Computer Vision LAB-02
Robotics Engineering, Unversity of Genova

VATS, KOHLI

27 October 2021

0.1 Abstract

The Lab-02 focused on the methods to add different types of noise, filters and applying Fourier transformation to the sample image. Using MATLAB appropriate approach is applied on the sample image provided during the lab and the changes are observed with the image.

0.2 Objective

1. Add Gaussian (standard deviation=20) and salt and pepper (density=20) noise to the provided images:
• display the images, the noisy images, and their histograms.
2. Remove the noise by using a moving average, a low-pass Gaussian filter and a median filter:
• use two different spatial supports: 3x3 pixels and 7x7 pixels.
• display the filters by using imagesc() and surf()
• display the resulting images and their histograms.
3. Implement the slides 41-45 "practice with linear filters":
• use filters with a spatial support of 7x7 pixels.
• display the filters by using imagesc() and surf()
• display the resulting images.
4. Apply the Fourier Transform (FFT) on the provided images:
• display the magnitude (log) of the transformed images.
• display the magnitude of the transformed low-pass Gaussian filter (spatial support of 101x101 pixels with sigma=5).
• display the magnitude of the transformed sharpening filter, slide 44 (the filter has a spatial support of 7x7 pixels, copy it in the middle of a zeros image of 101x101 pixels).

0.3 Methodology

1. Using imread() function first load the image. Use the tree image (fig.1) to proceed with the above objectives. To add Gaussian noise, a random matrix is created of the size of image and multiplied it with the given standard deviation (20). By using imagesc() and imhist()

function the desired result of image with added gaussian noise is obtained. (fig.2) and (fig.3) in the result section will show the same. The more the value of standard deviation the more noise we get.
To add salt and pepper noise create a random sparse matrix with value lying between [0,1] using sprand() where density is also mentioned. Take all the values that lie below 0.5 as L1 which are first negated and then multiplied to image matrix which creates the pepper noise or black dots and then values above 0.5 taken as L2 are multiplied to the result of input matrix and L1 to give a result image and histogram in (fig.4) and (fig.5) respectively of added salt and pepper noise.

2. For applying moving average with spatial support of 3x3 matrix first created a 3x3 ones matrix then each of its elements were divided by 9 which is known as kernel matrix. Convolute this matrix with noisy image to get moving average with spatial support of 3x3. Similarly, make 7x7 filter and then it is also convoluted. Resulting image can be seen in (fig.6) and (fig.7) respectively for 3x3 and 7x7. Also, histograms can be seen in (fig.8 and 9) respectively.
Create a two dimensional gaussian filter using fspecial('type') to get gaussian low pass filter and the imfilter() was used to implement the filter in the image. (Fig.10) shows the result and (fig.11) surface of the resulting image.
Median filter is used to remove the salt and pepper noise the command used is medfilt2('image', size of matrix). Two median filters a (3x3) and a (7x7) filter are created and the bigger the size of filter the better it reduces the salt and pepper noise is observed. (Fig.12, fig.13,fig.14 and fig.15) shows the result of the above.

3. According to slides 41-45 slide 41 is a 3x3 zero matrix with centre value as one and this matrix is convoluted with the noisy image to get a resulting sharp image. Create a 7x7 zero matrix using function zero(size of matrix) and then make the centre of matrix 1 to make similar filter as in slide 41 and then convolute this matrix with noisy image matrix to get result in the (fig.16) and then by using surf() command create a 3D surface of the output displayed in (fig.17). This linear filter clears the image and reduces its overall size.
If 1 in centre of matrix is shifted to any other value for instance, from

(4,4) to (1,1) this will result in the image to shift left and up. as results show in fig.(18) and fig.(19).
If the ones matrix is divided by an integer the filter made is known as box filter which creates a blurring effect on the image as shown in (fig.20) and (fig.21)
A sharpening filter can be produced by subtracting the box filter from twice of the linear filter. this filter creates a detailed and sharpened image when convoluted with an image matrix. fig.(22) and fig.(23) show the same.

4. (Fig.24,25,26,27) are the results obtained of Fourier transformed images. The command fftshift(fft2('image') is given for the same.
Create a random 101x101 matrix and does the Fourier transformation and Fourier shifting of the same. OUTGF shows the gaussian low pass filter which is then fft of this filter is done. Image results are shown in figure(28, 29).
The linear filter 7x7 created earlier is first copied in the middle of a 101x101 zero matrix using for loop. once the filter is copied we did a fft of this filter (fig.30) and then absolute values of this obtained filter is convoluted with original image to obtain results in (fig.31).

0.4 Results

The resulting images, histograms and 3D surfaces are shown in this section. A total of 30 results will be shown in this section.

Figure 1: sample image

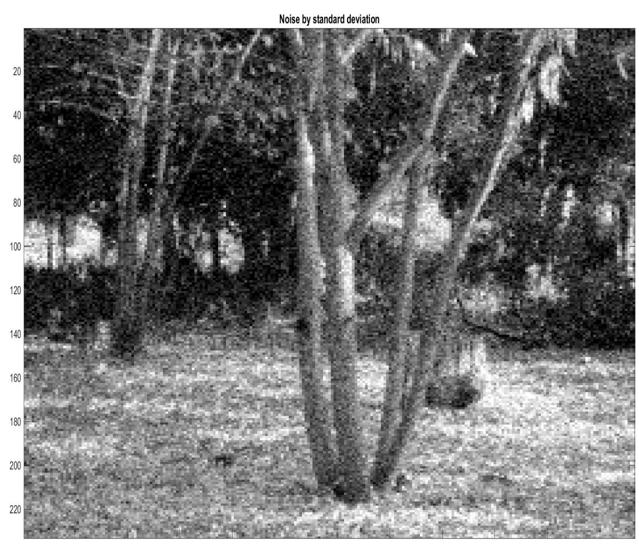

Figure 2: noise standard deviation on sample image

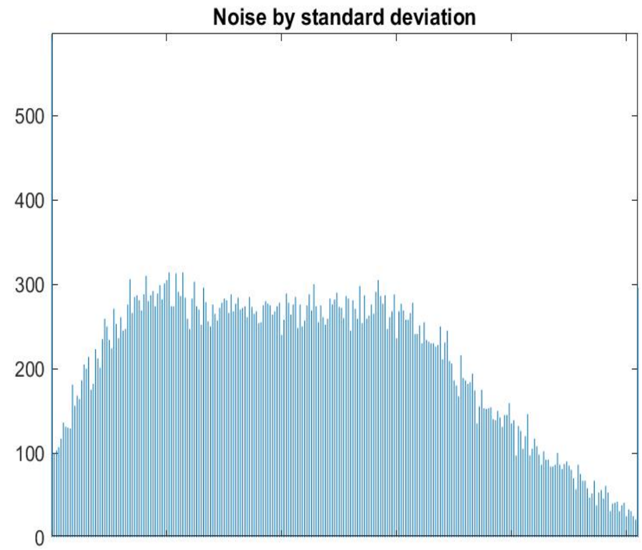

Figure 3: standard deviation histogram

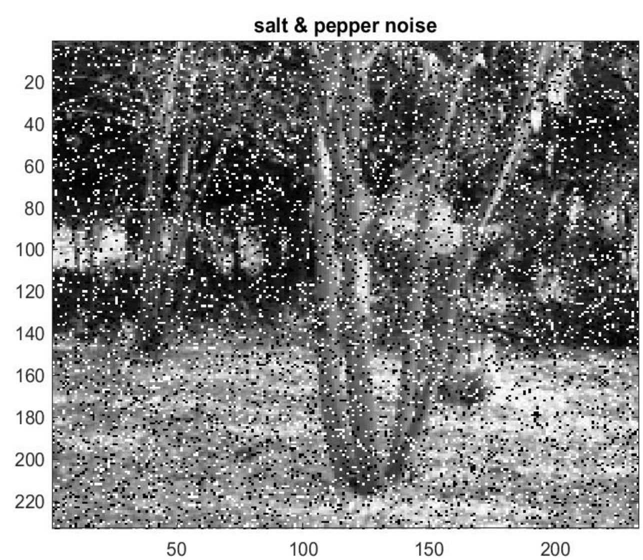

Figure 4: noise salt and pepper on sample image

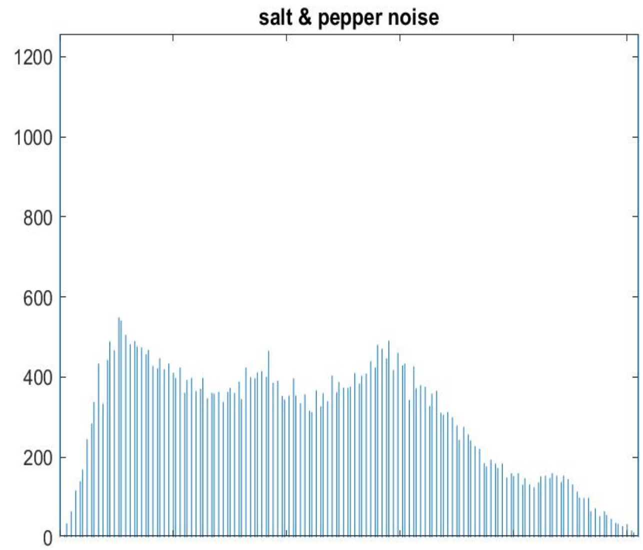

Figure 5: histogram of salt and pepper

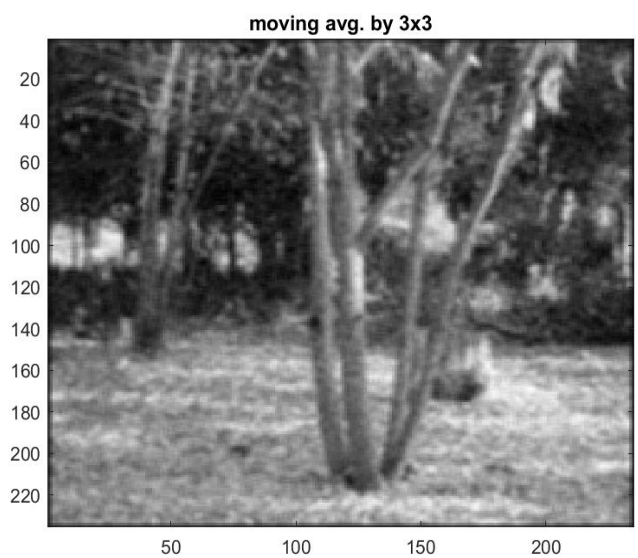

Figure 6: moving average 3x3 filter on noisy image

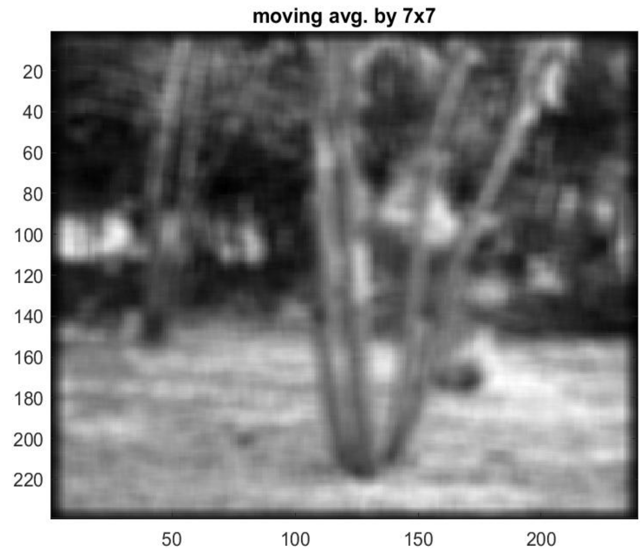

Figure 7: moving average 7x7 filter on noisy image

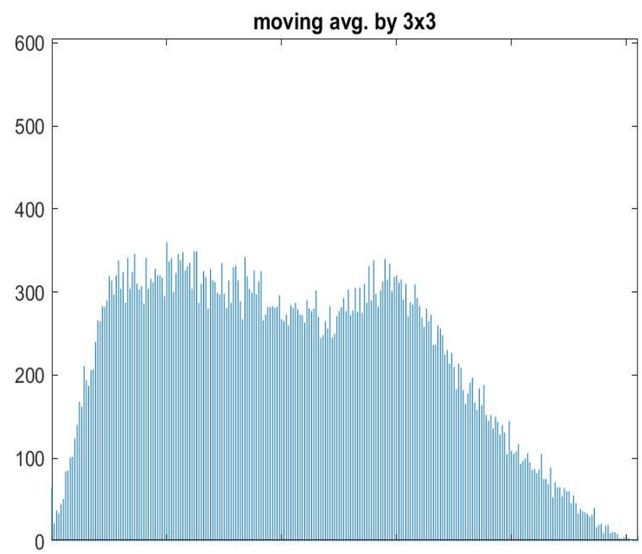

Figure 8: histogram of moving average 3x3 filter

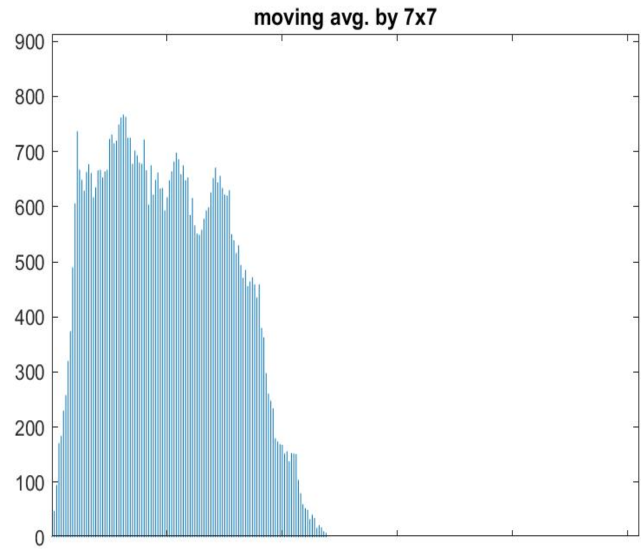

Figure 9: histogram of moving average 7x7 filter

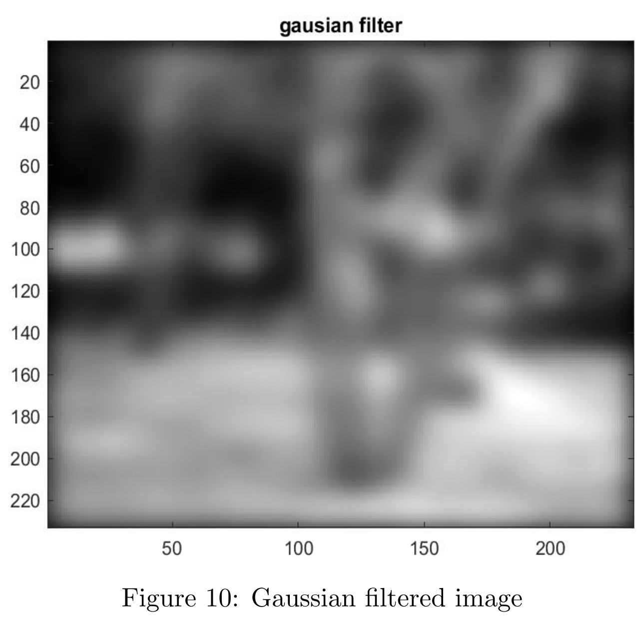

Figure 10: Gaussian filtered image

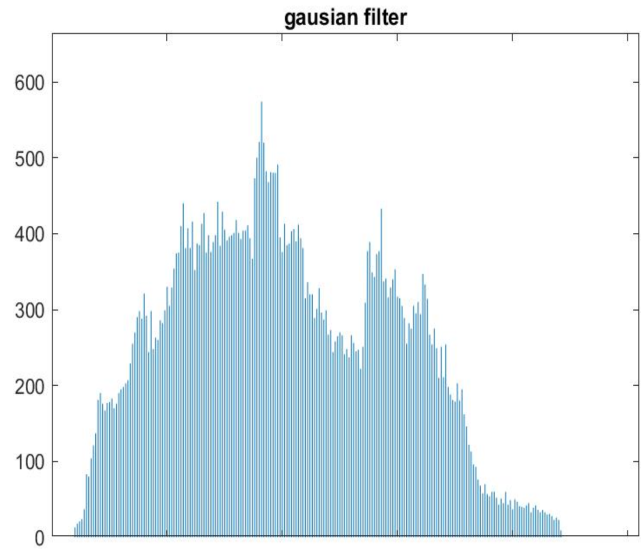

Figure 11: histogram of Gaussian filtered image

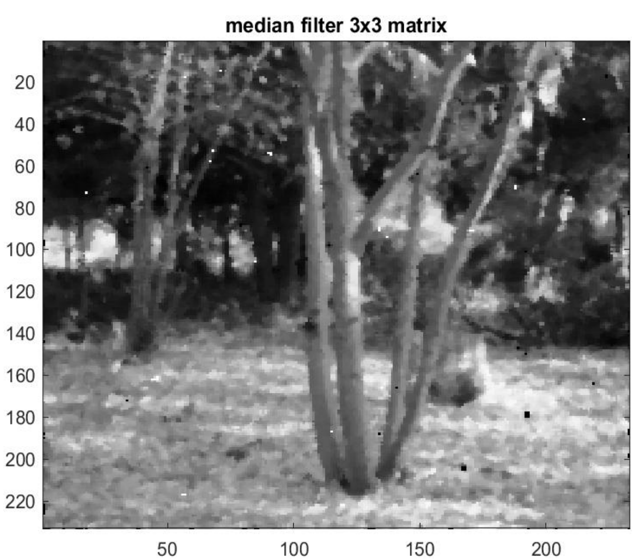

Figure 12: median 3x3 filter on salt and pepper noise image

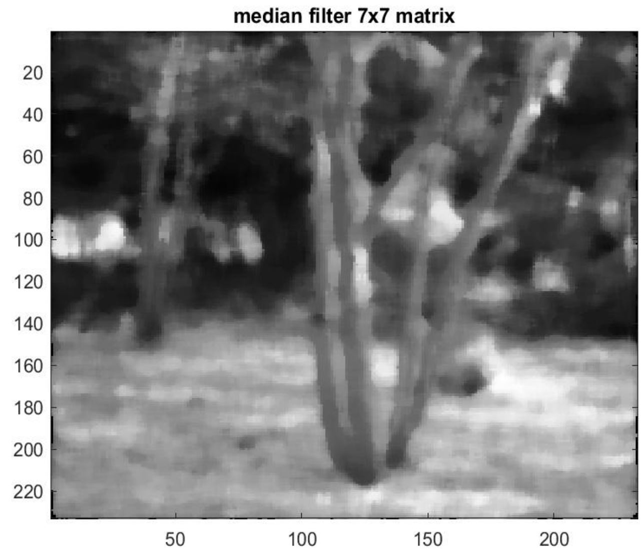

Figure 13: median 7x7 filter on salt and pepper noise image

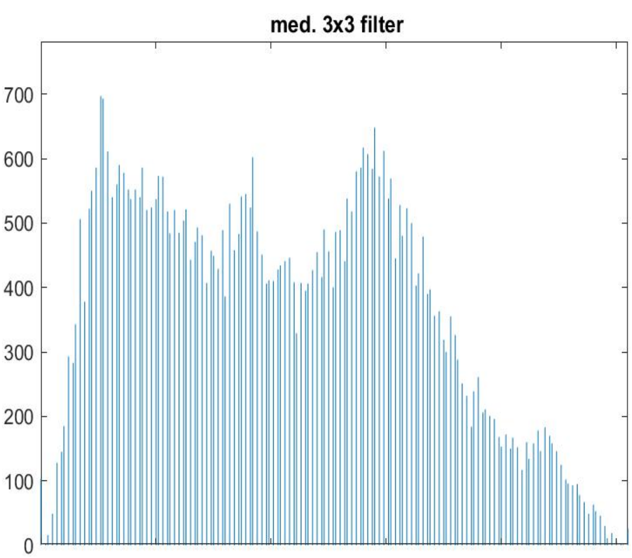

Figure 14: Histogram of 3x3 median filtered image

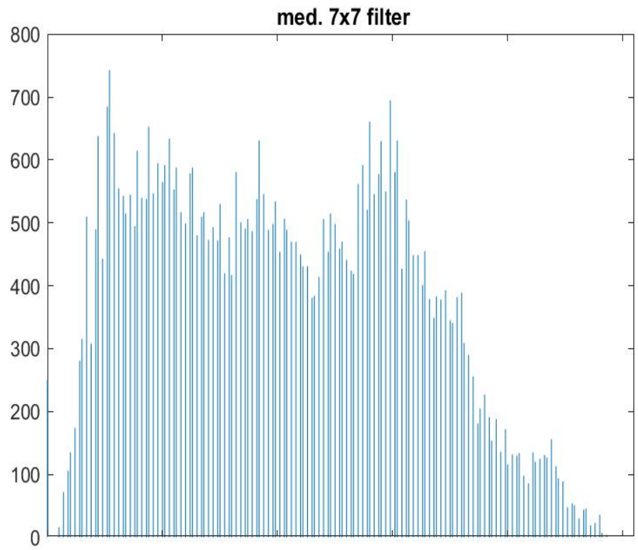

Figure 15: Histogram of 7x7 median filtered image

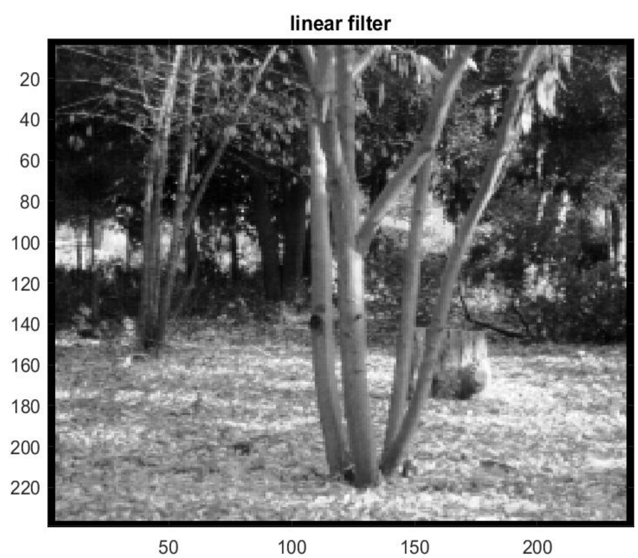

Figure 16: linear filter applied on sample image

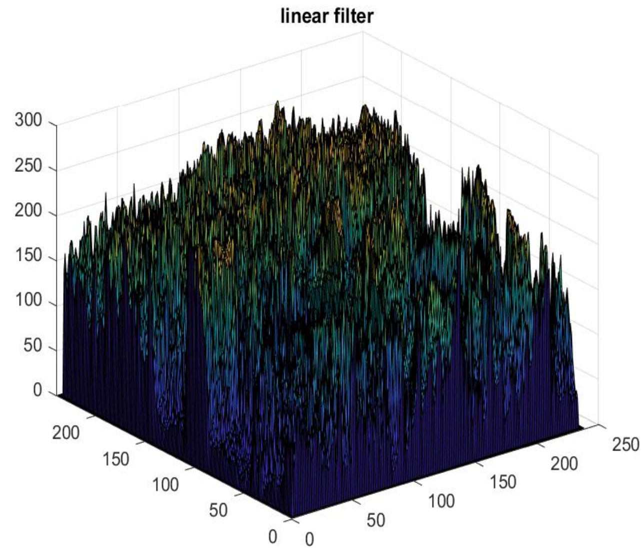

Figure 17: 3-D surface of the linear filtered image

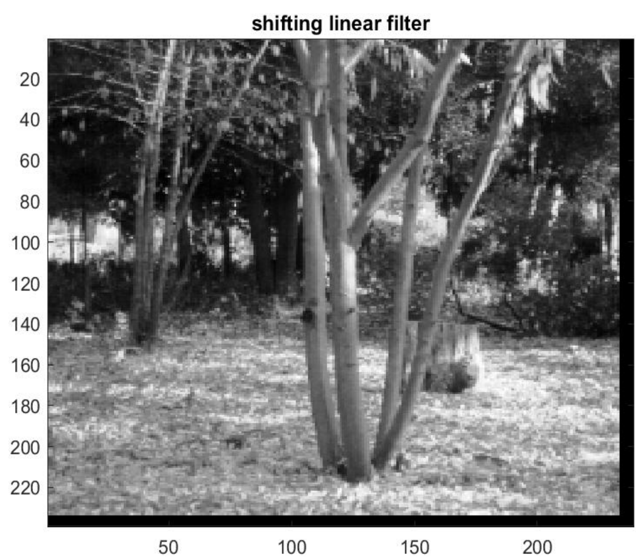

Figure 18: linear filter with shifting of image

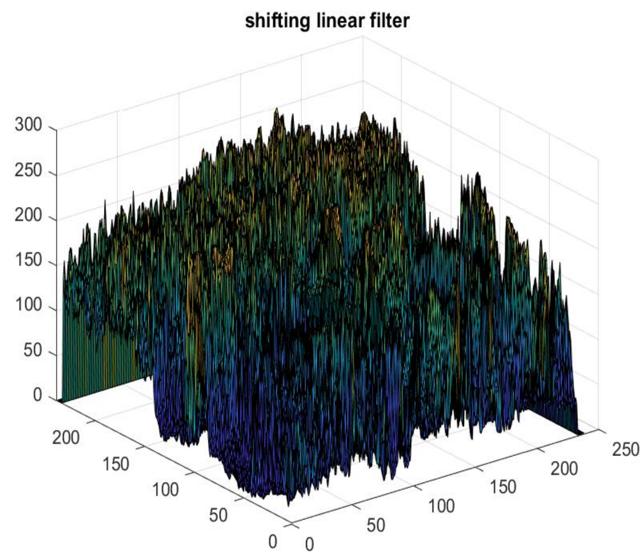

Figure 19: 3-D surface of linear shifting filter

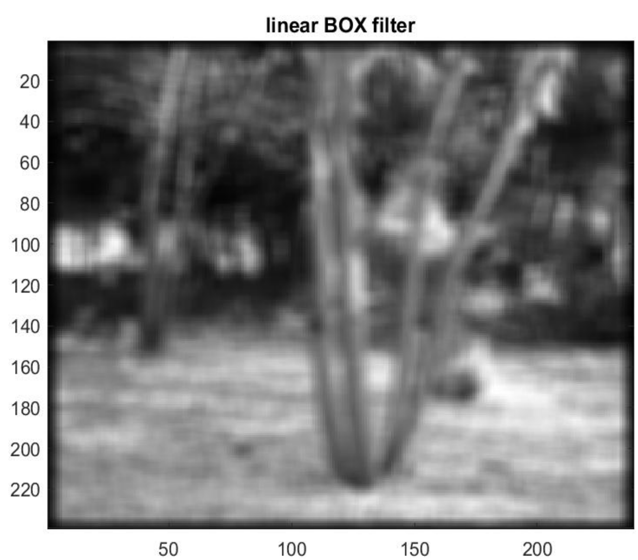

Figure 20: linear box filtered image

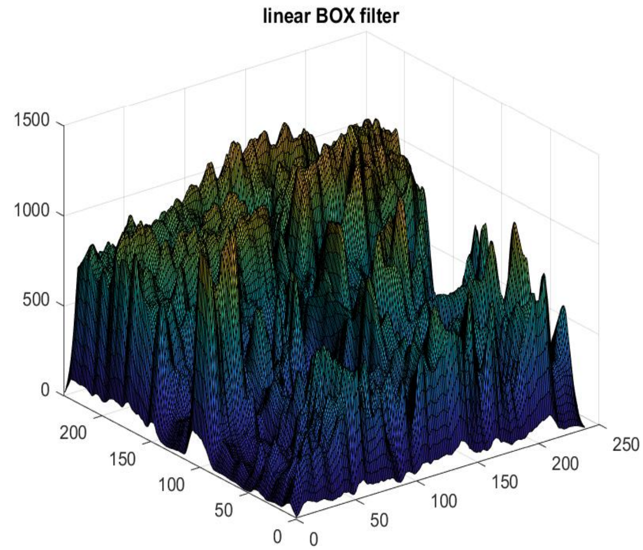

Figure 21: 3-D surface of the box filtered image

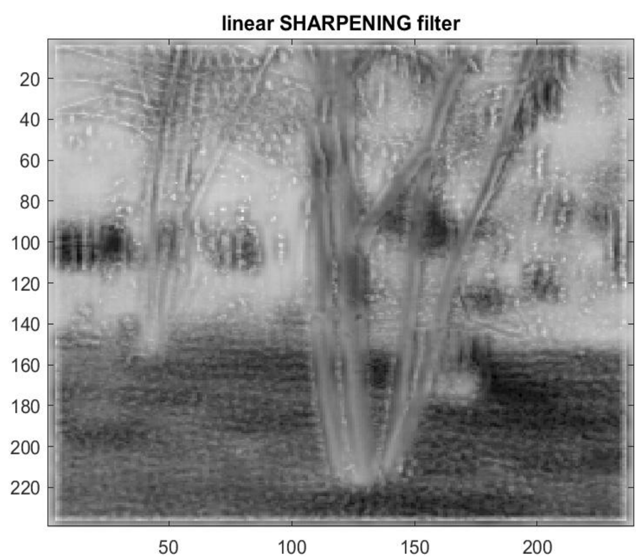

Figure 22: linear sharpening filtered image

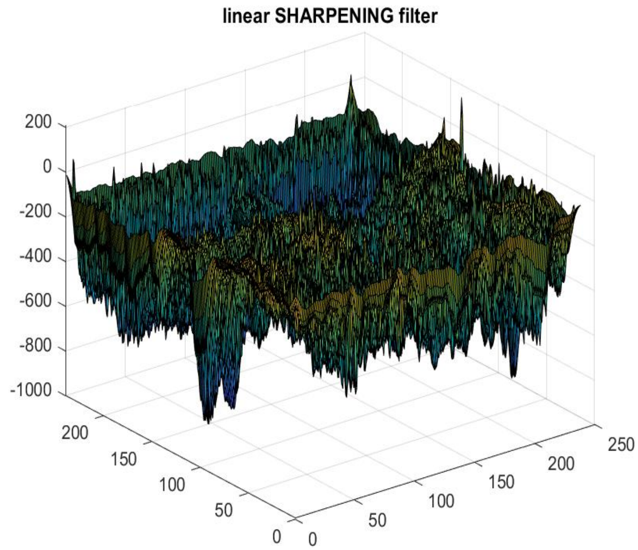

Figure 23: 3-D surface of the sharpening filtered image

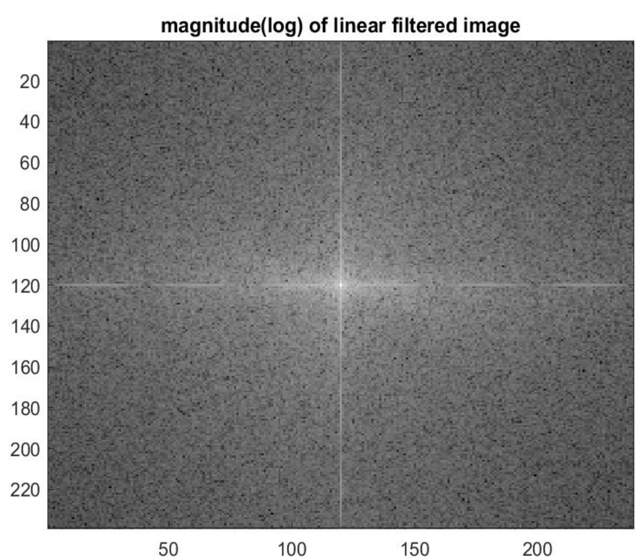

Figure 24: Fourier transformation of linear filter

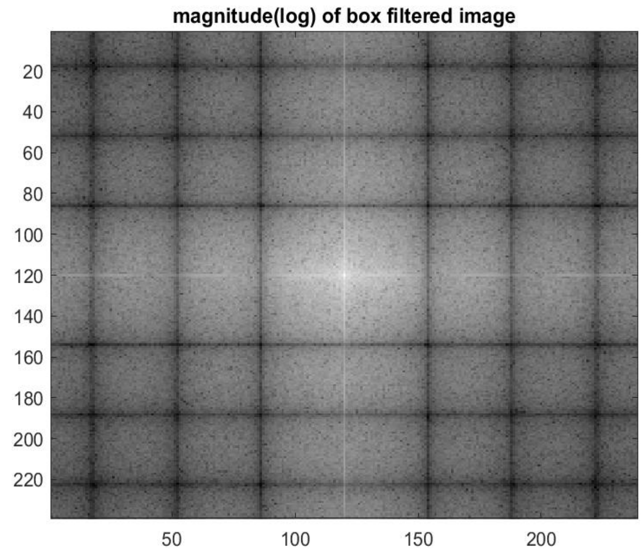

Figure 25: Fourier transformation of box filter

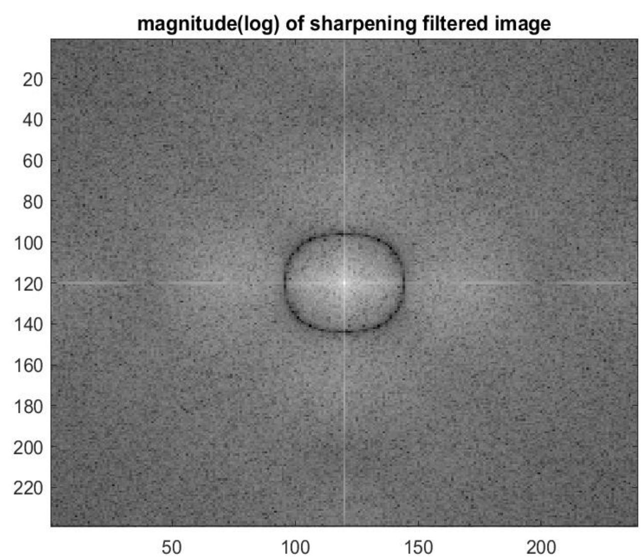

Figure 26: Fourier transform sharpening filter

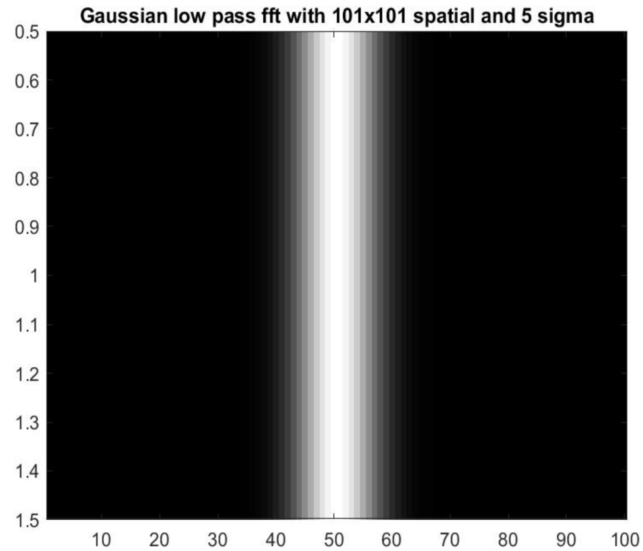

Figure 27: Gaussian low pass filter

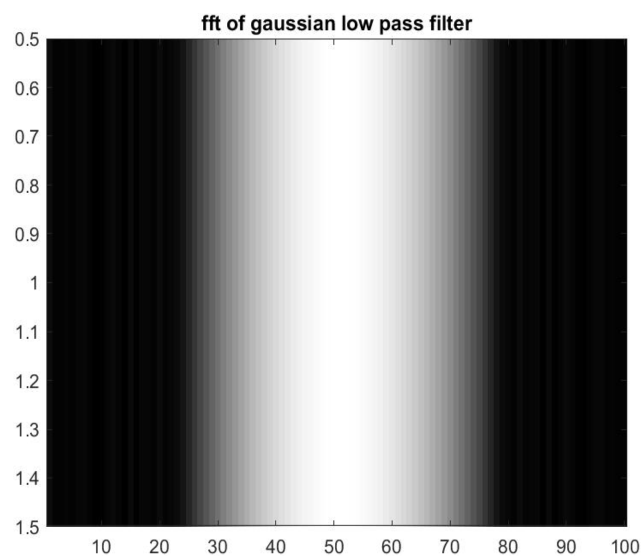

Figure 28: FFT of Gaussian low pass filter

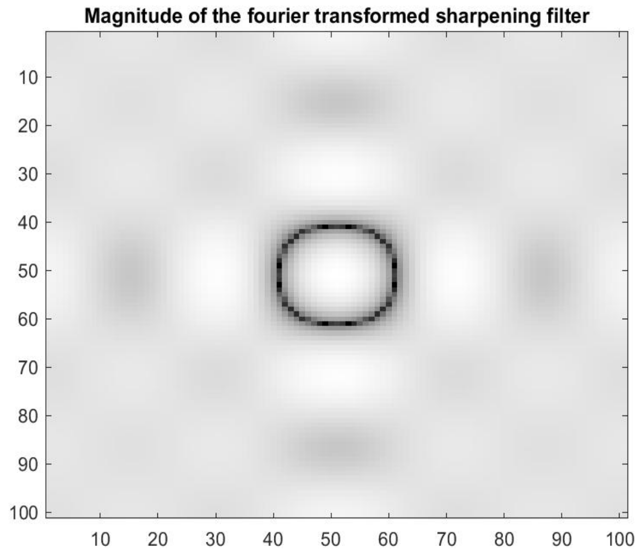

Figure 29: FFT of 101x101 linear filter with 7x7 filter copied in middle

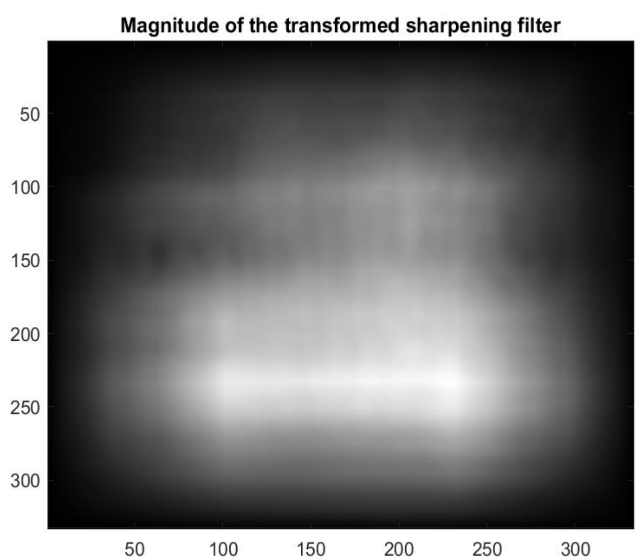

Figure 30: convoluted image with linear filter 101x101 with 7x7 filter copied in middle

0.5 Conclusion

In the conclusion, the effects of several variables on the results will be described.

1. When the given value of standard deviation is increased more noise is added to the image. and similarly when the density in salt and pepper noise is increased we get high value of added noise.

2. When a moving average filter is convoluted the image gets blurred to reduce the noise. It can be observed in the results as well, shown in figure 7 (which uses a filter of higher matrix value) is more blurred than figure 6.

3. Gaussian filter also blurs the image to reduce noise if the size of filter or sigma value increases.

4. Median filter decrease the effects of salt and pepper noise the bigger the size of filter the more effective it is.

5. Linear filter improves the quality of image by decreasing the size of image. It can also be used in shifting of images if the values of ones displaces from the centre.

6. A box filter basically blurs the image and when this filter is subtracted from the linear filter sharpening filter is created which brings more details in the image.

7. FFTs shown in figure describes what effects does different filters have on the image which are shown as magnitude (log) values in the images.