

Computer Vision

Report Assignment-1

Kaustav Vats (2016048)

Note - All the result are performed on grayscale images with zero padding.

Question 1

On increasing kernel size, the resulting image has less high frequency and less noise. On increasing kernel size image gets smoother. Edge details are not preserved.

Average filtering

Results of Average Filter on original image

Original Image



Average Filter of 3x3



Average Filter of 5x5



Average Filter of 11x11



Average Filter of 15x15



Question 2

On increasing kernel size, image quality is degraded. It seems like median filter is not good for images with high noise. On increasing kernel size image gets smoother. But median filter is better than Average filter on preserving edge details.

Salt and Pepper Noise

Original Image



10% Salt and Pepper Noise

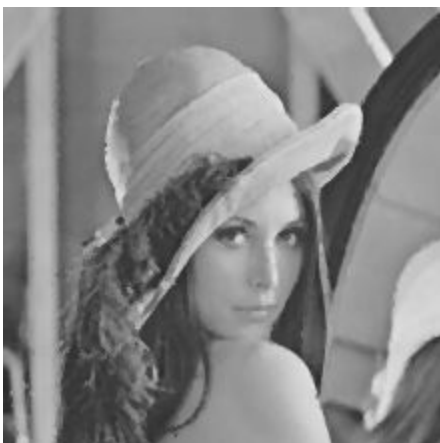


20% Salt and Pepper Noise

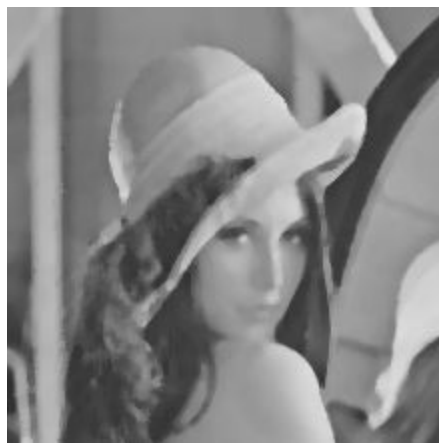


Results of Median Filtering on 10% Salt and Pepper Noise Image

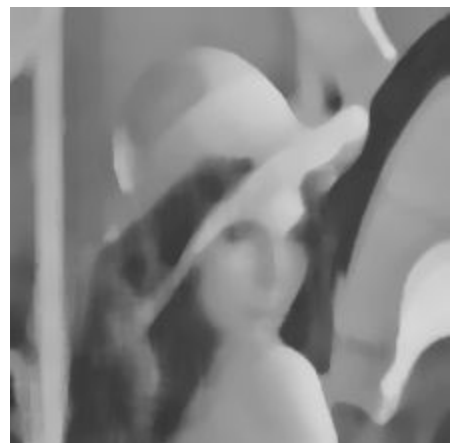
Kernel Size 3x3



Kernel Size 5x5



Kernel Size 11x11

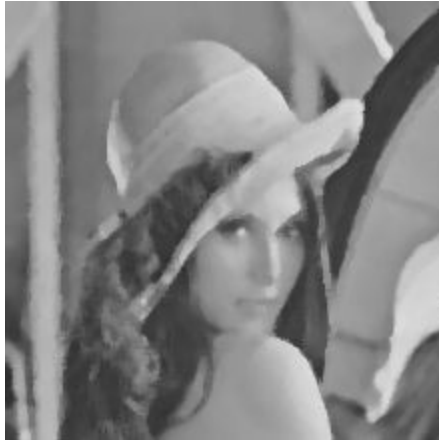


Results of Median Filtering on 20% Salt and Pepper Noise Image

Kernel Size 3x3



Kernel Size 5x5



Kernel Size 11x11



Question 3

Gaussian Filtering

Pixel gets more spread out in gaussian on increasing kernel size. Image get much smoother. On increasing kernel size of pixels also increase. Thus creating more blurriness in the image. It also reduces high frequency details significantly on increasing kernel size.

Results of Gaussian Filter on original image

Original Image



Gaussian Filter Kernel 3x3, $\sigma = 1$



Gaussian Filter Kernel 5x5, $\sigma = 3$



Gaussian Filter Kernel 11x11, $\sigma = 4$



Gaussian Filter Kernel 15x15, $\sigma = 7$



Results for varying the σ parameter

Gaussian Filter kernel 15x15

$\sigma = 1$

$\sigma = 3$



$\sigma = 6$



$\sigma = 11$



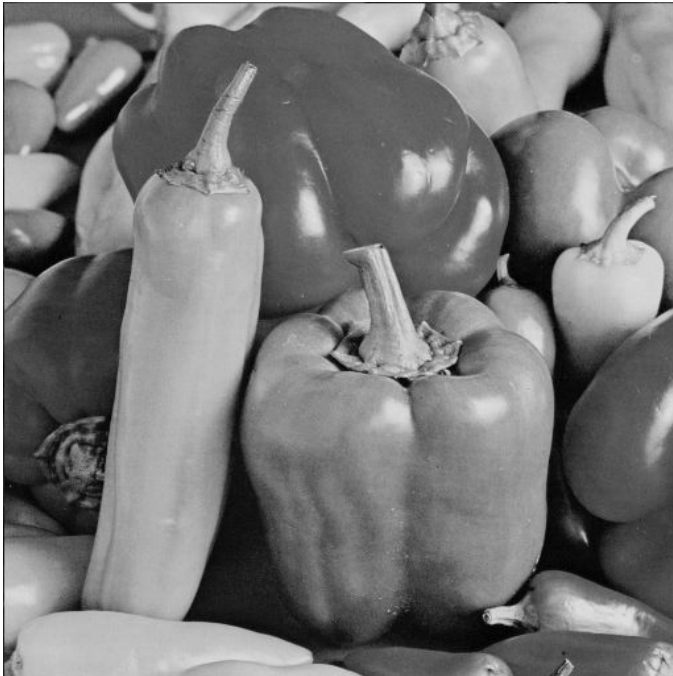
Question 4

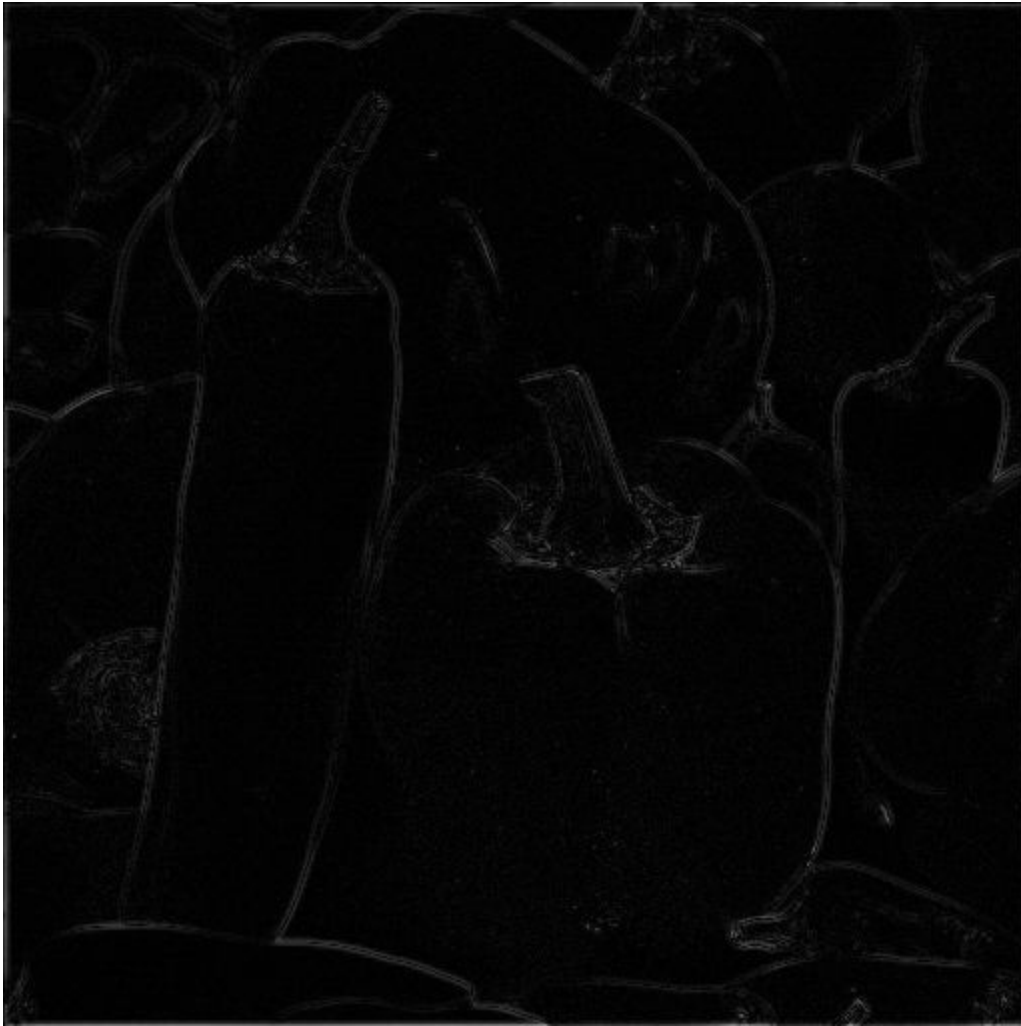
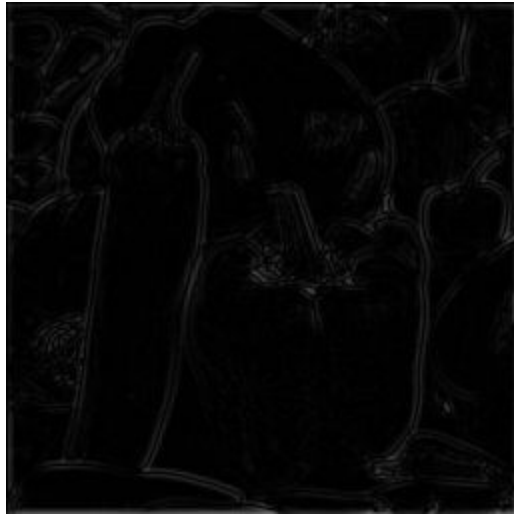
3 Level Gaussian and Laplacian Pyramid

Kernel Size = 5

$\sigma = 5$







Above images with absolute difference

Question 5

Average Filter

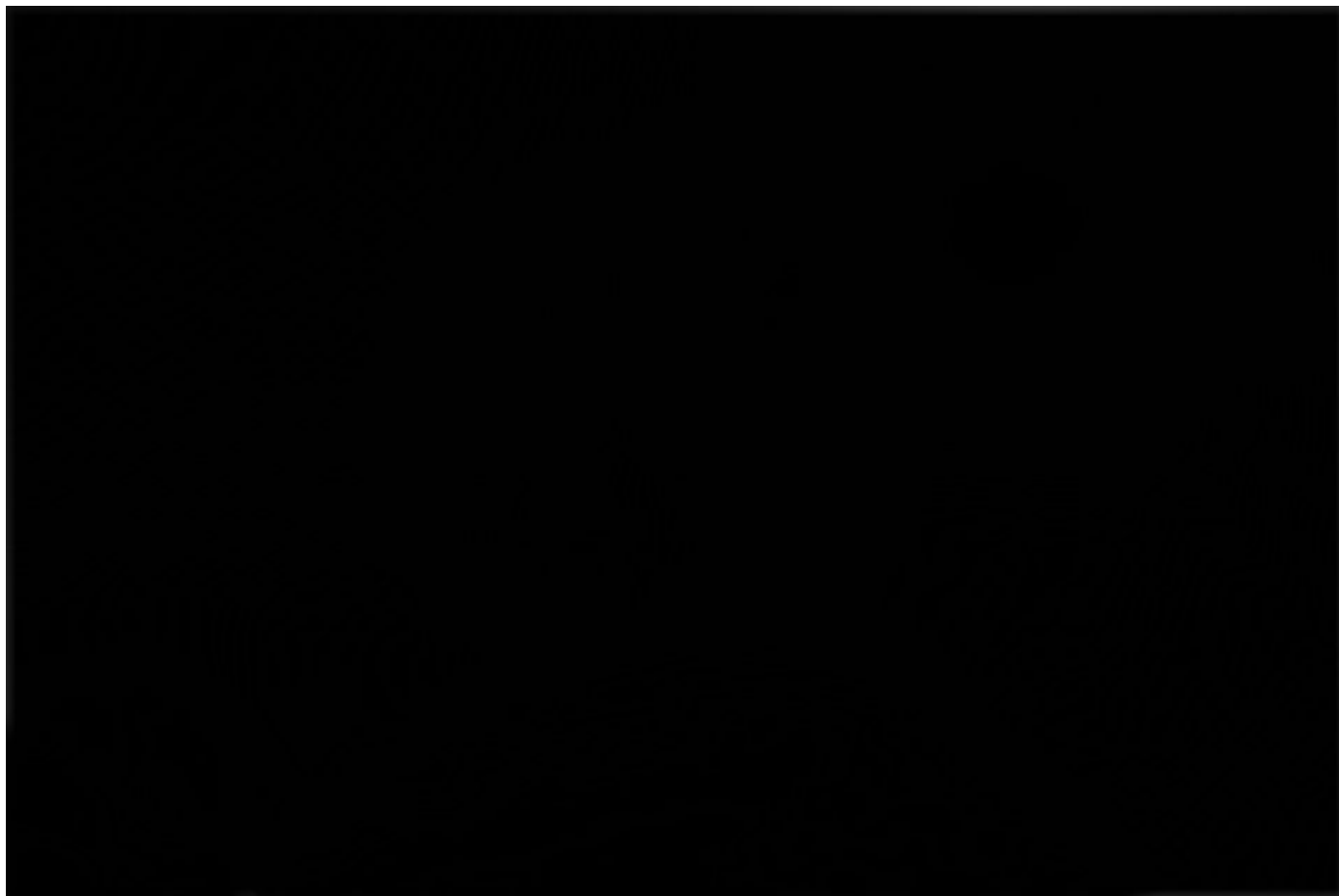
Output for self-implemented Average Filter(A)



Output for in-built Average Filter(B)



Difference



Difference is only visible on the boundary.

Median Filter

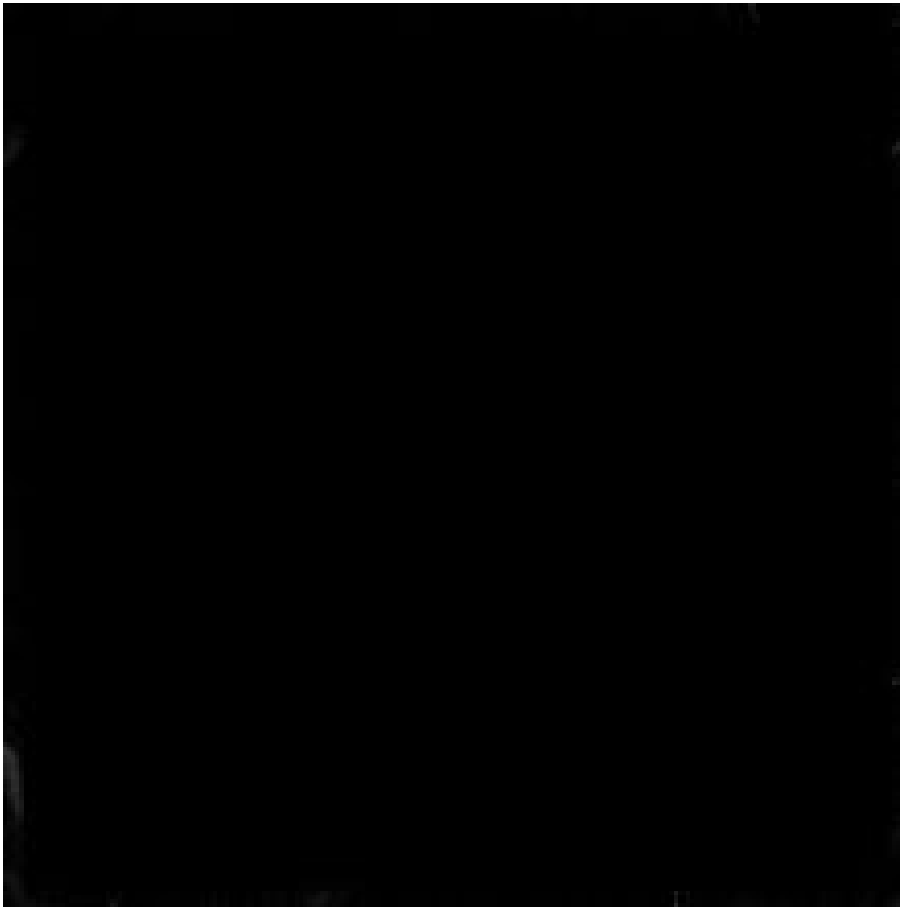
10% Salt and Pepper Noise

Noisy Image

Self-implemented Median Filter(A) Output for in-built Average Filter(B)



Difference



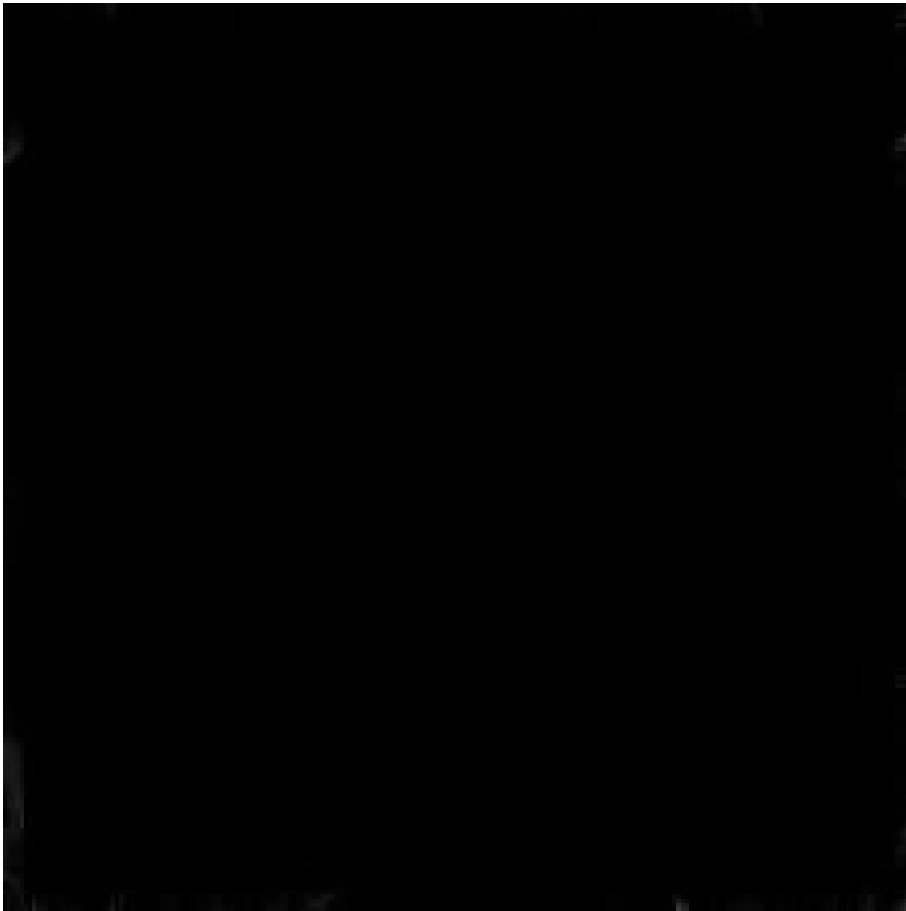
20% Salt and Pepper Noise

Noisy Image

Self-implemented Median Filter(A) Output for in-built Average Filter(B)



Difference



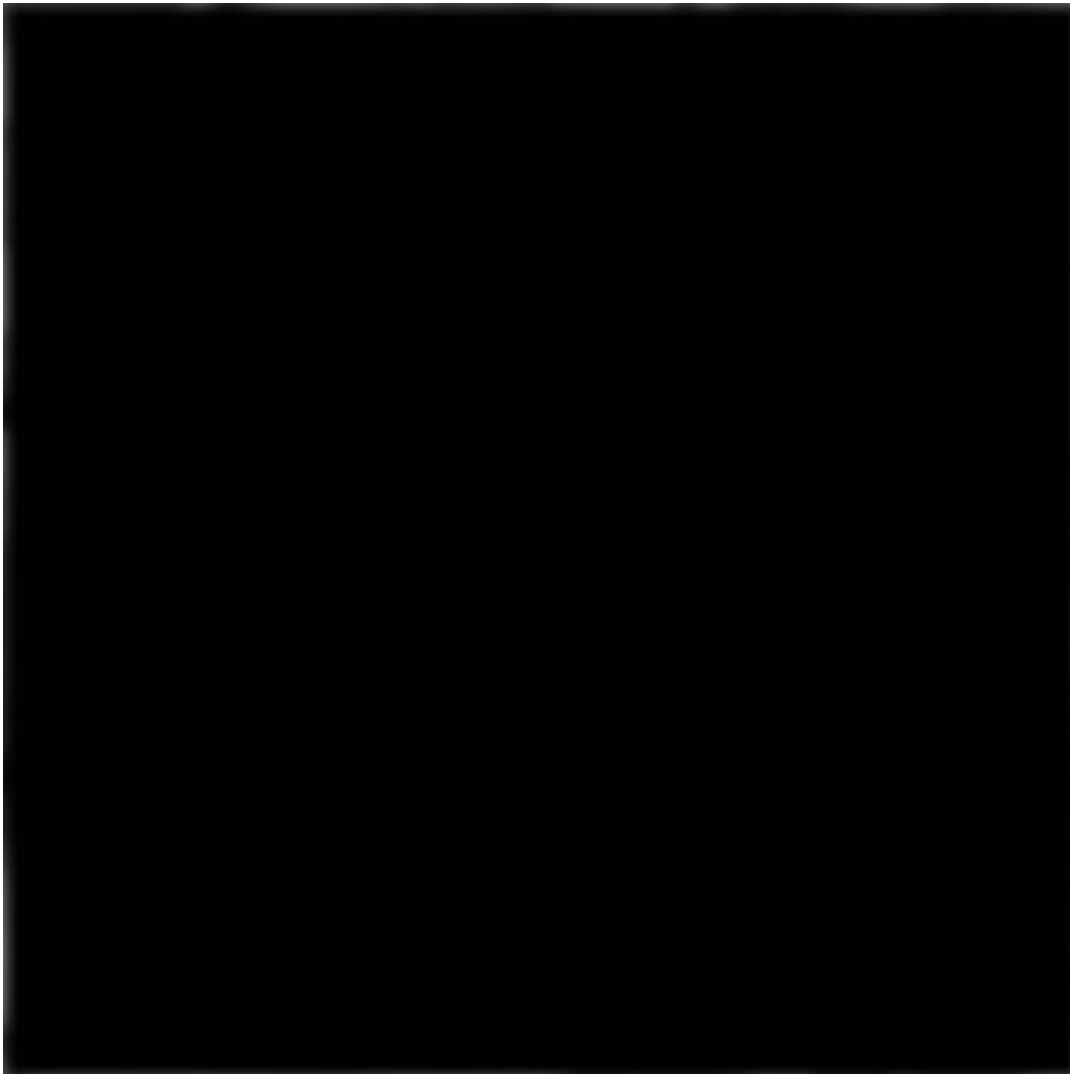
Gaussian Blur

Self-implemented Gaussian Filter(A)

Output for in-built Average Filter(B)



Difference



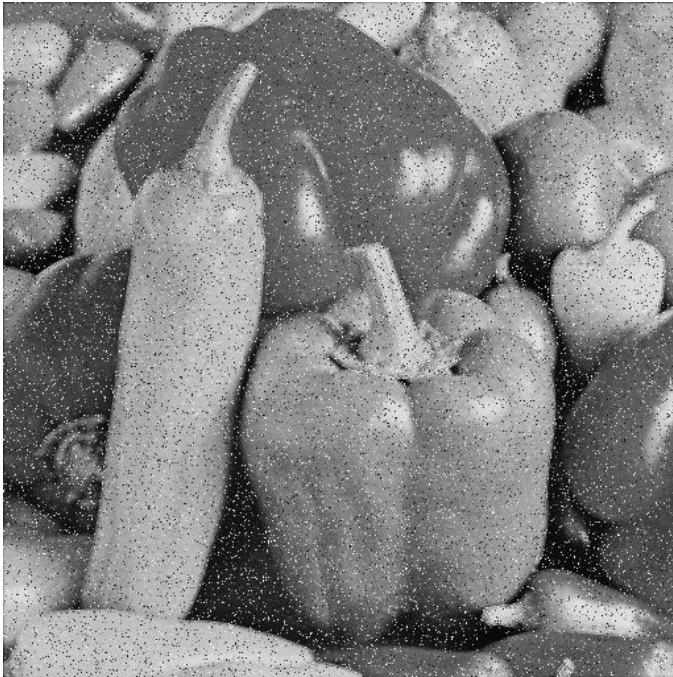
Difference visible on the boundary of the image. Maybe because of uint8 overflow or because of zero padding.

Question 6

Reduced LH, HL, HH band to value 0. In the result image, clearly see that the image is smooth and some of the high frequency noise is removed. (edges details are completely removed)

Haar Filter

Noisy Image (l')



Smooth Image



Question 7

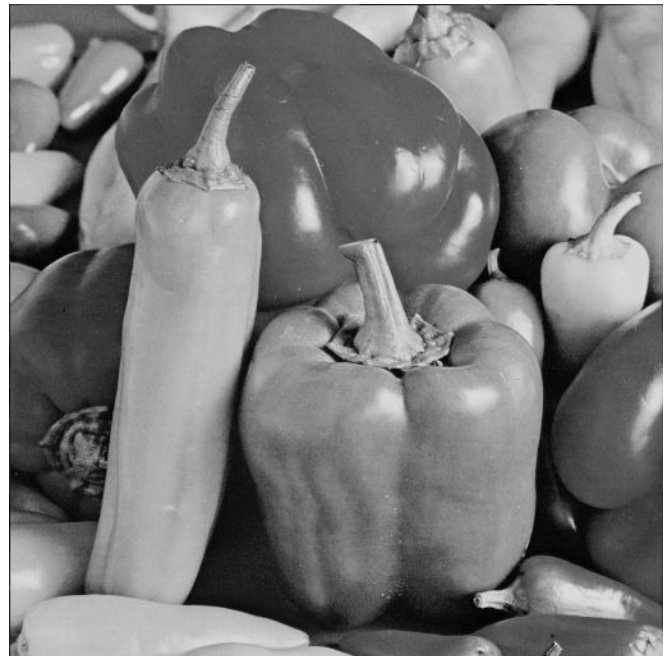
Performed 3 Level DWT, then added LL band of watermark image with product with some constant value 'a' to LL band of original image. Then combined new LL band with rest of the bands of original image to apply watermark then did IDWT to get the watermarked image.

Watermarking using DWT

Watermark Image



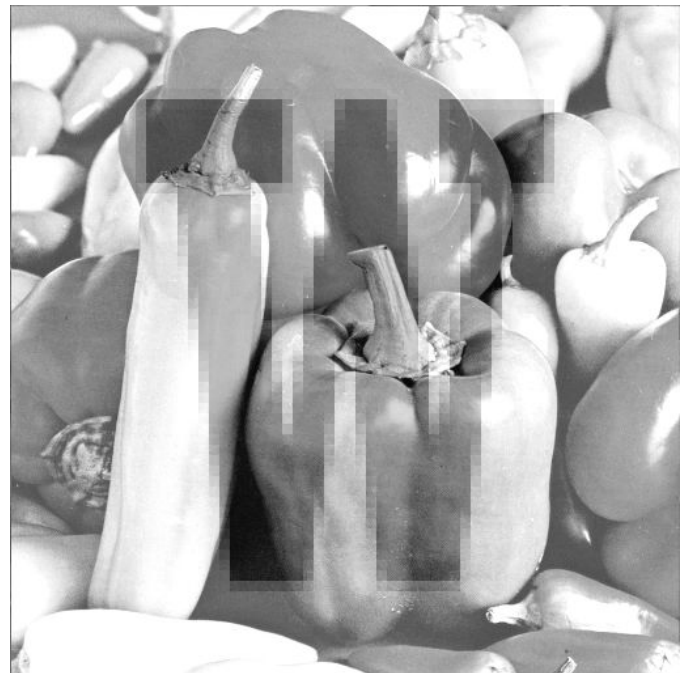
Original Image



Watermarking on Original Image



$A = 0.15$ (15%)



$A = 0.3$



A = 0.08 (8%)