00	493.173014
23	12.00	526.761759
23	16.00	539.532837
25	8.00	505.500814
25	12.00	545.830444
25	16.00	561.479736
31	8.00	531.173462
31	12.00	593.426188
31	16.00	619.436320

Table 2: Selecting Gaussian kernel for second scale

The best (lowest MSE) Gaussian filter for second scale was of **size 17** with **standard deviation 8.0**.

Choosing suitable threshold for Shrinkage estimator

Estimating variance of low pass noise ($\sigma^2_{\mu_Z}$)

$$\sigma^2_{\mu_Z} = \sum_k \sum_l w_1(k,l)^2 \sigma^2_Z = 12.5604$$

Estimating variance of noise of high pass in low pass of the image ($\sigma^2_{Z_2}$)

$$\sigma^2_{Z_2} = (1 - w_2(0,0))^2 \sigma^2_{\mu_Z} + \sum_{l,k[(l,k)\neq(0,0)]} (w_2(k,l))^2 \sigma^2_{\mu_Z} = 12.4829$$

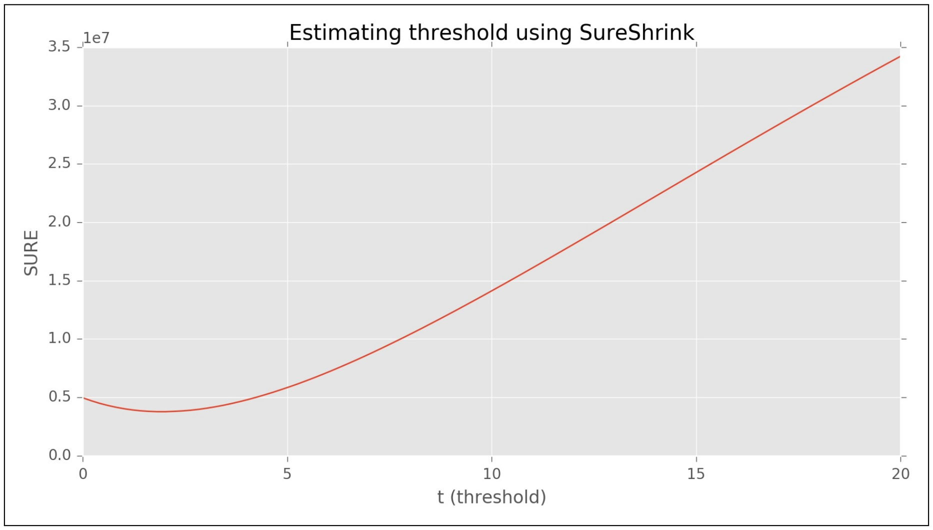

Figure 5: Comparing SURE for different values of 't'

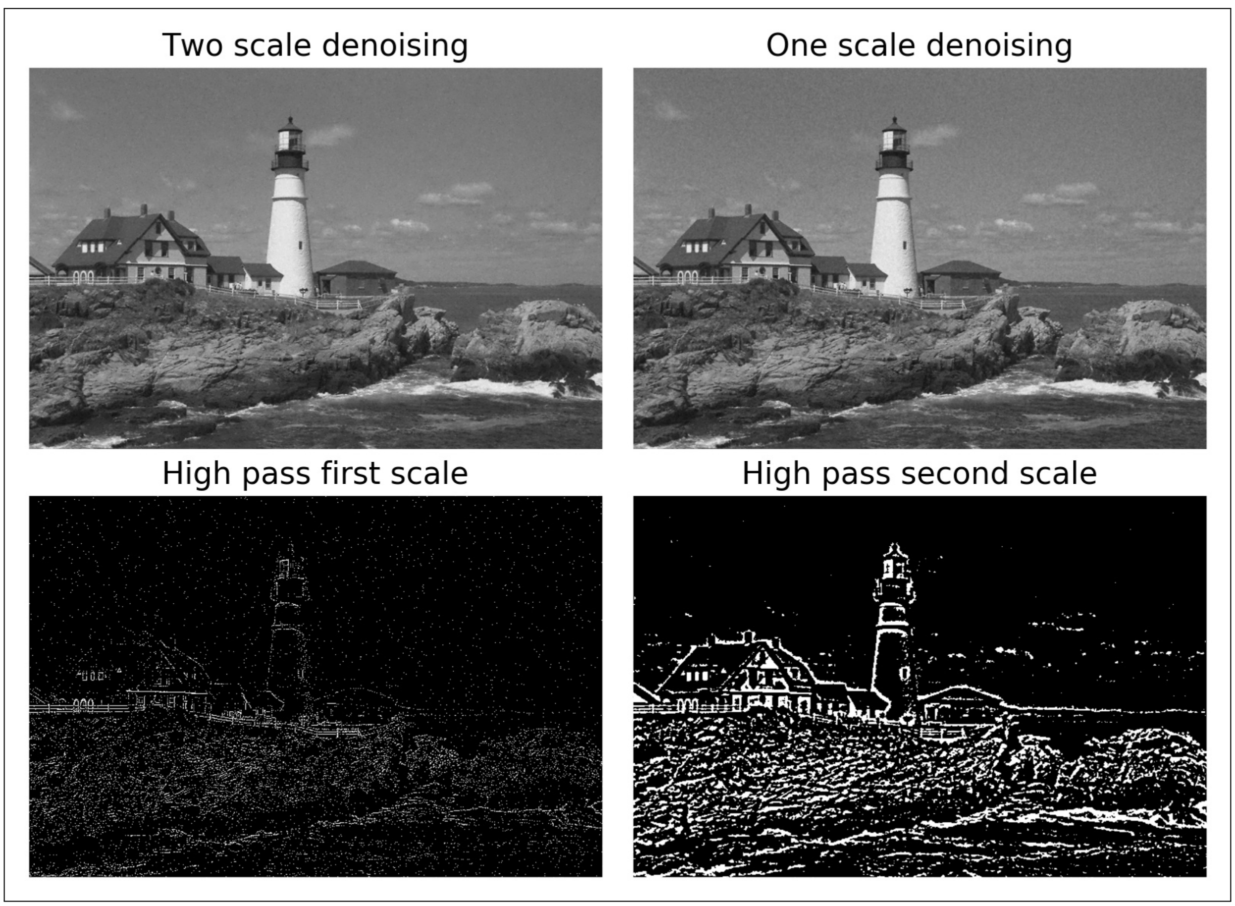

Figure 6: Comparing SURE for different values of 't'

Comparing MSE:
Two scale based on Shrinkage Estimator: **49.1538**
One scale based on Shrinkage Estimator: **49.4724**
MMSE method: **57.9407**
Gaussian low pass filter: **89.6144**

Observations:

1. From the above results, we can easily see that the two scale denoising based on *SureShrink* has further improved the quality of results. The MSE of the final output has further reduced which shows that it has become more closer to the original desired image. Moreover, the bottom right image in the above figure shows the high pass information in the low pass of the image which was not getting utilized by the other approaches discussed before.

2. For the second phase a different low pass filter was used because reusing the same filter from the first phase will not lead to any change in the low pass image. The following results proves this observation. On applying the same filter the MSE of new low pass image obtained got changed by only **11.1581**. But if we apply Gaussian filter of larger size and sigma value then the change in MSE is significant. Change in MSE with Gaussian of size 17 and sigma 4 was **200.2483** and with size 31 and sigma 8 was **358.5539.**

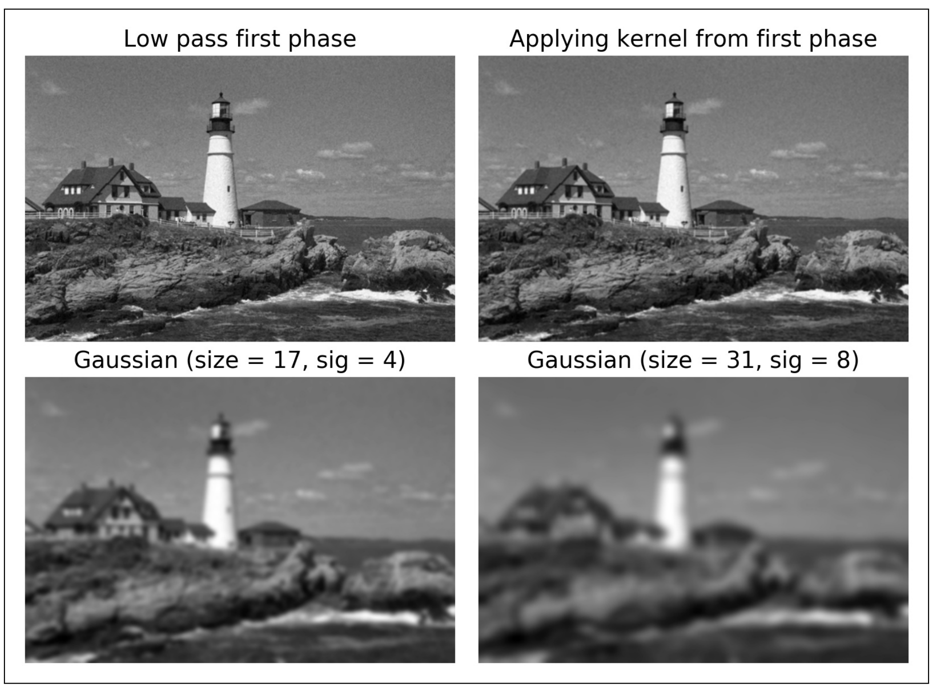

Figure 7: Consequences of using same filter in second pass

Problem 2

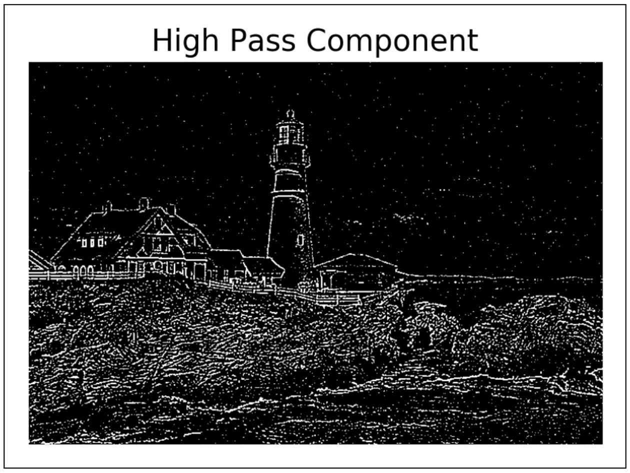

Figure 8: High pass component obtained using given filter

Part 1: Constant Gain

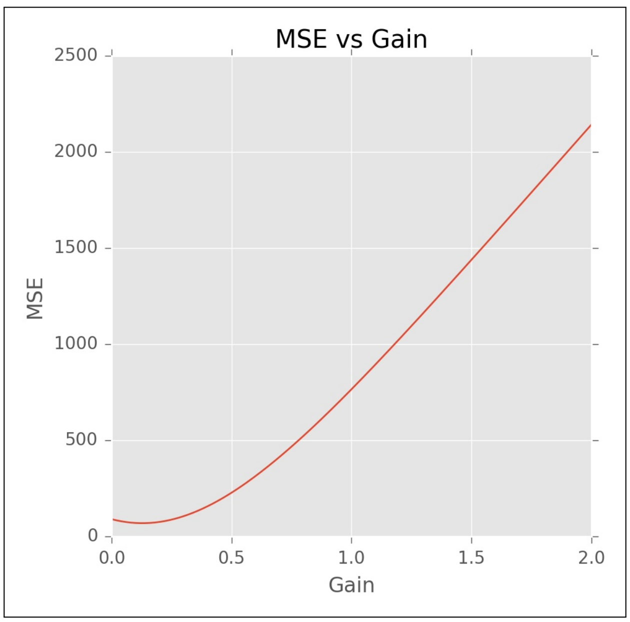

Figure 9: MSE vs gain

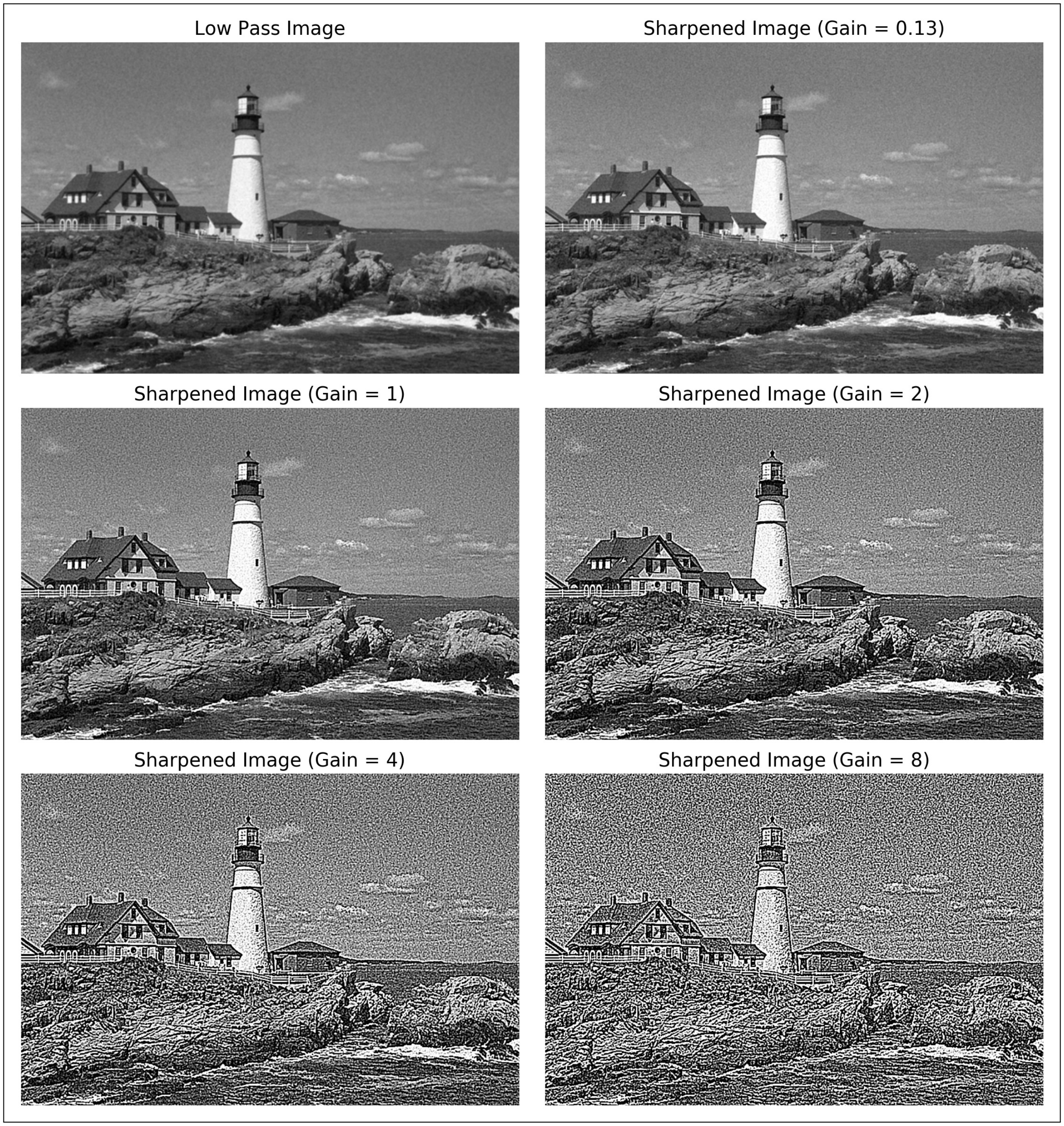

Figure 10: Comparing low pass and sharpened image

Observations:

We can clearly see from the graph between MSE and gain that as the gain increases the MSE also increases. This is because

$$\hat{x} = \mu_y + \lambda y_1, \text{ where } y_1 \text{ is high pass component of the image}$$

Since, the high component of the image also contains noise other than the edge information therefore, as we amplify high pass component the noise in the image also gets amplified which we can clearly see from the bottom right image in the above figure. Moreover, because the Laplacian filter was used for getting the high pass component, along with the edge information the noise also got captured. We can see from the graph that the MSE error becomes linear function of gain for its large values.

Part 2: Variable Gain

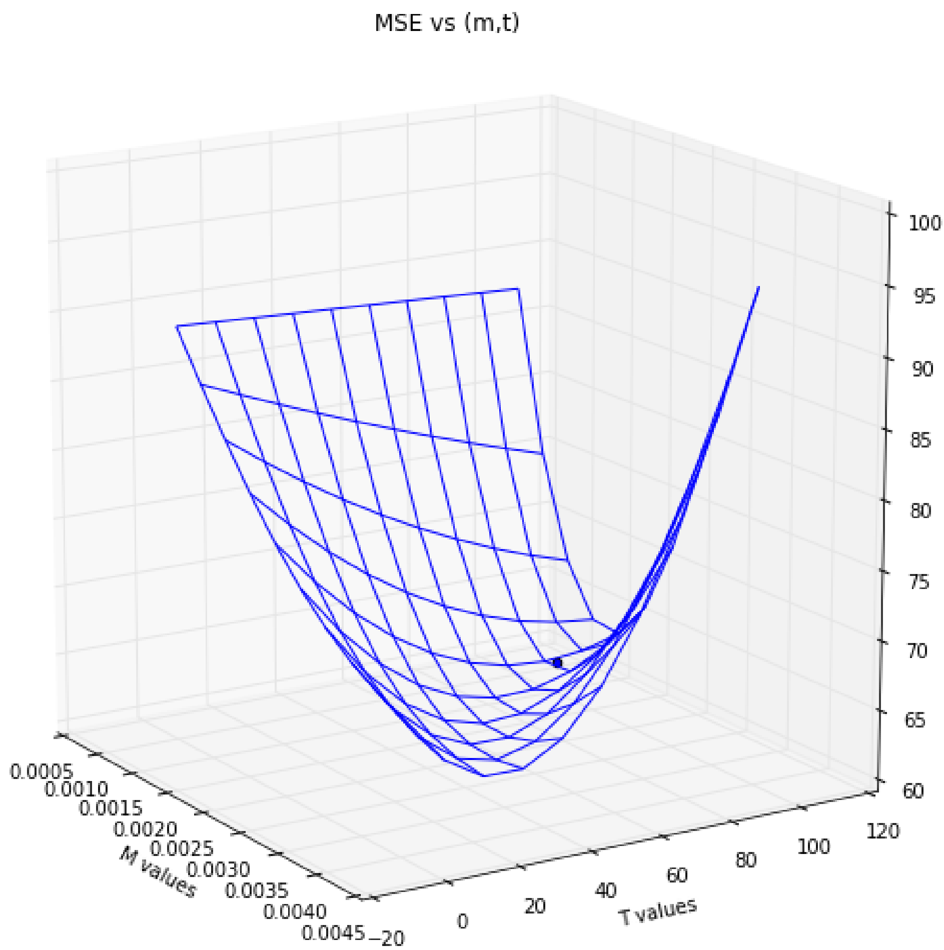

Figure 11: Getting optimal values of m and t

The lowest MSE was obtained at **m = 0.002** and **t = 89.0**.

Comparing MSE:
Low Pass Filter: **89.6383**
Constant Gain: **68.6534**
Variable Gain: **63.3468**

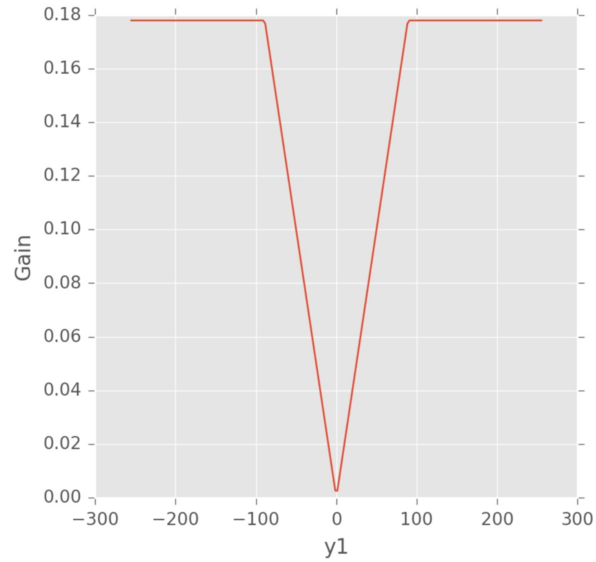

Figure 12: Gain variation vs y1 for chosen (m,t)

Figure 13: Visual output with variable gain

Observations

1. As compared to constant gain, the variable gain gave us better results in terms of MSE and visually as well.

2. The optimum value of constant gain gave us the range to search for the optimum values of parameters m and t. As the search range of m and t has to be such that $m \times t$ should be near to the optimal constant gain value for atleast one combination. In the above setup I chose a large value $(0, 100)$ for t because the gain is defined as:

$$\lambda(y_1) = \begin{cases} m|y_1| & |y_1| \leq t \\ mt & o.w. \end{cases}$$

So, for any small value of t will make the variable gain will be almost constant as it will affect only a small range of y_1. And for keeping the product of m and t near to the optimal of constant gain I chose m range to be small $(.001, .004)$

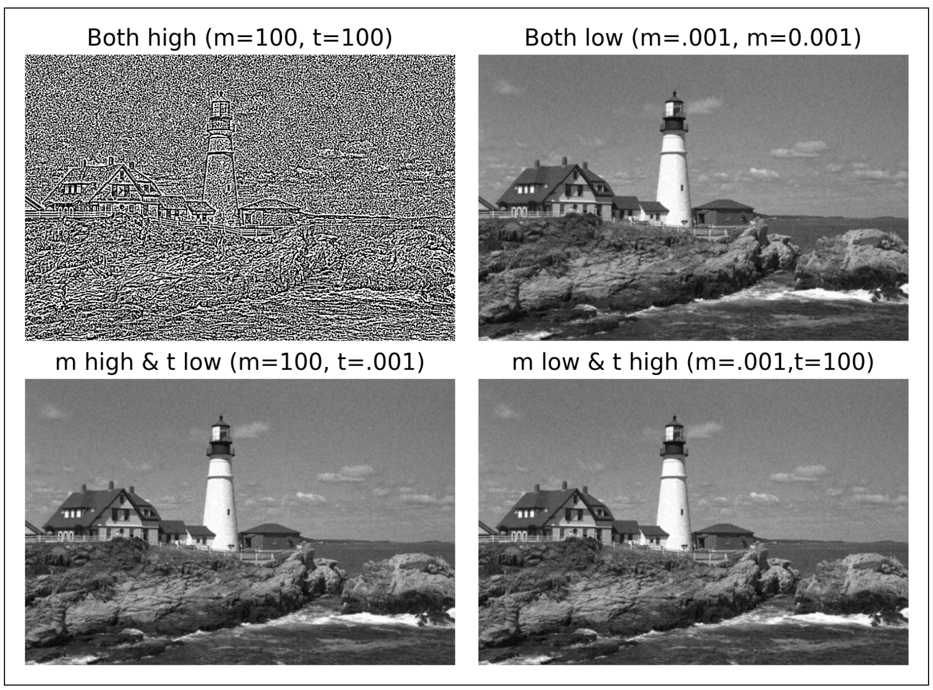

Figure 14: Visual comparison for different (m,t) combinations

MSEs: 15098.0013, 89.6205, 69.6400, 68.8058 respectively

We can see from the above results that if we take large values for both m and t then we get completely spoiled result. On using small values for both m and t we get no improvement in the MSE from low pass image. But one using values of m and t such that their product is approximately equal to the optimal value obtained in case of constant gain we experience much better results which proves the correctness of our observation.