Assignment 1 - DD2423 - Computer Vision and Image Processing

Alexandra Hotti

November 16, 2018

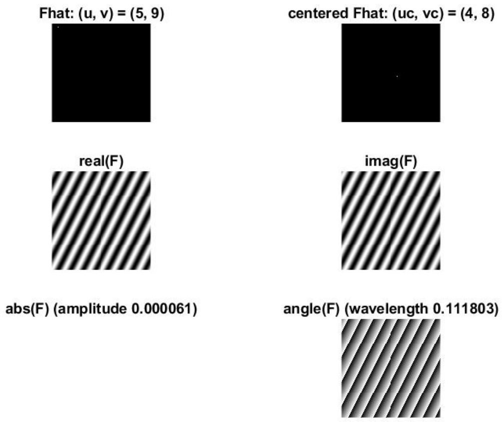
1 Properties of the discrete Fourier transform

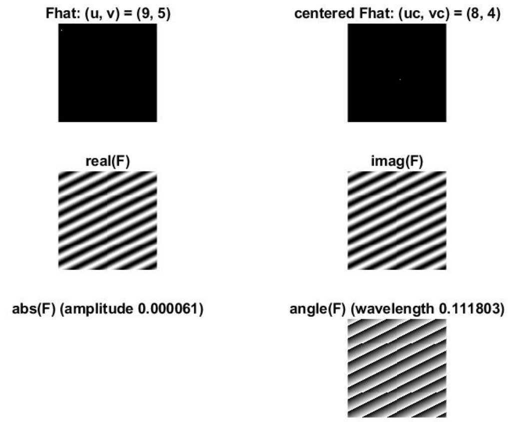
1.1 Basic Properties

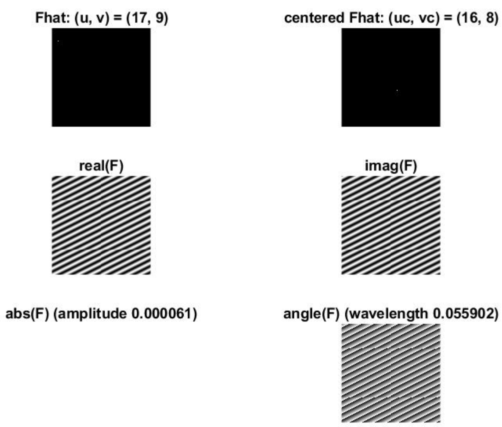
Question 1: Repeat this exercise with the coordinates p and q set to (5; 9), (9; 5), (17; 9), (17; 121), (5; 1) and (125; 1) respectively. What do you observe?

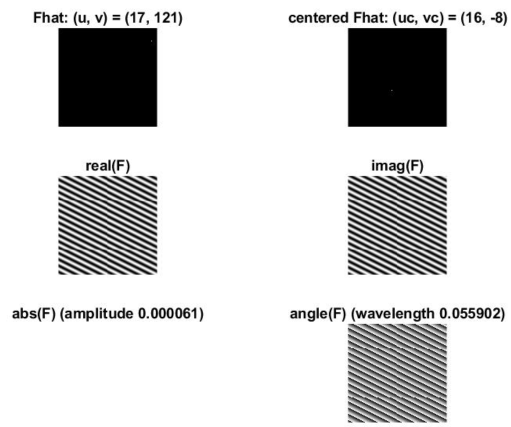
From the images below we can observe different things:

- **Frequency:** The larger the distance between (uc,vc) and the origin (0,0), the smaller the corresponding frequency becomes in the spatial domain. Thus we can infer that low frequencies are centered in the spatial domain and high frequencies can be found further from the center.
- **Magnitude:** In the spatial domain, all the images have the same amplitude. Since they all have the same magnitude in the Fourier domain.
- **Angle:** The direction of the wave in the spatial domain corresponds to the position of (uc,vc).
- **Real part:** Here we can see how the direction of the wave (cos(v)) corresponds to (uc,vc).
- **Imaginary part:** the imaginary part of the image is a sine wave. It corresponds to the real cosine part but it is shifted 90 degrees.


Fhat: (u, v) = (5, 9)
centered Fhat: (uc, vc) = (4, 8)
real(F)
imag(F)
abs(F) (amplitude 0.000061)
angle(F) (wavelength 0.111803)


Fhat: (u, v) = (9, 5)
centered Fhat: (uc, vc) = (8, 4)
real(F)
imag(F)
abs(F) (amplitude 0.000061)
angle(F) (wavelength 0.111803)


Fhat: (u, v) = (17, 9)
centered Fhat: (uc, vc) = (16, 8)
real(F)
imag(F)
abs(F) (amplitude 0.000061)
angle(F) (wavelength 0.055902)


Fhat: (u, v) = (17, 121)
centered Fhat: (uc, vc) = (16, -8)
real(F)
imag(F)
abs(F) (amplitude 0.000061)
angle(F) (wavelength 0.055902)

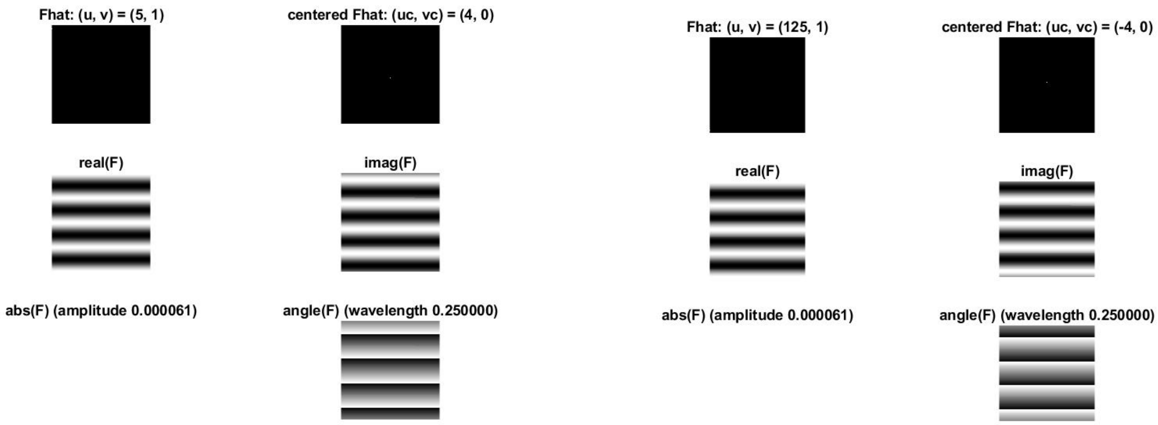


Figure 1: Corresponding images, Fourier transformations and their different components.

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.

If we do an inverse Fourier transform of the point/unit impulse in the Fourier domain, we can rewrite the expression by using Euler's formula from an exponential form into an expression consisting of a sine and cosine component:

$$f(m,n) = \frac{1}{\sqrt{MN}} \sum \sum \hat{f} \cdot (cos(2\pi(\frac{mu}{M} + \frac{nv}{N}) + i \cdot sin(2\pi(\frac{mu}{M} + \frac{nv}{N}))) \tag{1}$$

where $\hat{f}$ is given by:

$$\hat{f} = \delta(u - p, v - p) \tag{2}$$

In the images below one can see

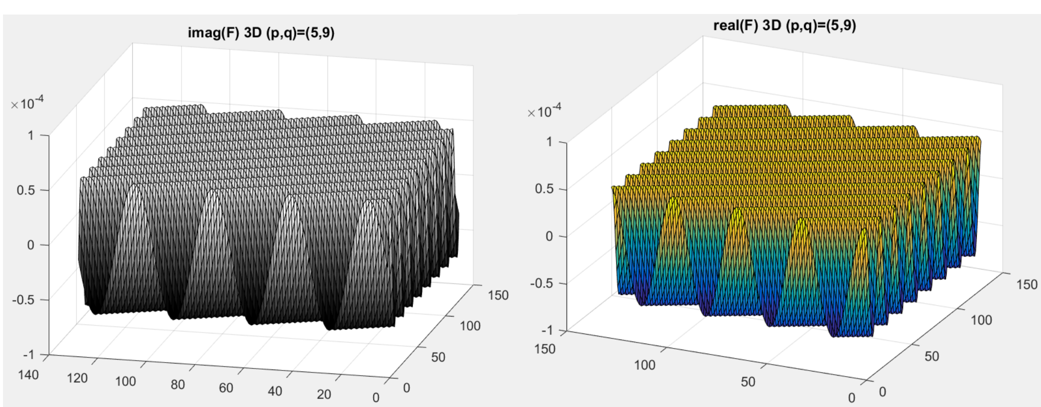


Figure 2: 3D plot of the imaginary and real parts of the

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

For refrence note that:

$$\delta(p, q) = 0 \tag{3}$$

for all $(p, q) \neq 0$.

Dirac's sifting property gives us:

$$\sum \sum f(x, y)\delta(x - x_0, y - y_0) = f(x_0, y_0) \tag{4}$$

Thus, performing the inverse Fourier transform on the unit impulse gives us:

$$f(m, n) = \frac{1}{\sqrt{MN}} \sum^{M-1} \sum^{N-1} F(u, v)e^{2\pi i(\frac{mu}{M} + \frac{nv}{N})}$$

Using that $M = N$ and that $F(u, v) = \delta(u - p, v - q)$

$$f(m, n) = \frac{1}{MN} \sum^{M-1} \sum^{N-1} \delta(u - p, v - q)e^{2\pi i(\frac{mu+nv}{M})}$$

Using the sifting property.

$$f(m, n) = \frac{1}{M} e^{2\pi i(\frac{mu+nv}{M})}$$

Finally, using Euler's formula leads to.

$$f(m, n) = \frac{1}{M}(cos(2\pi \cdot \frac{pm + qn}{M}) + i \cdot sin(2\pi \cdot \frac{pm + qn}{M})) \tag{5}$$

Thus, the amplitude for the above expression is given by:

$$max(f(m, n)) = max(\frac{1}{M}(cos(2\pi \cdot \frac{pm + qn}{M}) + i \cdot sin(2\pi \cdot \frac{pm + qn}{M}))) = \frac{1}{M} \cdot 1 = \frac{1}{M} \tag{6}$$

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.

Wavelength:

$$\lambda = \frac{1}{\sqrt{p^2 + q^2}} \tag{7}$$

Direction: The directions of the wave in the spatial domain is given by the direction of a vector computed as the difference between (p,q) and the origin (0,0) in the Fourier domain.

Thus higher frequency correspond to a larger distance between the origin and (p,q) which gives a shorter wave length.

Below you can see both how the direction and wavelength correspond to the position of the point in the Fourier domain in Figure 3.

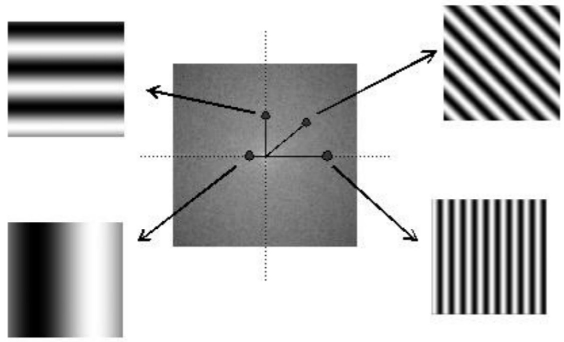


Figure 3: Relationship between a point in one domain and the corresponding direction of the waveform in the other domain.

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!

A fourier domain is periodic. Matlab places the origin in the upper left corner of the image. However, we would like for the zero frequencies to end up in the center of the image. We can use fftshift to handle the periodicity of the fourier domain while centering the point. Thus, fftshift switches the first quadrant with the third quadrant and the second quadrant with the fourth quadrant.

For instance (p,q) =(17,121) gets the centered coordinates (16,-8) Thus, the point will be moved from the 2:nd to the 3:rd coordinate. See Figure 4

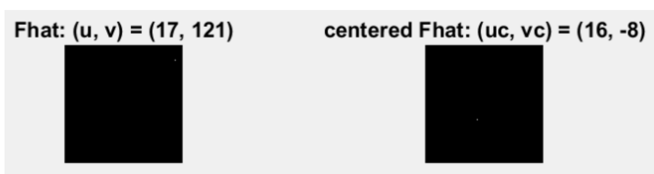


Figure 4: An example of what happens when we pass the point in the center and either p or q exceeds half the image size.

In general the points are shifted according to Figure 5

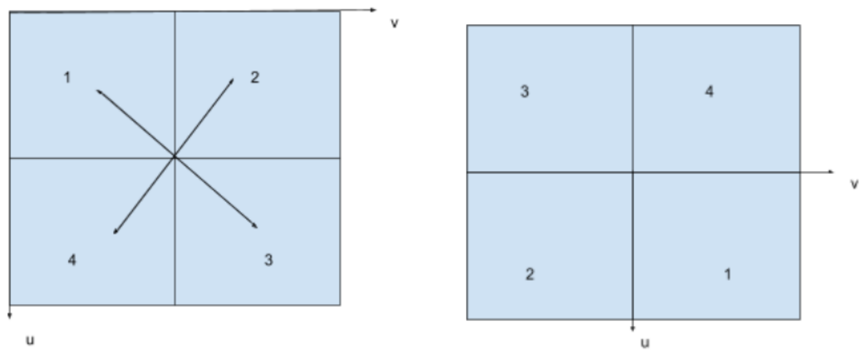


Figure 5: What happens in general when we pass the point in the center and either p or q exceeds half the image size

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

This part of the code calculates the centered coordinates of (p,q) in the interval [-pi,pi]. Matlab originally centers the image so that we are observing the interval [0,2pi]. Then minus one is added because matlab uses (1,1) as its center. Lastly, fftshift(Fhat) is used to swap the quadrants like in Figure 5.

1.2 Linearity

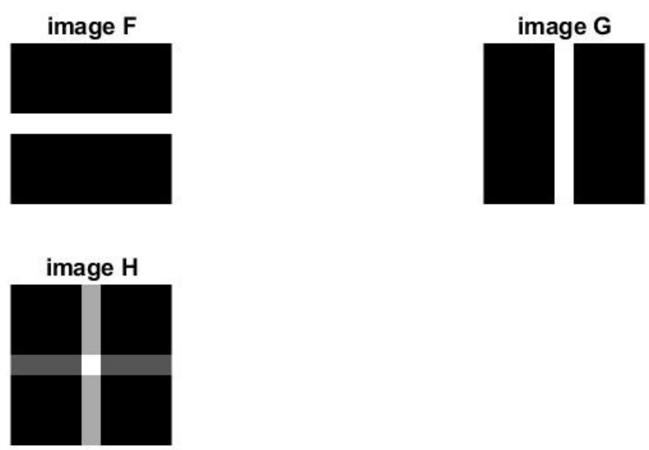


Figure 6: Images used to display the linearity property of the Fourier transform.

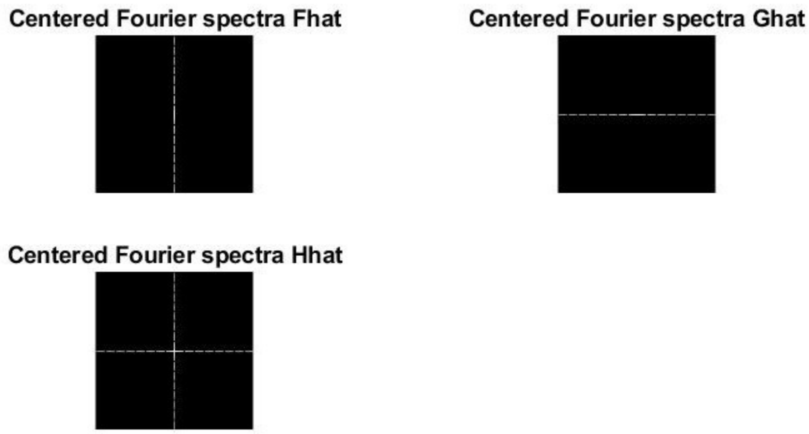


Figure 7: Fourier transformations of images used to display the linearity property of the Fourier transform.

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!

For instance, in the image of F in Figure 5 we can see that in the x direction no change occurs, thus there are only smooth changes. Along the x direction at different points of y we either have only black or only white along an entire row. In the y direction however, we have sharp contrasting edges and thus we get quick changes. Since when we move solely along the y direction the image changes quickly

Therefore, in the Fourier spectrum we will only get frequencies in the y-direction in Figure 8. Mathematically this will result in a dirac function in the y -direction.

Also, if we look at the fourier spectrum in 3D we can see that the rectangle becomes a sinc function along the axis (we observe its magnitude, the abs value). Which is consistent with box plots.

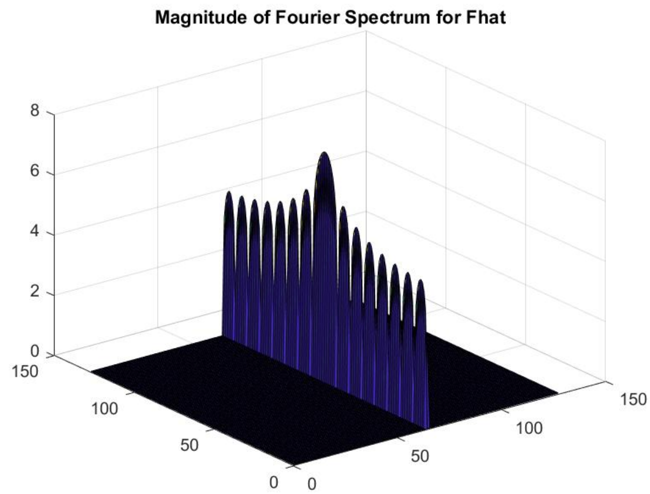


Figure 8: The fourier spectrum becomes a sinc function in 3D.

Because of the dirac function, we can observe how the sinc function only is defined along one of the axes. Since the dirac function will equal 1 only along the u axis in the Fourier domain of F for instance and will equal 0 at all other positions.

Question 8: Why is the logarithm function applied?

We want to be able to see all the frequencies that make up the image. Thus, we want to enhance frequencies with low magnitudes. Since the brightness of the frequencies with high magnitudes overtakes the image so that we cannot see frequencies with low magnitude, therefore these frequencies look black as if they were not components of the image.

The logarithmic function achieves this by suppressing high magnitudes and enhancing low magnitudes.

When we have an image, which results in a Fourier spectrum with a wide range of values and these values are scaled for instance a 8 bit system the brightest pixels will dominate the display at the expense of lower values of the spectrum. Thus we could get a spectrum where the majority of the image appears black if there is a large spread between a few of the really high values and the smaller values [1].

The solution is to use a log transform which enhances small pixel values and compresses high pixel values.

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

The Fourier transform is a linear operation. Therefore, it does not matter if we sum (and scale) the images before or after we transform them.

$$\begin{aligned} 1 \cdot f + 2 \cdot g &= h \\ F(h) = F[1 \cdot f + 2 \cdot g] &= 1 \cdot \hat{f} + 2 \cdot \hat{g} \end{aligned} \tag{8}$$

In general:

$$
\begin{aligned}
F[af_1(m,n) + bf_2(m,n)] &= a\hat{f}_1(u,v) + b\hat{f}_2(u,v) \\
af_1(m,n) + bf_2(m,n) &= F^{-1}[a\hat{f}_1(u,v) + b\hat{f}_2(u,v)]
\end{aligned}
\tag{9}
$$

1.3 Multiplication

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.

Multiplication in one of the domains (spatial or Fourier) corresponds to convolution in the other domain.

$$F(fg) = F(f) * F(g) \tag{10}$$

Thus, we can either:

- Multiply the two images directly in the Fourier domain
- Or we can use the Fourier transform on them separately, use convolution and then apply the inverse Fourier transform to get the same image.

Below is an example. The first Figure 9 contains the result of multiplication between the two images F and G in the spatial domain.

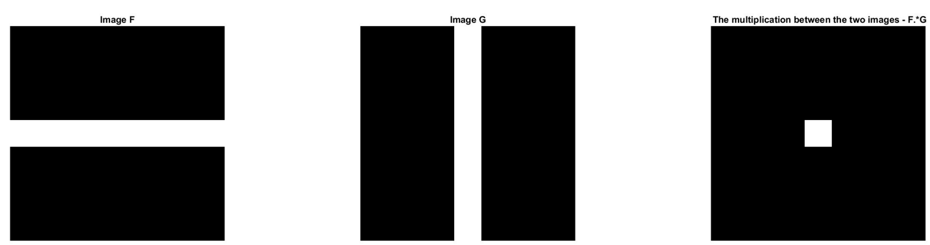


Figure 9: The multiplication between two images F and G in the spatial domain.

Next, here to the left in Figure 10 is the result from convolution between the individual Fourier transformations between F and G in the Frequency domain. To the right is instead the Fourier transform of the multiplication between F and G in the spatial domain.

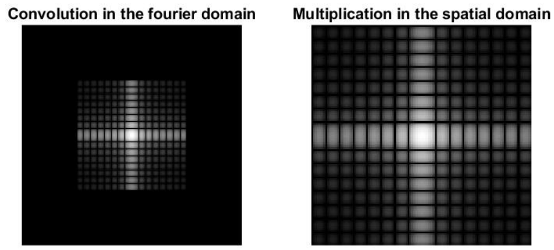


Figure 10: To the left is the result from convolution between the individual Fourier transformations between F and G in the Frequency domain. To the right is instead the Fourier transform of the multiplication between F and G in the spatial domain.

1.4 Scaling

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

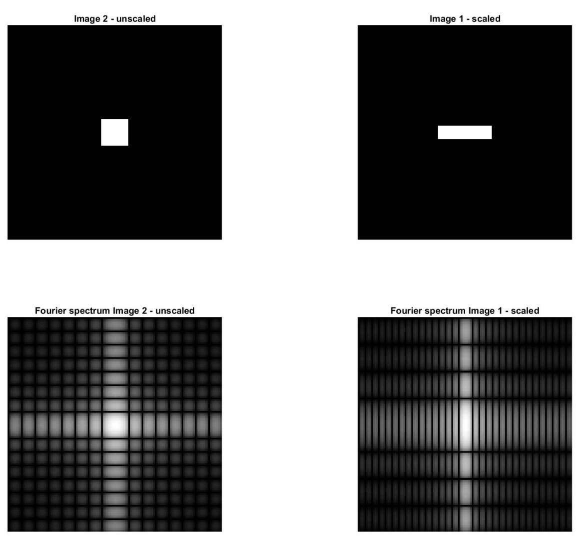


Figure 11: Results for

The original spatial image has been suppressed in the y-direction and expanded in x-direction. Which gives us the opposite result in the Fourier domain. In the Y- and in the X-directions we get:

Y-direction - Fast change: compression spatial → faster change → thus we need more high frequencies. → compression in Fourier (more area/magnitude further out by high frequencies) →from above we can see that the sinc function becomes more spread out.

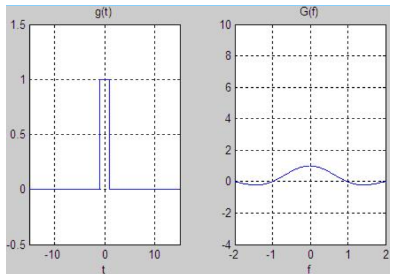


Figure 12: Fast changes in y-dir results in a spread out sinc function.

X-direction - Slow change: expansion spatial → slower change → thus we need less high frequencies instead we need low frequencies (smooth changes). → compression in Fourier. The sinc function becomes compressed

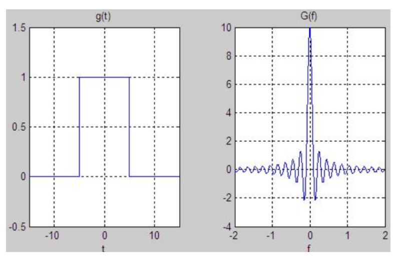


Figure 13: Slow changes in x-dir results in a sinc function with most of its mass around the center were the low frequencies are.

1.5 Rotation

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

A rotation in either the Fourier or the spatial domain becomes a rotation in the other domain. The rotation does not affect the magnitude of the Fourier spectrum, but it rotates the image and the Fourier transformation by the same angle see Figure14.

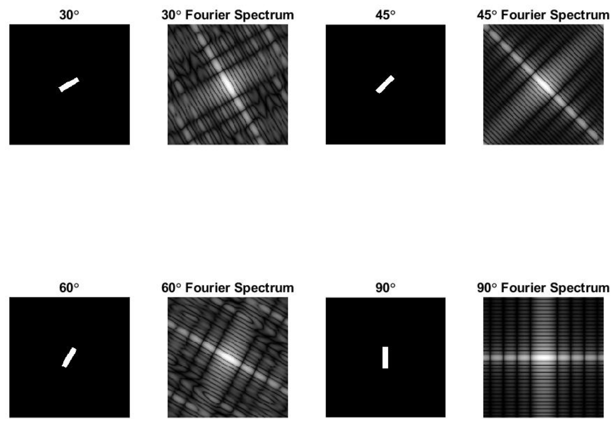


Figure 14: A rotation in one domain become a rotation in the other domain.

Artifacts: The pixels cannot become perfectly straight when we shift the image. Since they do not "fit" into the matrix grid. Therefore, we get some high frequencies to represent these artifacts. This can be seen for instance in the image with the 30-degree rotation see Figure15.

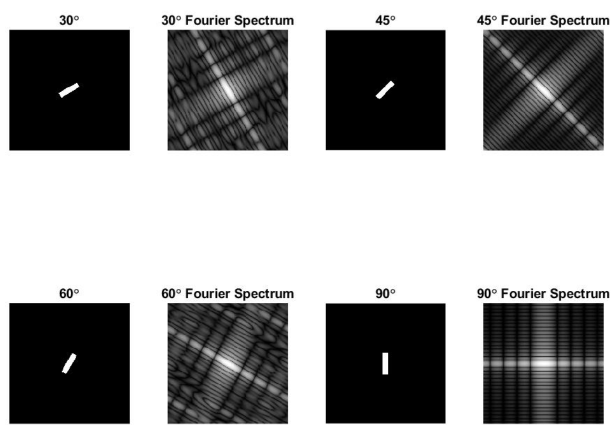


Figure 15: Rotating an image by 30°.

1.6 Fourier phase and magnitude

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

Phase: The phase defines how wave forms are shifted along their directions thus it describes where things are located in the image and thereby where edges will end up in the image.

Magnitude: The magnitude defines how large wave forms are, what grey levels are on either side of the edges. I.e. the magnitude gives us the different intensities in an image.

Therefore the phase information is more important than the magnitude when preserving the object in the image. This is shown in Figure 16.

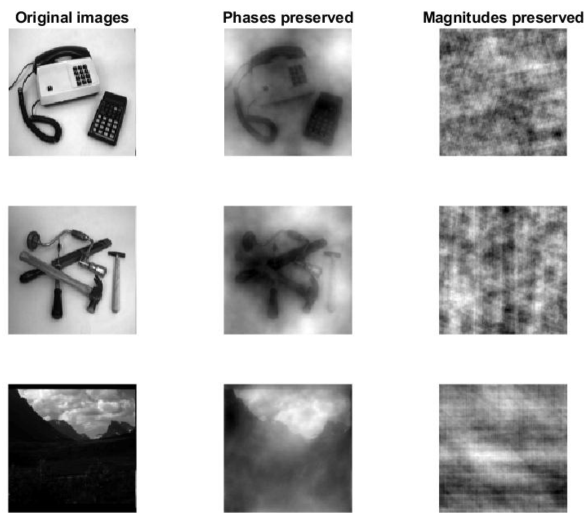


Figure 16: Rotating an image by 30°.

2 Gaussian convolution implemented via the fast Fourier transformation

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?

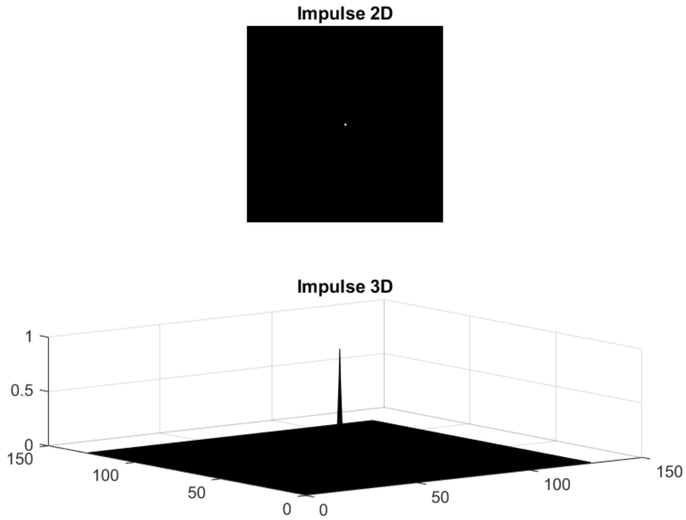


Figure 17: The input impulses in 2D and 3D.

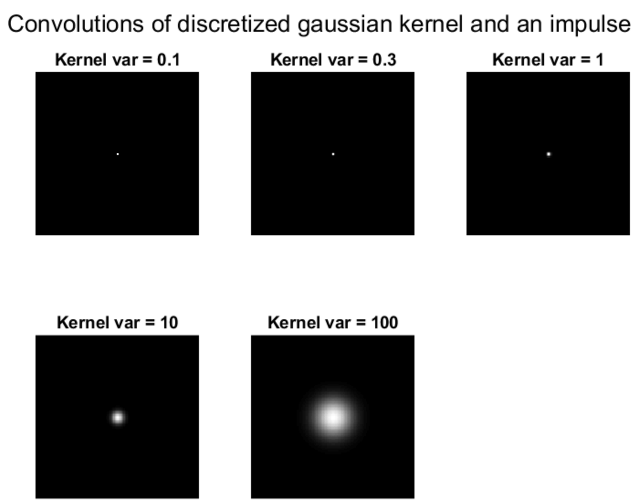


Figure 18: Impulse responses for different variances.

Table 1: Variance 0.1

0.0133	$1.9559 \cdot 10^-14$
$1.9559 \cdot 10^-14$	0.0133

Table 2: Variance 0.3

0.2811	$1.7596 \cdot 10^-15$
$1.7596 \cdot 10^-15$	0.2811

Table 3: Variance 1.0

1.0	$2.1444 \cdot 10^-15$
$2.1444 \cdot 10^-15$	1.0

Table 4: Variance 10

10	$1.0679 \cdot 10^-15$
$1.0679 \cdot 10^-15$	10

Table 5: Variance 100

100	$8.6991 \cdot 10^-17$
$8.6991 \cdot 10^-17$	100

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.

We can see that the higher the variance, the more similar the covariance matrix becomes to the ideal case.

When t is 0.3 and 0.1, i.e ¡1 the shape of the Gaussian is like an impulse spike and it is smaller than the size of a single pixel. Therefore, it does not resemble the ideal case until it is equal to or larger than 1.

A small impulse like a Gaussian in the spatial image domain corresponds to expansion into a Gaussian with a large variance in the Fourier domain.

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?

As the variance t is increased the images become increasingly blurry, resulting in more smoothing. This is reasonable since an expansion in the spatial domain results in suppression in the Fourier domain. Therefore, when the variance is increased in the spatial domain the result is a larger Gaussian in the spatial domain but a smaller one in the Fourier domain. Therefore, the cut-off frequency is increased, and more high frequencies are suppressed.

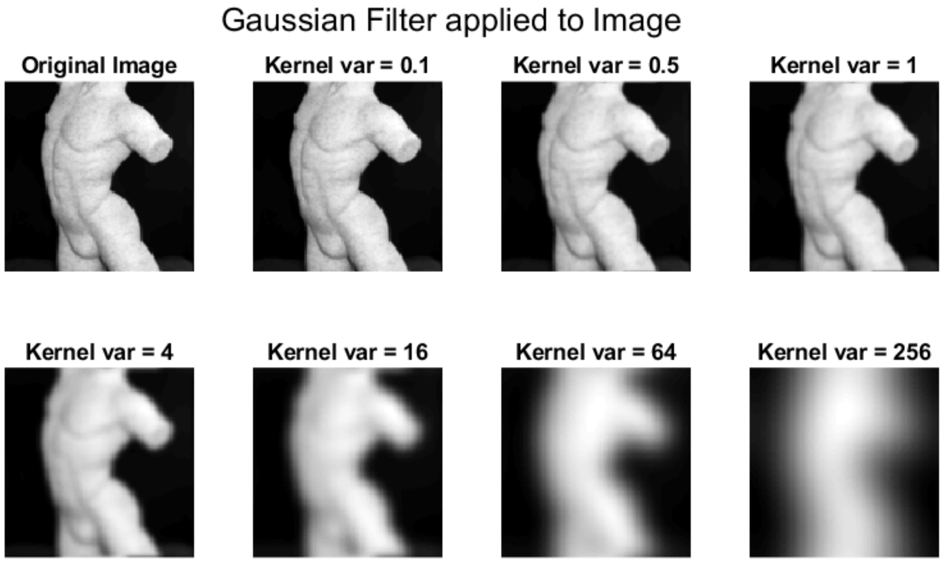


Figure 19: Filtering an image with a Gaussian with increasing variance.

3 Smoothing

3.1 Smoothing of noisy data

Here salt and pepper as well as Gaussian noise is added too an image and attempted to be removed via Gaussian smoothing, Median filtering and Ideal low-pass filtering. Different suitable values of the parameters of each filters are tried such as standard deviation for the Gaussian filter, window size for the median filter and cut-off frequency for the ideal low-pass filter.

In Figure 20 Salt and Pepper as well as Gaussian Noise has been added to an image.

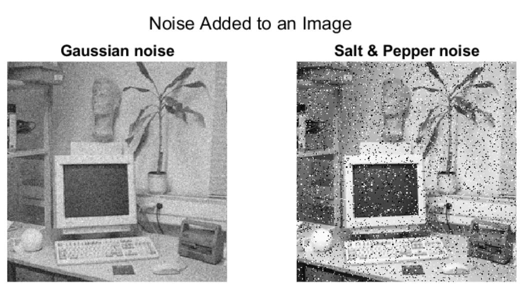


Figure 20: Gaussian noise added to an image to the left and Salt and Pepper noise added to the same image to the right.

3.1.1 Gaussian Smoothing

Here the noise is attempted to be removed via Gaussian smoothing with filters with different variances. These results can be found in Figure 21 and 22.

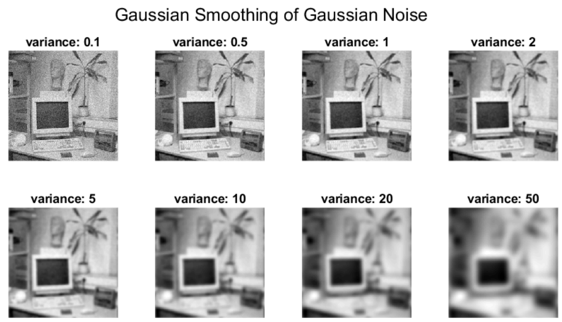


Figure 21: Gaussian smoothing of Gaussian noise with different variances.

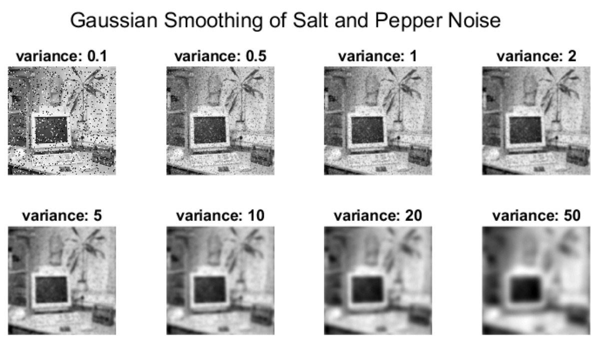


Figure 22: Gaussian smoothing of S&P noise with different variances.

3.1.2 Median Filtering

Here the noise is attempted to be removed via Median Filtering with filters with different window sizes. These results can be found in Figure 23 and 24.

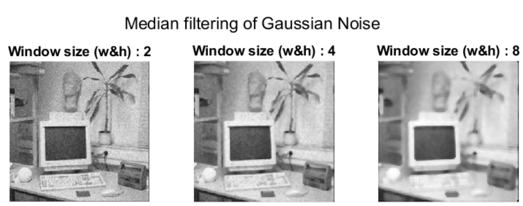


Figure 23: Median Filtering of Gaussian noise.

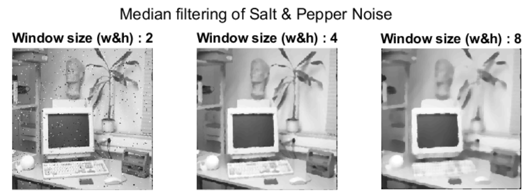


Figure 24: Median Filtering of S&P noise.

3.1.3 Ideal Low Pass Filtering

Here the noise is attempted to be removed via Ideal Low Pass Filtering with with different cut-off frequencies. These results can be found in Figure 25.

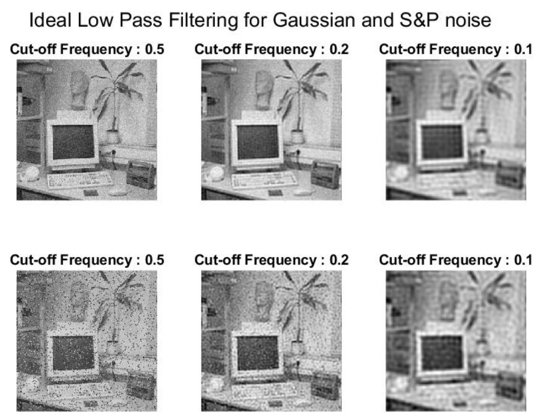


Figure 25: Ideal Low Pass Filtering of an image with different cut-off frequencies.

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).

Median Filter:

- Eliminates local extreme values (e.g. salt-and-pepper).
- Good at preserving edges.
- Creates painting-like images.

From the results in Figure 24 the negative painting effect on salt and pepper noise is clearly distinguishable. However, we can also see that it has effectively removed all the salt and pepper noise.In Figure 23 the same painting like effect occurs when the median filter is applied to the image with Gaussian noise. Also, the noise is not removed. However, the filter has preserved the image edges.

Gaussian Filter:

- Easy to interpret what it does since it is a Gaussian in both the Fourier and Spatial domain.
- Good at removing Gaussian noise.
- Smooths the edges of images, i.e. they become blurry.

In Figure 22 the Gaussian filter har integrated the salt and pepper noise into the picture instead of removing the noise. In Figure 21 we can see that when we increase the standard deviation the amount of smoothing increases, but the image becomes more blurry. Yet, the filter is able to remove the high frequency noise in the image. When the variance is between 1 and 10 the noise is removed but image gets blurry.

Ideal Los Pass Filter:

- Easy to understand in the frequency domain.
- The cut off results in a "box" in one domain which results in a sinc function in the other domain which ultimately results in a ringing effect.

In Figure 25 the ideal low pass filter it appears as if the filter can remove salt and pepper noise ans preserve edges when a low cut off frequency is used. However, it results in a ringing effect. In Figure 25 when applied to the Gaussian noise the ideal low pass filter has also resulted in a ringing effect in the image.

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

Median filters: Performs well on salt and pepper noise as it uses the median. Also, it is good at preserving edges and creates paint like images.

Gaussian smoothing: When the variance increases the blurring of the images increases. Also, it is good at removing Gaussian noise but it blurs image edges.

Ideal low pass: Results in a box like/oval shape which results in a sinc function in the Spatial domain thus we get a ringing effect.

Gaussian noise: It is hard to remove when we want to preserve edges.

Salt and Pepper noise: Can be effectively removed with a median filter.

3.2 Smoothing and sub sampling

Here the effects of smoothing when applied to images sampled at lower resolutions is studied.

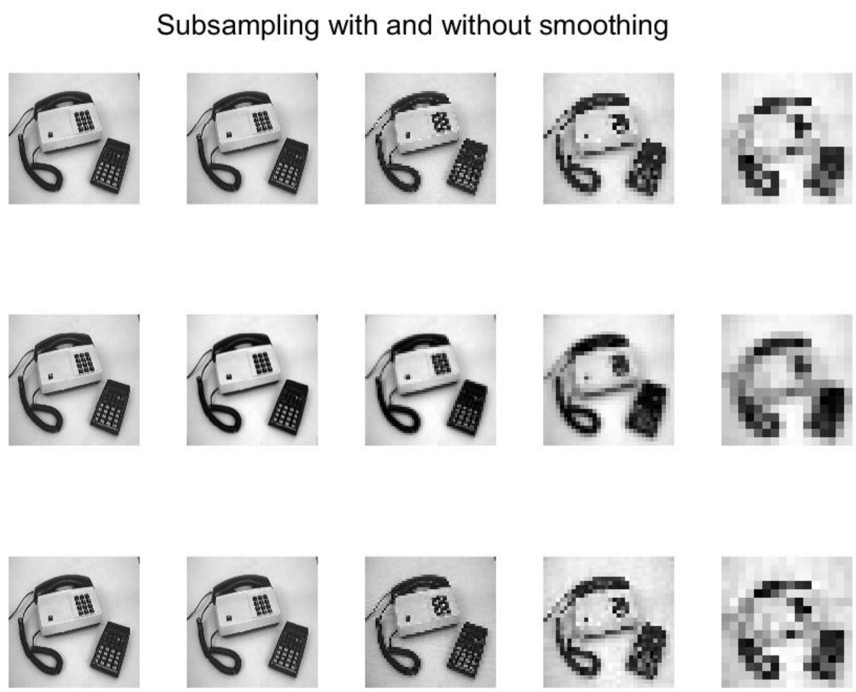


Figure 26: The first row contains the result from sub sampling an image to lower resolutions. On the second row the sub sampled images on the first row have been smoothed via a Gaussian Filter and on the third row via an Ideal Low Pass Filter.

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

We can see in the second row that smoothing with both the low pass filter and the Gaussian after sampling gives better results than if we do not use a filter. However, we notice ringing effects from the low pass filter. Using higher cut-off frequencies gave less ringing.

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

If we smooth the image after we sample from it the image becomes less distorted when we reconstruct it later from the subsamples. This is the result of the so-called Nyquist frequency. Since aliasing causing artifacts occur when we sample below the Nyquist frequency. Meaning that we cannot recover the wave form after sampling if we sample at a lower rate than this. Thus, if we apply a lowpass ideal or Gaussian filter first before sampling we remove high frequencies and therefore avoid artifacts.

The Nyqvist rate says that: If the signal is sampled at a rate equal or greater to than twice its highest frequency, the original signal can be completely recovered from its samples.

References

[1] Rafael C. Gonzalez and Richard E. Woods. *Digital image processing.* Prentice Hall, Upper Saddle River, N.J., 2008.