Answers to questions in Lab 1: Filtering operations

Name: Serkan Arda Yilal Program: Msc. Information and Network Engineering

Instructions: Complete the lab according to the instructions in the notes and respond to the questions stated below. Keep the answers short and focus on what is essential. Illustrate with figures only when explicitly requested.

Good luck!

Question 1: Repeat this exercise with the coordinates p and q set to (5, 9), (9, 5), (17, 9), (17, 121), (5, 1) and (125, 1) respectively. What do you observe?

Answers:
With each coordinate, we see the white point which has the value of 1 in different positions for Fhat and centered Fhat in Fourier domain. Whenever the coordinate changes, real value and imaginary value of its inverse Fourier transform also changes accordingly. While we are changing the coordinates, amplitude stays the same as white since we always have one point that has value 1 in the Fourier domain. On the other hand, phase changes similarly as real and imaginary value since as we discussed it contains the edge properties of the image and it looks similar to the image when we observe it intuitively.

Question 2: Explain how a position (p, q) in the Fourier domain will be projected as a sine wave in the spatial domain. Illustrate with a Matlab figure.

Answers:
In order to observe relationship between position (p,q) and the projected sine wave in the spatial domain, usage of centered Fhat will be more useful for us. When we observe the centered Fhat, as the centered (p,q) gets away from the origin, the frequency of sine wave in spatial domain will increase. Also the direction of sine wave will be defined by the direction of a line that we draw from the origin to the (p,q) in centered Fhat. It can be visualized as in the following:

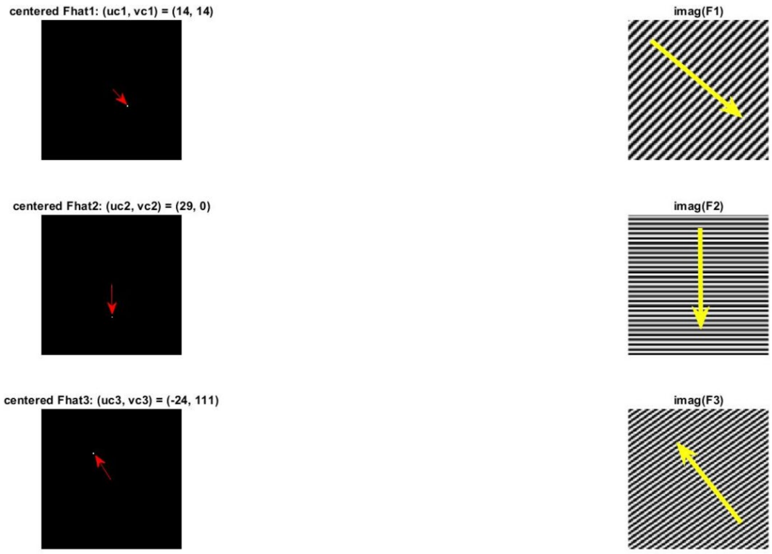

Figure 2.1 – Relationship Between Point Position and Sine Wave

Question 3: How large is the amplitude? Write down the expression derived from Equation (4) in the notes. Complement the code (variable amplitude) accordingly.

Answers:

In order to calculate the amplitude, we need to get the real and imaginary part of F then utilize the following expression:

$$|\mathrm{F}| = \sqrt{\mathrm{real(F)}^2 + \mathrm{imag(F)}^2}$$

From Equation (4), we derive real and imaginary parts by substituting sinusoidal functions into the exponential by using Euler`s formula and the equation becomes:

$$F(x) = \frac{1}{N} \sum_{u \,€\, [0 \ldots N-1]^2} Fhat(u)\ (\cos\frac{2\pi u^t x}{N} + j \sin\frac{2\pi u^t x}{N})$$

Where cosine represents the real part and sinus represents the imaginary part. In our code, we are directly able to use real() and imag() functions after finding F to calculate amplitude. We know that the amplitude corresponds to the peak value of Fourier Spectrum.

So, it can be calculated as in the following:

$$A = \frac{\max\,(|F|)}{N}$$

Which is 1/128.

Question 4: How does the direction and length of the sine wave depend on p and q? Write down the explicit expression that can be found in the lecture notes. Complement the code (variable wavelength) accordingly.

Answers:

After we transform (p,q) position into centered Fhat Fourier domain by finding uc and uv, we can find the direction of the sine wave by drawing a line from the origin to the (uc,uv) point which will be the direction. For wavelength, we can use the following expression stated in lecture notes:

$$\lambda = \frac{2\pi}{\sqrt{{w_1}^2 + {w_2}^2}}$$

Also, we know from lab notes that w corresponds to:

$$w_d = \frac{2\pi u}{N}$$

In our case, u and v are uc and vc for frequencies in x and y axis. So, the general expression for wavelength will be:

$$\lambda = \frac{2\pi}{\sqrt{\left(\frac{2\pi u_c}{N}\right)^2 + \left(\frac{2\pi v_c}{N}\right)^2}}$$

Question 5: What happens when we pass the point in the center and either p or q exceeds half the image size? Explain and illustrate graphically with Matlab!

Answers:

When we actually pass the point in the center, it means that p or q has exceeded half of the image size and it corresponds to passing to the other side of the domain in centered Fhat. To visualize it, we can show two examples where we are just behind exceeding the half of the image size and passing the half of the image size as in the following:

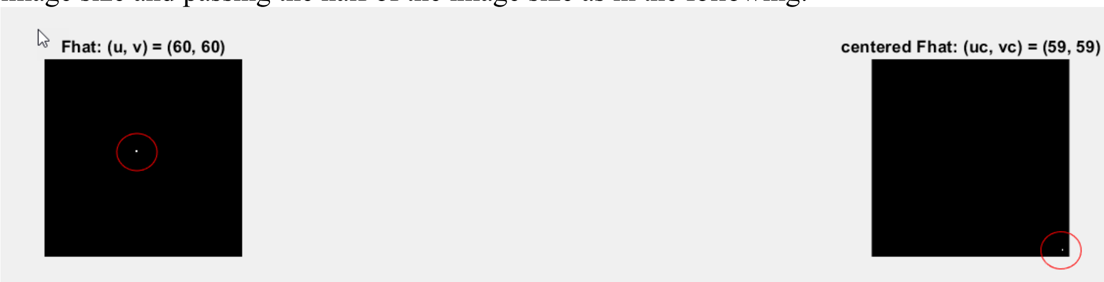

Figure 5.1- Fourier Transform of point (u,v) = (60,60) before and after centering

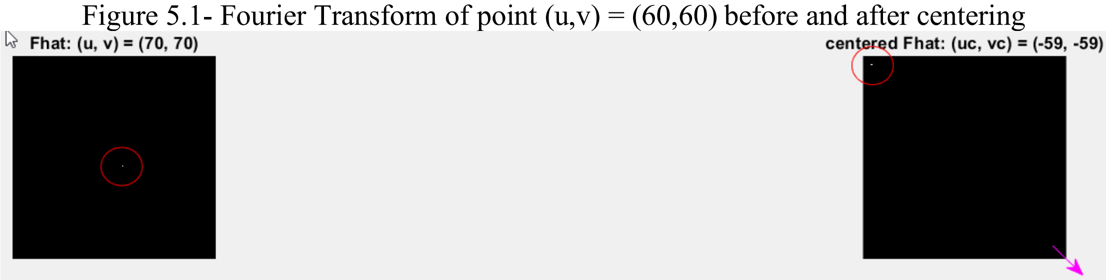

Figure 5.2- Fourier Transform of point (u,v) = (70,70) before and after centering

This happens in centered Fhat because intervals of each axis is [-pi,pi) while it is [0,2pi) in non-centered Fhat visualization.

Question 6: What is the purpose of the instructions following the question *What is done by these instructions?* in the code?

Answers:

This part of the code actually converts the (p,q) coordinate (which corresponds to (u,v) in fftwave() function to the coordinate in the centered Fhat. To do that, it checks if p or q exceeds the half of the image size. We subtract 1 from p and q so that at position (p,q) = (1,1) we can correspond to origin in centered Fhat which is (uc,uv) = (0,0).

Question 7: Why are these Fourier spectra concentrated to the borders of the images? Can you give a mathematical interpretation? Hint: think of the frequencies in the source image and consider the resulting image as a Fourier transform applied to a 2D function. It might be easier to analyze each dimension separately!

Answers:

If we analyze F and G separately in the beginning, it can be easier to understand the relationship between images and their Fourier spectra. For image F, we see a horizontal white line. This creates a frequency along with the y axis of the image. As a result, since the origin of the unshifted spectrum starts from the corner, we see the Fourier spectra concentrated on vertical border. The same case is valid for G with the difference that we have a vertical white line in the image which creates a frequency along with the x axis so it creates concentration on horizontal border in Fourier spectra.

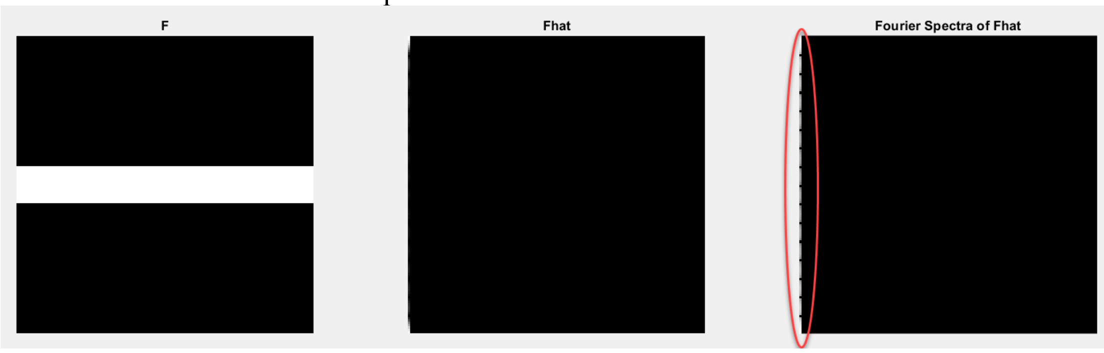

Figure 7.1-Vertical Border Spectrum of Image F

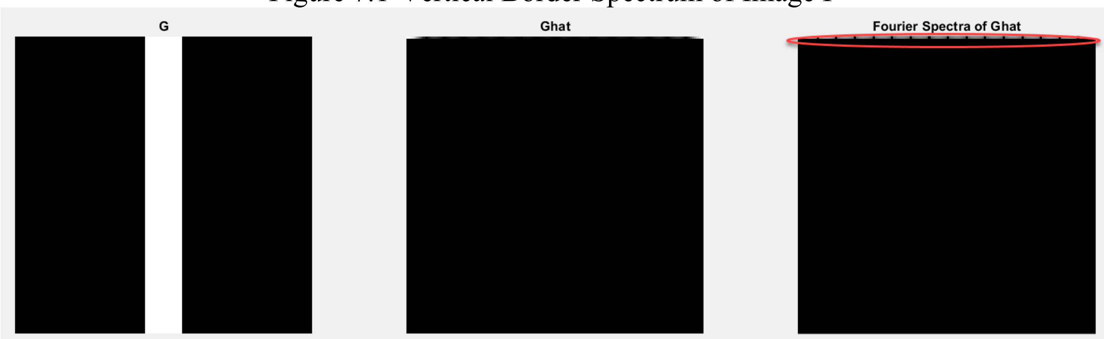

Figure 7.2-Horizontal Border Spectrum of Image G

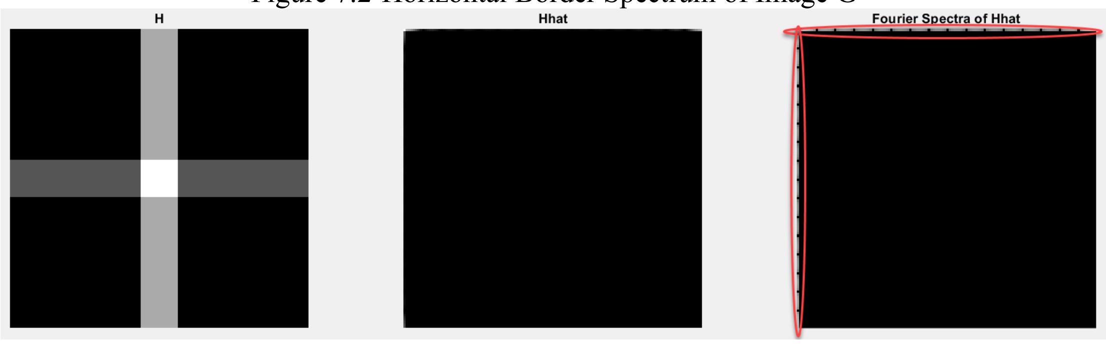

Figure 7.3-Additiveness correspondence of Spatial and Fourier Domain for F and G

To represent it in mathematical expression, we can use the following general formula for Fourier transform:

$$\hat{F}(u,v) = \frac{1}{\sqrt{MN}} = \sum_{m=0}^{M-1}\left[\sum_{n=0}^{N-1}\left[f(m,n)e^{-2\pi i\left(\frac{mu}{M}+\frac{nv}{N}\right)}\right]\right]$$

If we go through image G, we know that it is 1 for 57 <= n <= 72. This helps us to evaluate the above transform in two different dimensions:

$$\hat{F}(u,v) = \frac{1}{\sqrt{MN}} = \sum_{n=57}^{72}\left[e^{-2\pi i\left(\frac{nv}{N}\right)}\sum_{m=0}^{M-1}\left[e^{-2\pi i\left(\frac{mu}{M}\right)}\right]\right]$$

If we consider the second part of the equation where we have summation for m, we can consider it as Kronecker (discrete) delta function which is 1 only for m = 0. Hence, Fourier transform will have a non-zero value when m = 0 which is the reason of border effect.

Question 8: Why is the logarithm function applied?

Answers:

When we switch to the Fourier spectrum, the range of the values becomes so large (for example [0,2048] in Fourier Spectra of F) compared to the 8 bits in the display and this situation causes bright values in the center to dominate the others. In order to increase the visual detail in Fourier spectrum significantly, we apply logarithm function so that the range will not be large and domination will not occur.

Question 9: What conclusions can be drawn regarding linearity? From your observations can you derive a mathematical expression in the general case?

Answers:

From additions of images in spatial domain and their Fourier Spectrums, we can derive that linear operations in spatial domain corresponds the same action in Fourier Spectrums which can be represented with a mathematical expression as in the following:

$$\mathcal{F}[\, a * F(x,y) + b * G(x,y)] = a * Fhat(u,v) + b * Ghat(u,v)$$

Question 10: Are there any other ways to compute the last image? Remember what multiplication in Fourier domain equals to in the spatial domain! Perform these alternative computations in practice.

Answers:

In this example, we have multiplied two images and then took Fourier Transform of that multiplication. In order to achieve the same image, we can also directly take Fourier transforms of F and G individually and convolve them in Fourier Domain since multiplication in spatial domain corresponds in Convolution in Fourier Domain as in the following equation:

$$F(h.f) = F(h) * F(f)$$

Since we are working on the frequency domain and Matlab is not aware of scaling factor, we need to normalize the convolution in frequency domain by the size of the image which results with:

$$\frac{F(h) * F(f)}{128^2}$$

Then by taking inverse Fourier transform of the convolution in frequency domain and using fftshift() function in order to center the image, we get the following results which satisfies our first equation:

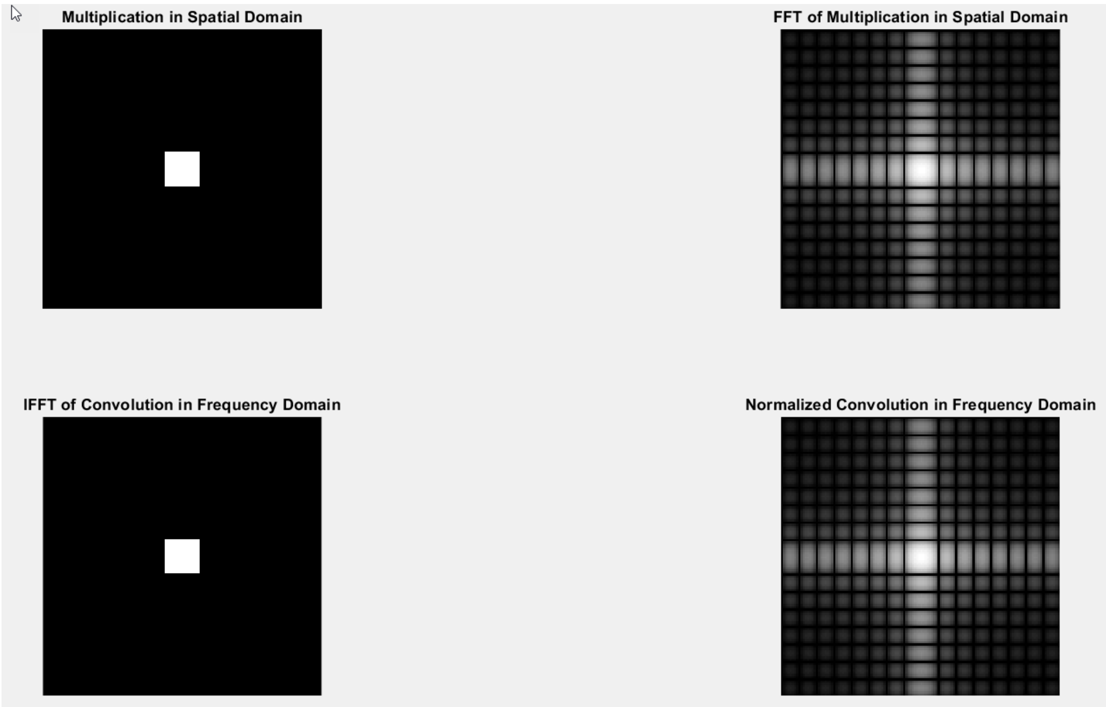

Figure 10.1- Comparison of Multiplication in Spatial Domain and Convolution in Frequency Domain

Question 11: What conclusions can be drawn from comparing the results with those in the previous exercise? See how the source images have changed and analyze the effects of scaling.

Answers:

When we compare the source images, we see that white area in the new image has half vertical length of the previous one (16 became 8) and twice horizontal length of the previous one (16 became 32). This change resulted in Fourier domain as a decrease of the wavelength of sinusoidal in vertical and increase of the wavelength in horizontal. This also means that frequency increased vertically and decreased horizontally. For that reason, when we observe Fourier spectrums of the images, we can see vertically it is spread to higher frequency and horizontally it is spread to lower frequency compared to the previous image.

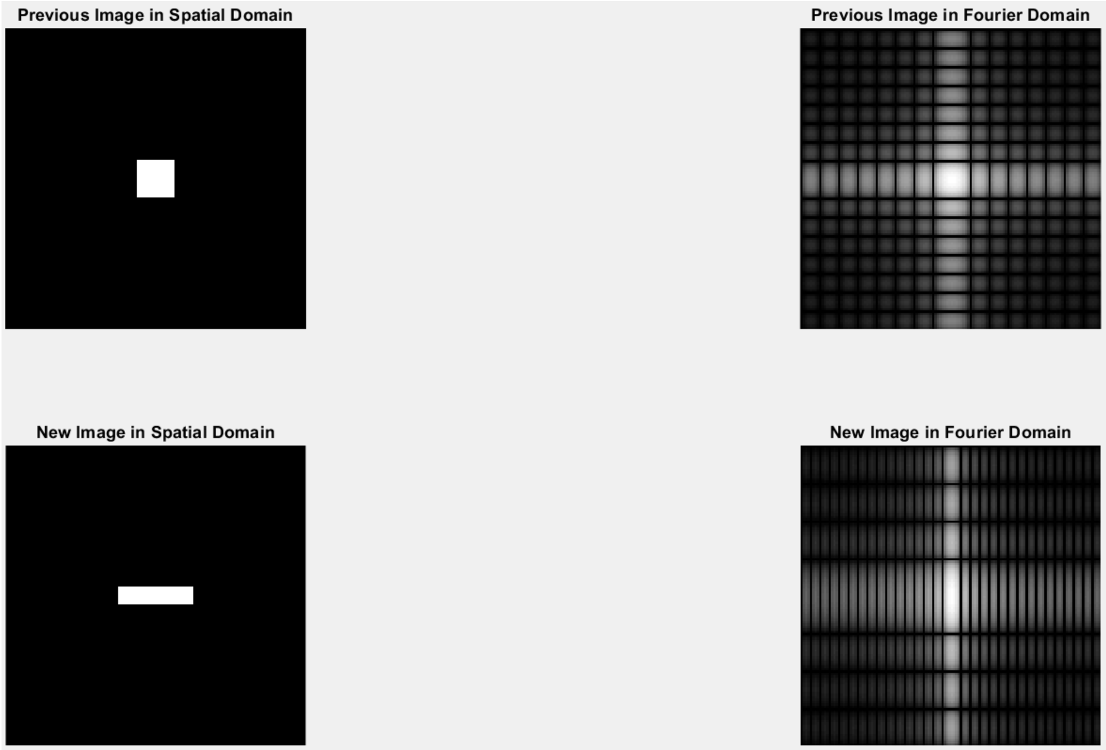

Figure 11.1- Effect of Scaling on Spatial and Fourier Domain

Question 12: What can be said about possible similarities and differences? Hint: think of the frequencies and how they are affected by the rotation.

Answers:

As we rotate the original image, we can observe that similar rotation also occurs in Fourier spectrum such that a frequency in the orthogonal direction of the change in the source image will occur.
Other than rotation in the frequency, we also see same additional waves in the Fourier spectrum of image for example with rotation 30° compared to the Fourier spectrum of original image. The reason is that 30° rotation of original image is not displayed with straight lines in spatial domain since diagonal pixels are not straight. Because of that, we see these wave effect in rotated Fourier transform while we don't see them in original Fourier transform due to white area can be represented with straight lines without illusion of empty dots.
For the inverse rotation of Fourier spectrum, we can say that in the center all of them looks similar to the Fourier Spectrum of the original image. But depending on the rotation angle, we see empty spaces in corners or borders or both. The reason is that in order to display rotated versions, we increase the size of the matrix and as a result, some of the parts represented empty considering we don't have any information regarding these parts.

The implementations in rotation can be observed as in the following:

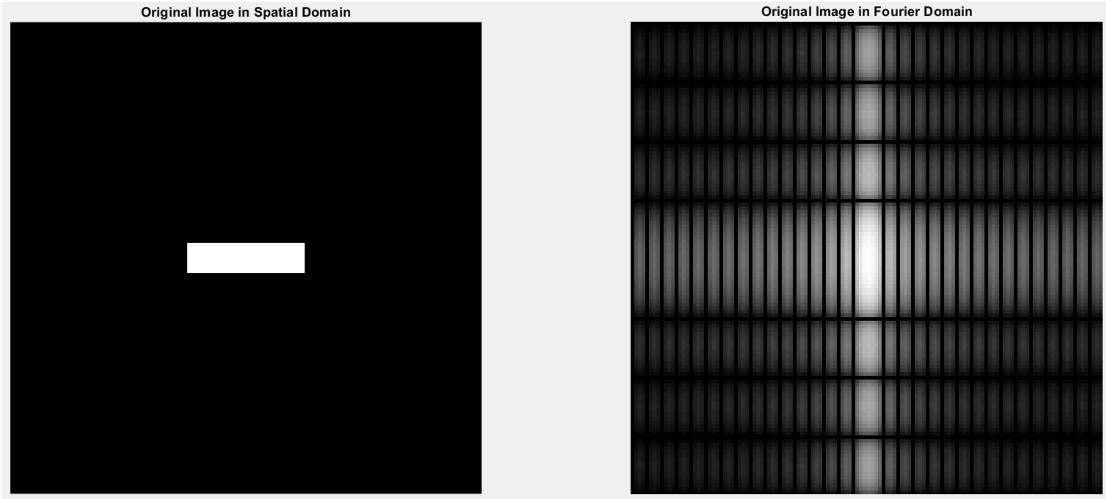

Figure 12.1-Spatial and Fourier Representation of Original Image

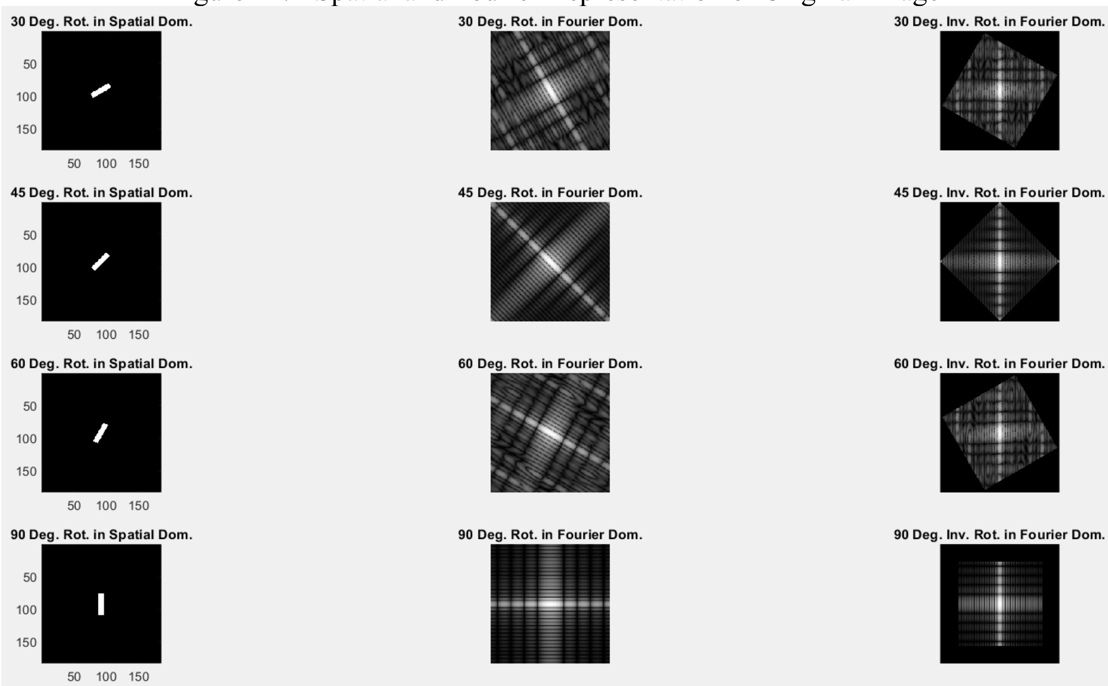

Figure 12.2- Representation of rotation for (30,45,60,90) in Spatial, Fourier Domain and inverse rotation in Fourier Domain

Question 13: What information is contained in the phase and in the magnitude of the Fourier transform?

Answers:

When we analyze original, pow2image, and randphaseimage versions of 3 different images, we see that even though we change amplitude in pow2image, we are still able to recognize what is contained in the image. The reason is that phase contains information about the location of the edges in the image.
On the other hand, when we keep amplitude and randomize the phase with randphaseimage, we are no longer able to recognize the original image. Because phase is the most important component in terms of recognizing the image intuitively. While the amplitude here defines what the grey levels or the intensities of the image are on either side of the edges, it cannot help us due to lack of edge information so it is less important.

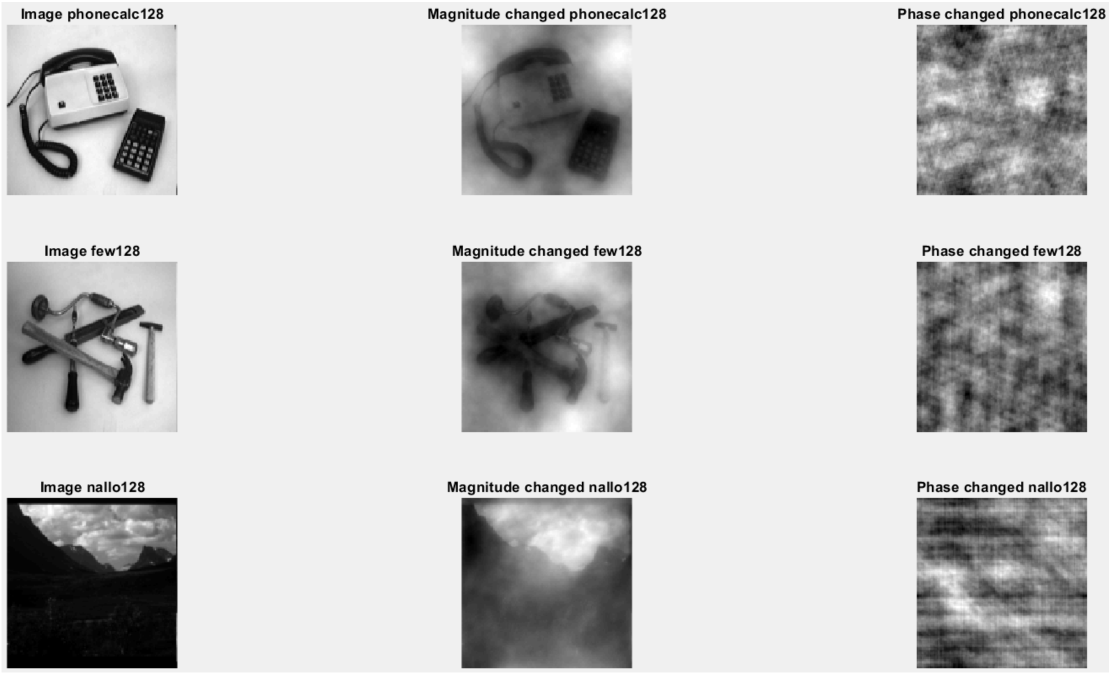

Figure 13.1-Effect of Magnitude Change and Phase Change on images phonecalc128, few128, and nallo128

Question 14: Show the impulse response and variance for the above-mentioned t-values. What are the variances of your discretized Gaussian kernel for t = 0.1, 0.3, 1.0, 10.0 and 100.0?

Answers:

Impulse responses for gaussfft function for different t values are as in the following:

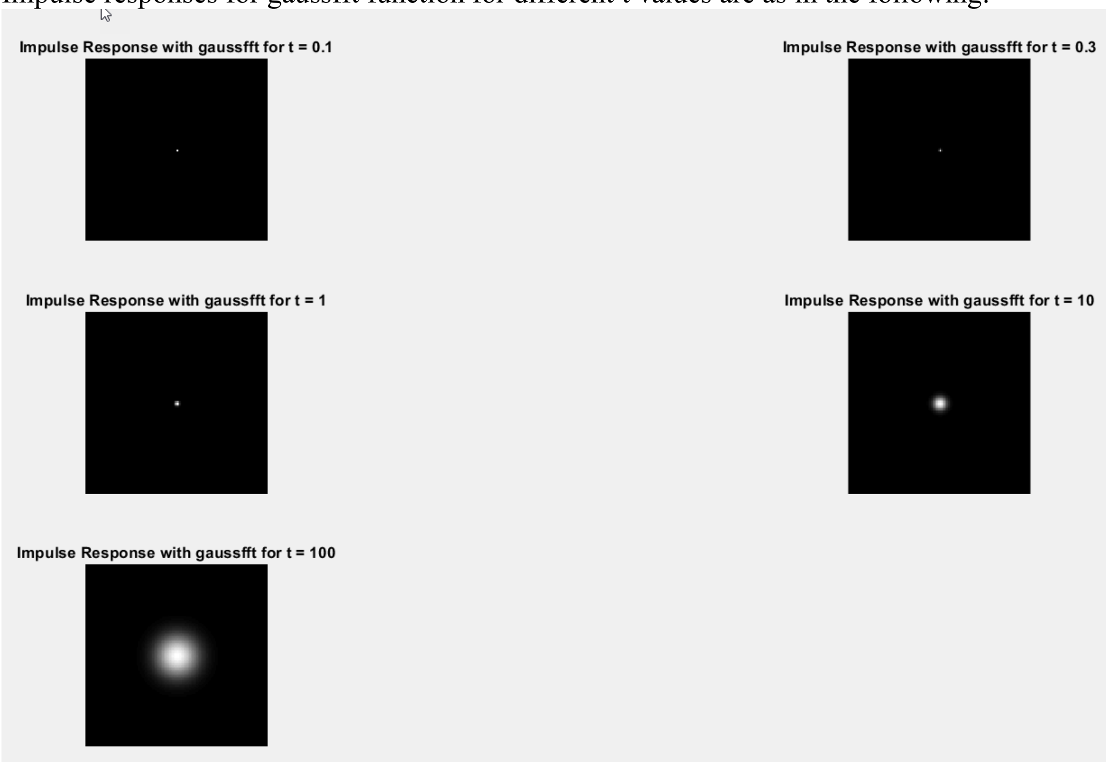

Figure 14.1- Impulse Responses for different t values(0.1,03,1,10,100)

The variance values for each t value are also calculated as in the following in covariance matrix form:

```
The spatial covariance matrix for t = 0.1 is:
    0.0133    0.0000
    0.0000    0.0133

The spatial covariance matrix for t = 0.3 is:
    0.2811    0.0000
    0.0000    0.2811

The spatial covariance matrix for t = 1 is:
    1.0000    0.0000
    0.0000    1.0000

The spatial covariance matrix for t = 10 is:
   10.0000    0.0000
    0.0000   10.0000

The spatial covariance matrix for t = 100 is:
  100.0000   -0.0000
   -0.0000  100.0000
```

Figure 14.2-Spatial Covariance Matrix Values for Computed Variance

Question 15: Are the results different from or similar to the estimated variance? How does the result correspond to the ideal continuous case? Lead: think of the relation between spatial and Fourier domains for different values of t.

Answers:

In order to analyze the differences between computed and estimated variance and also between computed and continuous case, absolute difference between these variances can be calculated as in the following:

Computed Variance Vs.
Estimated Variance

```
The spatial covariance matrix difference between computed and estimated variance for t = 0.1 is:
   0.0867    0.0000
   0.0000    0.0867

The spatial covariance matrix difference between computed and estimated variance for t = 0.3 is:
   0.0189    0.0000
   0.0000    0.0189

The spatial covariance matrix difference between computed and estimated variance for t = 1 is:
   1.0e-06 *

   0.2112    0.0000
   0.0000    0.2112

The spatial covariance matrix difference between computed and estimated variance for t = 10 is:
   1.0e-11 *

   0.4905    0.0012
   0.0012    0.4768

The spatial covariance matrix difference between computed and estimated variance for t = 100 is:
   1.0e-06 *

   0.5608    0.0000
   0.0000    0.5608
```

Figure 15.1- Absolute Difference Between Computed and Expected Variance Values

Computed Variance
Vs. Continuous Case

```
The spatial covariance matrix difference between computed variance and continuous case for t = 0.1 is:
   0.0867    0.0000
   0.0000    0.0867

The spatial covariance matrix difference between computed variance and continuous case for t = 0.3 is:
   0.0189    0.0000
   0.0000    0.0189

The spatial covariance matrix difference between computed variance and continuous case for t = 1 is:
   1.0e-06 *

   0.2112    0.0000
   0.0000    0.2112

The spatial covariance matrix difference between computed variance and continuous case for t = 10 is:
   1.0e-11 *

   0.1837    0.0001
   0.0001    0.1863

The spatial covariance matrix difference between computed variance and continuous case for t = 100 is:
   1.0e-06 *

   0.6718    0.0000
   0.0000    0.6718
```

Figure 15.2- Absolute Difference Between Computed and Continuous Case Variance Values

As we can see, for t < 1, the absolute differences between computed variance and other variances are higher while for t >= 1, it is much lower.

Question 16: Convolve a couple of images with Gaussian functions of different variances (like t = 1.0, 4.0, 16.0, 64.0 and 256.0) and present your results. What effects can you observe?

Answers:

For that question, images such as phonecalc128, few128, and nallo128 are convolved with Gaussian functions for different variances and the results were as in the following:

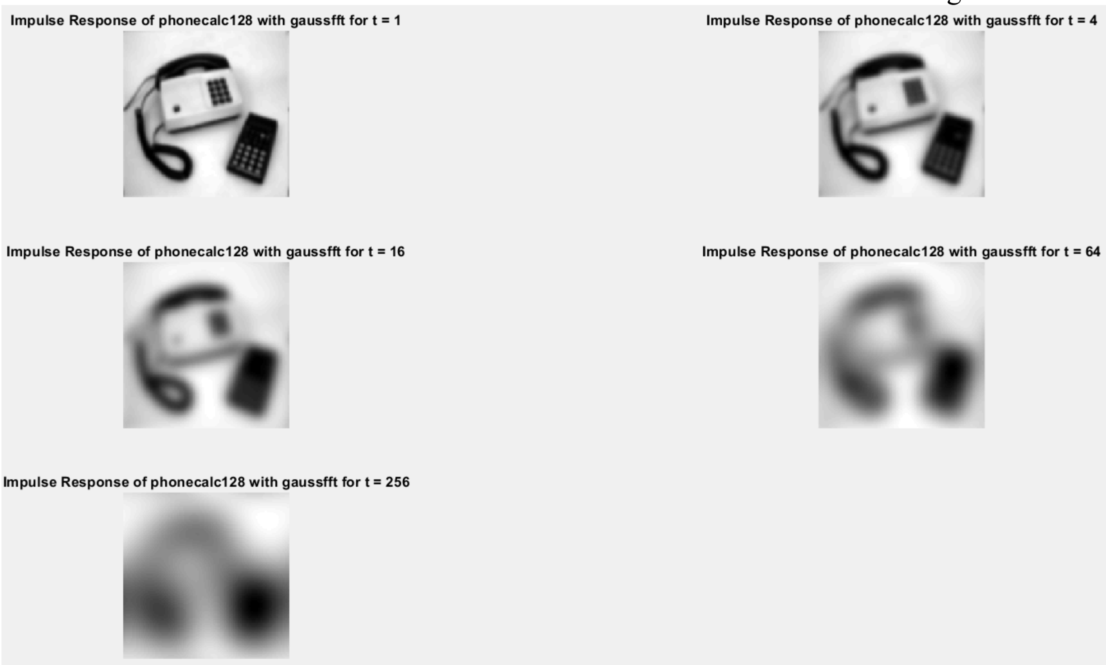

Figure 16.1- Impulse Responses of phonecalc128 with gaussian filter for t =(1, 4, 16, 64, 256)

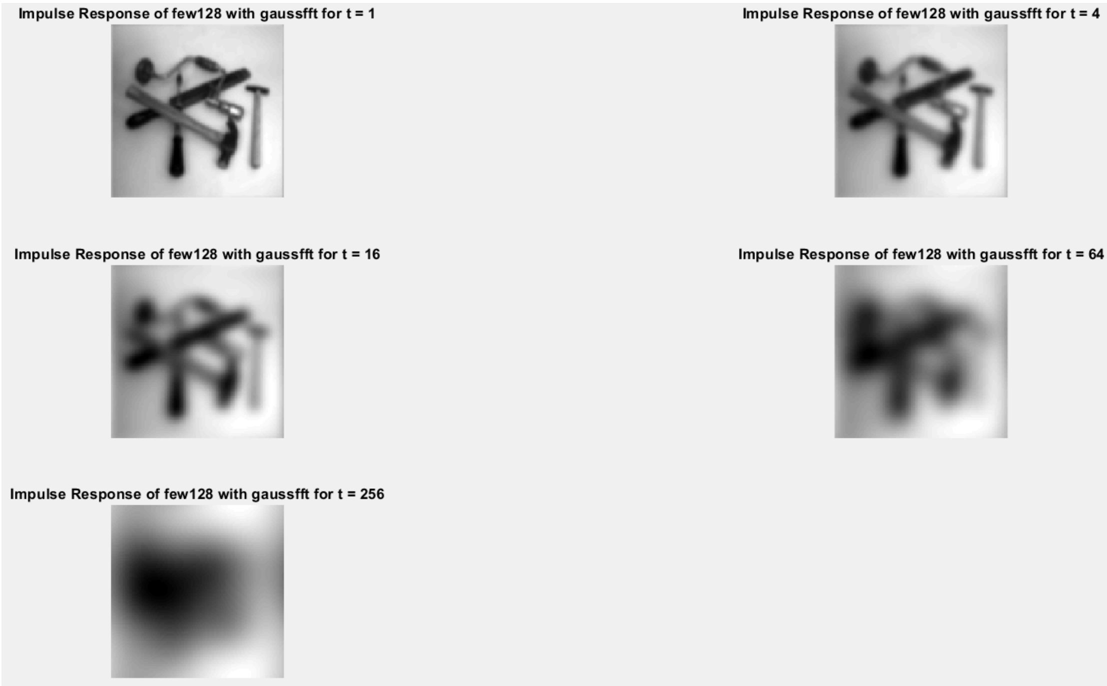

Figure 16.2- Impulse Responses of few128 with gaussian filter for t =(1, 4, 16, 64, 256)

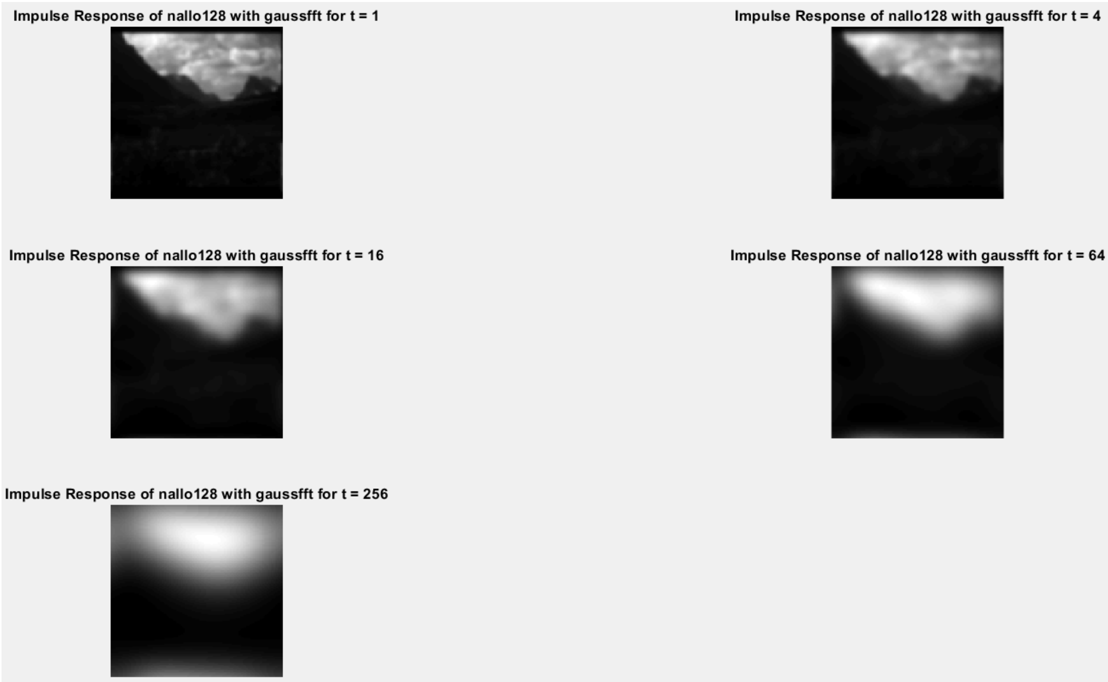

Figure 16.3- Impulse Responses of nallo128 with gaussian filter for t =(1, 4, 16, 64, 256)

For all images that we convolved with Gaussian Kernel, we can say that as we increase the variance, the images started to get blurred. The main reason is that as we increase the variance of the Gaussian Kernel, it corresponds to a lower cut-off frequency in Fourier Domain. This means as variance increases, less frequencies from higher frequencies are covered. This results with a behavior of low-pass filter which is the reason of blurring.

Question 17: What are the positive and negative effects for each type of filter? Describe what you observe and name the effects that you recognize. How do the results depend on the filter parameters? Illustrate with Matlab figure(s).

Answers:

<u>Gaussian Filter:</u>
+ Performs good on Gaussian Noised Image.
+ Good at smoothing image.
- Performs poor with Salt and Pepper Noised image.
- Due to smoothing, we lose information on edges.

<u>Median Filter:</u>

+ Performs good on both Gaussian Noised Image and Salt and Pepper Noised Image.
+ Preserves edges well depending on the window size.
- Output looks like painting after filtering.
- Some information is lost due to the spreading effect of filter on pixels.

Ideal Low Pass Filter:

+ Ideal in frequency domain.
+ Preserves information well.
- Creates ringing and blurring effect as cut of frequency decreases.
- Does not perform well on clearing noise on both Gaussian Noised and Salt and Pepper Noised Images.

Results of the noise additions and filtering operations can be seen as in the following illustrations:

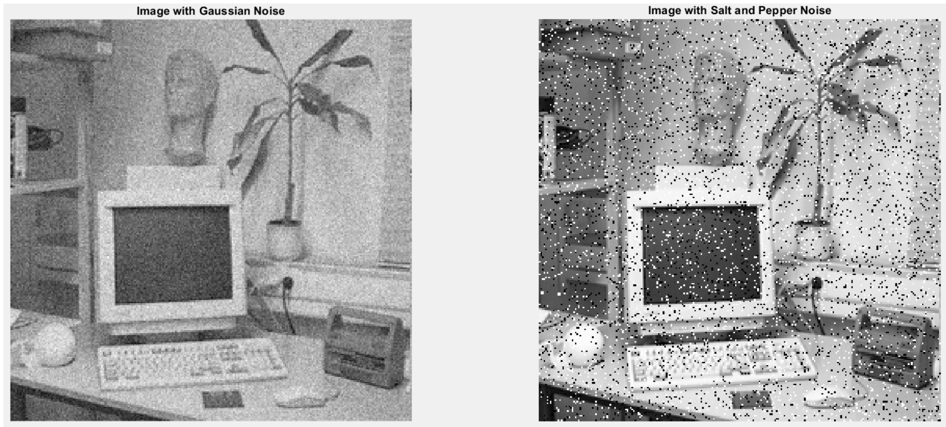

Figure 17.1-Image with Gaussian Noise and SAP Noise

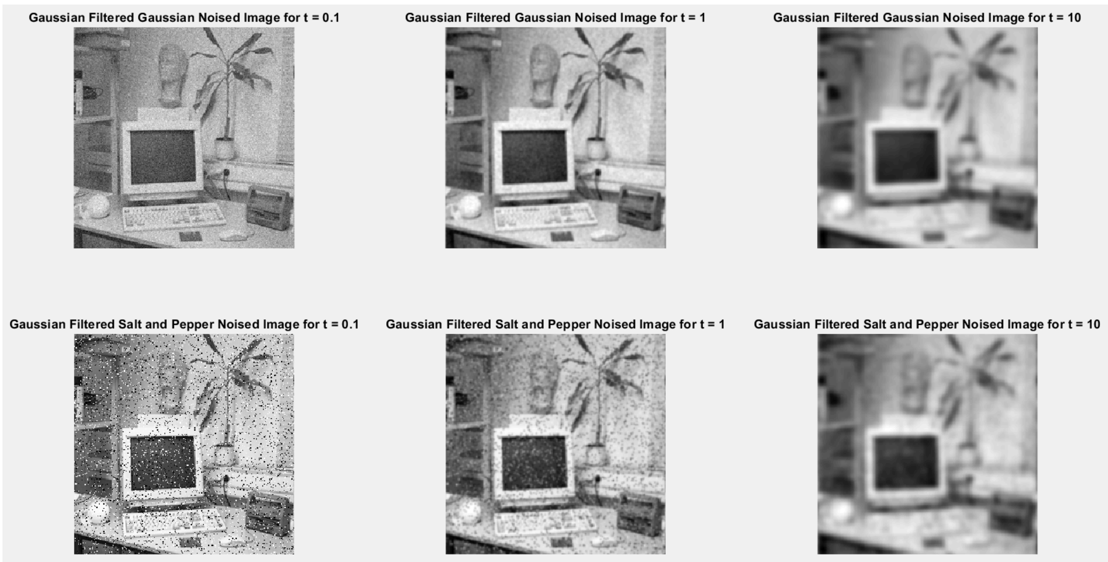

Figure 17.2-Effect of Gaussian Filter on Gaussian and SAP Noise for t=(0.1, 1, 10)

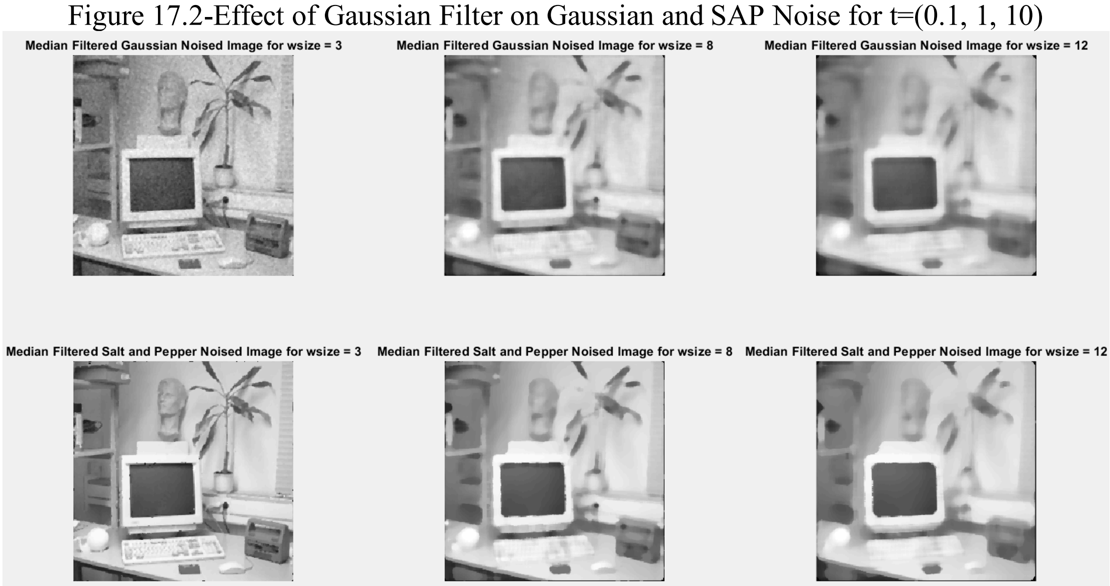

Figure 17.2-Effect of Median Filter on Gaussian and SAP Noise for wsize=(3, 8, 12)

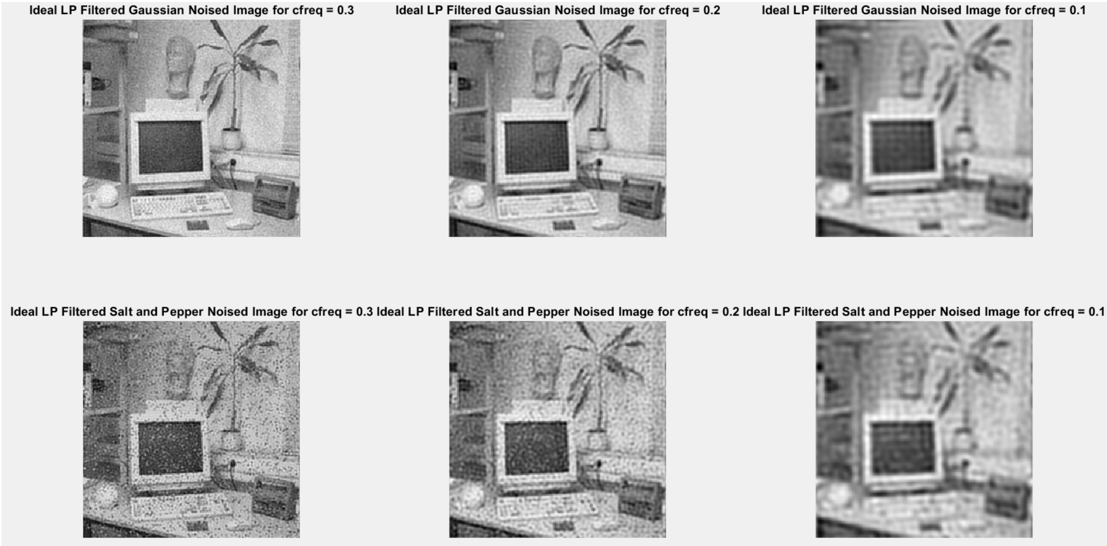

Figure 17.3-Effect of Ideal LP Filter on Gaussian and SAP Noise for cfreq=(0.3, 0.2, 0.1)

Question 18: What conclusions can you draw from comparing the results of the respective methods?

Answers:

-For Gaussian Filter: It performs really well on Gaussian Noised Image compared to the other two filters. But for SAP noised image, it performs poorly and adds only blurring as the variance is increased.

-For Median Filter: Considering its performance on both Gaussian and SAP noised images, it is the best filter on the average compared to the other filters. But as window size is increased, it starts to lose information and create a painting effect while the other filters do not encounter such information loss.

-For Ideal Low Pass Filter: Considering its effects on both noise types, it has the worst performance in comparison to the other filters. The reason is that while it is trying to remove the noise by suppressing higher frequencies, it creates ringing and blurring effect which shows less detail and looks like noise as well.

Question 19: What effects do you observe when subsampling the original image and the smoothed variants? Illustrate both filters with the best results found for iteration i = 4.

Answers:

It is clear that as we subsample the image, it starts to be distorted and produce artifacts after each subsampling operation. On the other hand, smoothed variants have higher durability against subsampling and they are less distorted for higher number of subsampling operations. The effects of subsampling and smoothing can be observed as in the following illustrations:

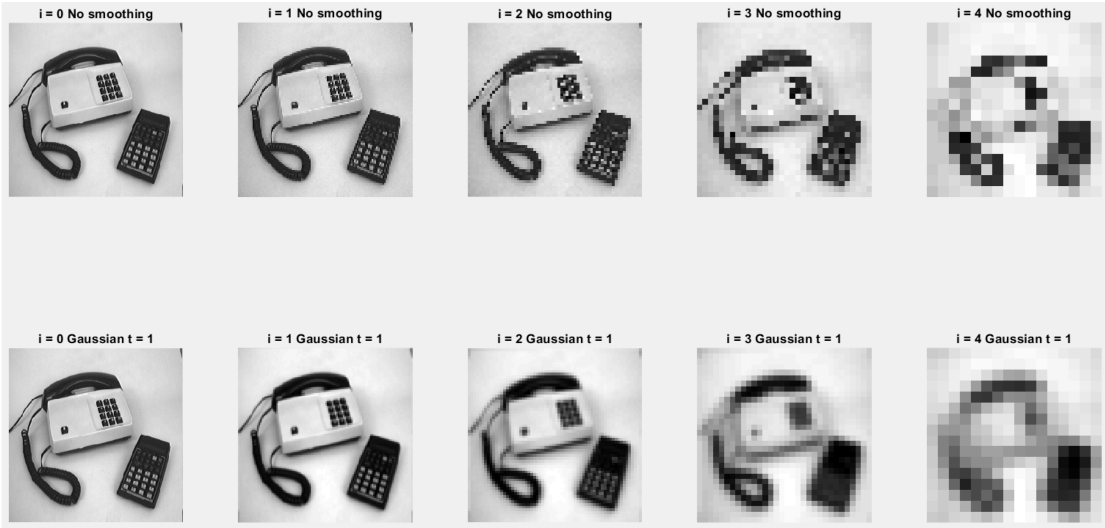

Figure 19.1- Comparison of subsampling without smoothing and with Gaussian Smoothing for t=1

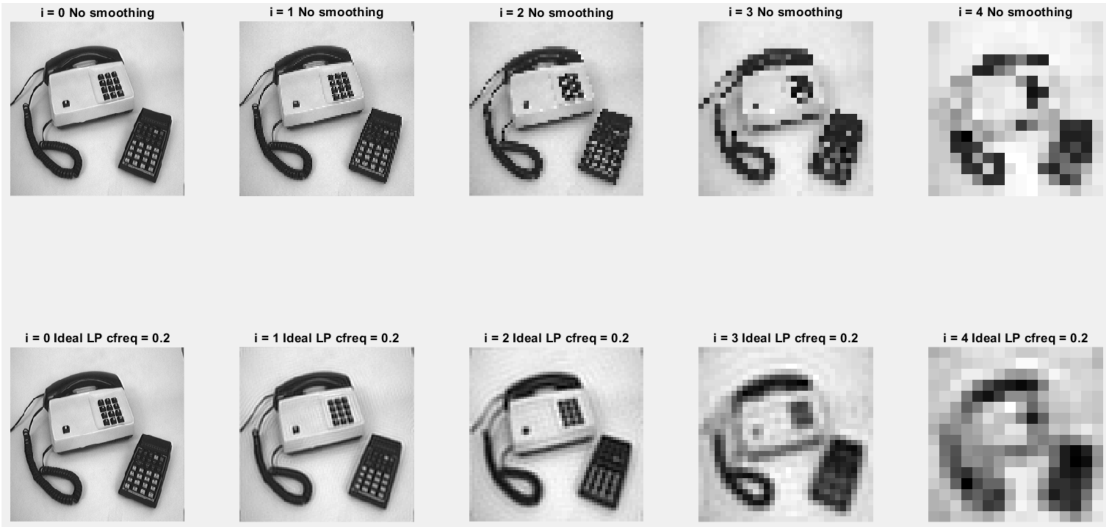

Figure 19.2- Comparison of subsampling without smoothing and with Ideal LP Smoothing for cfreq=0.2

Question 20: What conclusions can you draw regarding the effects of smoothing when combined with subsampling? Hint: think in terms of frequencies and side effects.

Answers:

In subsampling, we face the problem of having a sampling rate lower than the Nyquist rate which results with aliasing problem. When we apply smoothing, we see that since we are getting rid of higher frequencies, the effect of aliasing such as artifacts and structural distortions starts to disappear. Considering low frequencies still contains the basic and important features of the image, we are still able to identify images better compared to the non-smoothed sampling.
Still, we may face some side effects such as change in the color intensities for Gaussian Filtering and ringing effects for Ideal Low Pass Filter. In order to avoid from these side effects as much as possible, the related parameters for each filtering option should be optimized so that a sustainable preservation of quality can be achieved throughout ongoing subsampling operations.