Lab 1: Filter Operations

Basic Operations

Q1: 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?

Observations:

- The same amplitude on every wave
- The direction(phase) of the wave depends on the position of the point in regards to the center in the Fourier domain
- The distance from the center in the Fourier domain determines the wavelength in the spatial domain
- The phase difference between the real and the imaginary part is , the real part is a cosine wave and the imaginary a sinus wave.

Q2: 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.

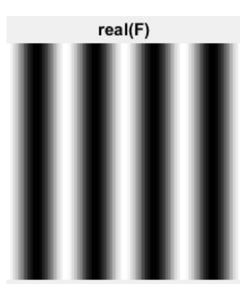
$$\mathrm{f(m,n)} = \frac{1}{\sqrt{NN}} \sum_{u=0}^{N\text{-}1} \sum_{v=0}^{N\text{-}1} \hat{f}(u,v) e^{2\pi i (\frac{mu}{N} + \frac{nv}{N})} =$$

$$= \frac{1}{N} \sum_{u=0}^{N\text{-}1} \sum_{v=0}^{N\text{-}1} \hat{f}(u,v) \left[\left(\cos(2\pi(\frac{mu}{N} + \frac{nv}{N}) + \mathrm{i} \sin(2\pi(\frac{mu}{N} + \frac{nv}{N})) \right] =$$

$$= \frac{1}{N} \left(cos(2\pi(\frac{mp + nq}{N}) + \mathrm{i} \sin(2\pi(\frac{mp + nq}{N})) \right) =$$

In the above formula we see the Euler formula being used.

Matlab figure.



Q3: How large is the amplitude? Write down the expression derived from Equation (4) in these notes. Complement the code (variable amplitude) accordingly.

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

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

$$= \left|\frac{1}{N} \sum_{m=0}^{N-1} \sum_{n=0}^{N-1} \hat{f}(u,v) \left[\left(\cos(-2\pi(\frac{mu}{N}+\frac{nv}{N}) + \mathrm{i}\sin(-2\pi(\frac{mu}{N}+\frac{nv}{N})\right)\right]\right| =$$

$$= \left|\frac{1}{N}\left(cos(-2\pi(\frac{mp+nq}{N}) + \mathrm{i}\sin(-2\pi(\frac{mp+nq}{N})\right)\right| =$$

$$= \left|\frac{1}{N}\left(cos(-2\pi(\frac{mp+nq}{N}) + \mathrm{i}\sin(-2\pi(\frac{mp+nq}{N})\right)\right| =$$

$$= \left|\frac{1}{N}\right|\left|\left(cos(-2\pi(\frac{mp+nq}{N}) + \mathrm{i}\sin(-2\pi(\frac{mp+nq}{N})\right)\right| =$$

$$= \frac{1}{N}\sqrt{\left(\cos -2\pi\left(\frac{mp+nq}{N}\right)\right)^2 + \left(i\sin -2\pi\left(\frac{mp+nq}{N}\right)\right)^2} = \frac{1}{N}$$

According to above we complement the amplitude in the code by 1/N

Q4: 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.

- Angular frequency $\omega = (\omega_1\ \omega_2)^T$

  ω_1 = angular frequency in *x* direction

  ω_2 = angular frequency in *y* direction
- Frequency

$$f = \frac{\omega}{2\pi}$$

- Wavelength

$$\lambda = \frac{2\pi}{\|\omega\|} = \frac{2\pi}{\sqrt{\omega_1^2+\omega_2^2}}$$

The frequencies in both directions $(f_1, f_2) = (p, q)$
The wavelength and angular frequencies are then calculated as

$$\lambda = \frac{2\pi}{\|\omega\|} = \frac{2\pi}{\sqrt{{\omega_1}^2 + {\omega_2}^2}}$$

$$\omega_1 = 2\pi f_1 = 2\pi \frac{u}{N}$$

$$\omega_2 = 2\pi f_2 = 2\pi \frac{v}{N}$$

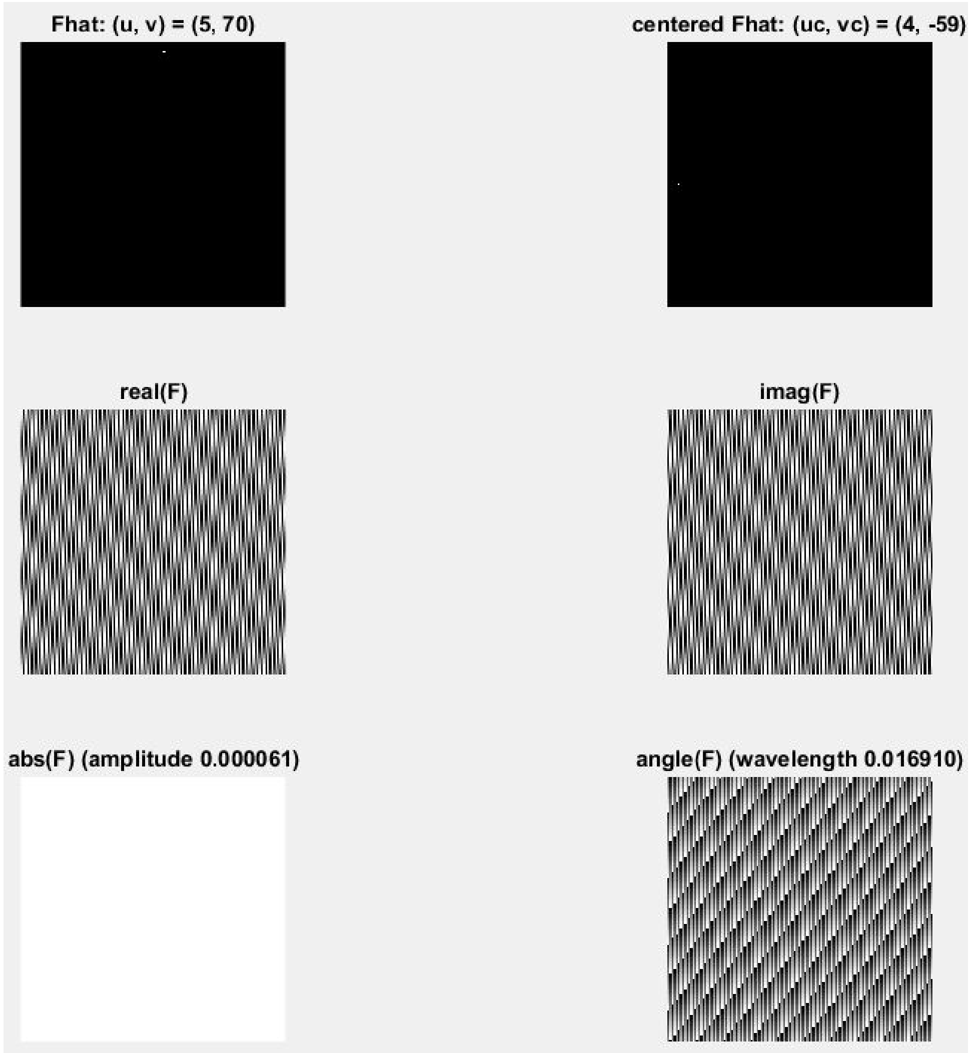
Fom above formulas we notice that the distance from the center determines the length of the sine wave in the spatial domain. To determine the direction of the sine wave we need to calculate the phase angle.

$$\text{Phase angle: } \phi(\omega) = \arg \hat{f}(\omega) = \tan^{-1} \frac{Im(\omega)}{Re(\omega)}$$

The phase angle is dependent on w_1 and w_2 which in our case is dependent on p and q.

Q5: 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!

The discrete Fourier transform of the image is periodic with interval N. In the fourier domain, If we go past the borders of the image in the u and v direction, the image is repeated and the point is placed accordingly on the other side of the image.



Q6: What is the purpose of the instructions following the question What is done by these instructions? in the code?

This changes the coordinates of (u,v) so that we can represent the same coordinate in our centered image. The code also handles the case of exceeding the image borders, if you go outside the image frame in the u or v direction, we are slunged back into the image by subtracting the image size. We now have a correct way to represent points in our shifted image.

Linearity

Q7: 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!

Let's consider the Fourier transform of the image F:

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

Since:

$$f(m,n)=\begin{cases}1 & if\ 57<m<73\\ 0 & otherwise\end{cases}$$

We can rewrite the previous equation as follows:

$$\hat{F}(u,v)=\frac{1}{N}\sum_{m=57}^{73}\sum_{n=0}^{N-1}e^{-2\pi i\left(\frac{mu}{N}+\frac{nv}{N}\right)}=$$

$$=\frac{1}{N}\sum_{m=57}^{73}\sum_{n=0}^{N-1}e^{-2\pi i\left(\frac{mu}{M}+\frac{nv}{N}\right)}=\frac{1}{N}\sum_{m=57}^{73}e^{-2\pi i\left(\frac{mu}{N}\right)}\sum_{n=0}^{N-1}e^{-2\pi i\left(\frac{nv}{N}\right)}$$

$$=\frac{1}{N}\sum_{m=57}^{73}\sum_{n=0}^{N-1}e^{-2\pi i\left(\frac{mu}{M}+\frac{nv}{N}\right)}=\frac{1}{N}\sum_{m=57}^{73}e^{-2\pi i\left(\frac{mu}{N}\right)}\sum_{n=0}^{N-1}e^{-2\pi i\left(\frac{nv}{N}\right)}$$

We introduce the kronecker delta function as:

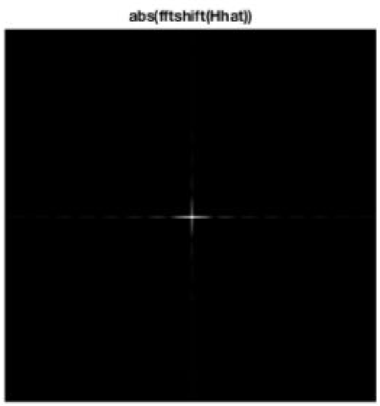
$$\delta(v)=\frac{1}{N}\sum_{n=0}^{N-1}e^{-2\pi i\left(\frac{nv}{N}\right)}=\begin{cases}1 & if\ v=0\\ 0 & if\ v\neq 0\end{cases}$$

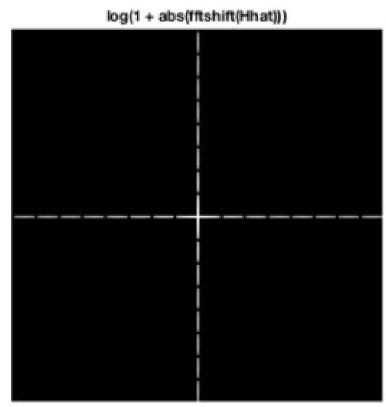
Thus the above equals

$$\hat{F}(u,v)=\sum_{m=57}^{73}e^{-2\pi i\left(\frac{mu}{N}\right)}\delta(v)$$

Q8: Why is the logarithm function applied?

The logarithm function is applied to enhance contrast in the frequency domain, the image becomes more visible. The dynamic range of an image can be compressed by replacing each pixel value with its logarithm. This has the effect that low intensity pixel values are enhanced. Due to the large dynamic range, we can only recognize the largest value in the center of the image. All remaining values appear as black on the screen. If we instead apply the logarithmic operator to the Fourier image smaller pixel values are enhanced and therefore the image shows significantly more details.





Q9: What conclusions can be drawn regarding linearity? From your observations can you derive a mathematical expression in the general case?

General conclusion;

$$\mathcal{F}\left[a\, \mathrm{f}_1(m,n) + b\, \mathrm{f}_2(m,n))\right] = a\, \hat{f}_1(u,v) + b\, \hat{f}_2(u,v)$$

$$a\, \mathrm{f}_1(m,n) + b\, \mathrm{f}_2(m,n)) = \mathcal{F}^{-1}\left[a\, \hat{f}_1(u,v) + b\, \hat{f}_2(u,v)\right]$$

You can add two functions (images) or rescale a function, either before or after computing the Fourier transform. It leads to the same result.

Multiplication

Q10: 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.

Convolution in the Fourier domain is equal to multiplication in the spatial domain.

$$\mathcal{F}(\mathrm{hf}) = \mathcal{F}(\mathrm{h}) * \mathcal{F}(\mathrm{f})$$

Scaling

Q11: 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.

Compression in regards to the center in the spatial domain is equal to expansion of the wavelength in Fourier domain. This is related to the location of points in regards to the center. Remember that points closer to the center have a larger wavelength and lower frequency.
Expanding increases distance from center and thus gives a shorter wavelength in the Fourier spectra.

Scaling property:

$$\mathscr{F}\{g(ct)\} = \frac{G(\frac{f}{c})}{|c|}$$

Rotation

Q12. What can be said about possible similarities and differences? Hint: think of the frequencies and how they are affected by the rotation.

In 2d the new point after rotation can be expressed by

$$\begin{bmatrix} \breve{x} \\ \breve{y} \end{bmatrix} = \begin{bmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}.$$

We can then express the old coordinates rotated back the same angle with the new coordinates.

$$\begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix} \begin{bmatrix} \breve{x} \\ \breve{y} \end{bmatrix} = \begin{bmatrix} \breve{x}\cos\theta - \breve{y}\sin\theta \\ \breve{x}\sin\theta + \breve{y}\cos\theta \end{bmatrix}.$$

If we perform the Fourier transform here we obtain:

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

$$= \frac{1}{N}\sum_{m=0}^{N-1}\sum_{n=0}^{N-1} f(m_1\cos\theta - n_1\sin\theta\,, m_1\sin\theta + n_1\cos\theta)e^{-2\pi i\left(\frac{(m_1\cos\theta - n_1\sin\theta)u}{N}+\frac{(m_1\sin\theta + n_1\cos\theta)v}{N}\right)} =$$

$$= \frac{1}{N}\sum_{m=0}^{N-1}\sum_{n=0}^{N-1} f(m_1\cos\theta - n_1\sin\theta\,, m_1\sin\theta + n_1\cos\theta)e^{-2\pi i\left(\frac{(u\cos\theta + v\sin\theta)m_1}{N}+\frac{(-u\sin\theta + v\cos\theta)n_1}{N}\right)} =$$

Q13: What information is contained in the phase and in the magnitude of the Fourier transform?

The magnitude of the FFT is how much energy there is in the sine waves used to build up your image. The phase is how those sine waves are positioned. In other words the phase is a measure of the displacement of the various sinusoids with respect to their origin. The apparent conclusion is that the phase is far more important than the magnitude for images.

Gaussian convolution implemented via FFT

Q14: : 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?

v1 =

0.0133	0.0000
0.0000	0.0133

v2 =

0.2811	0.0000
0.0000	0.2811

v3 =

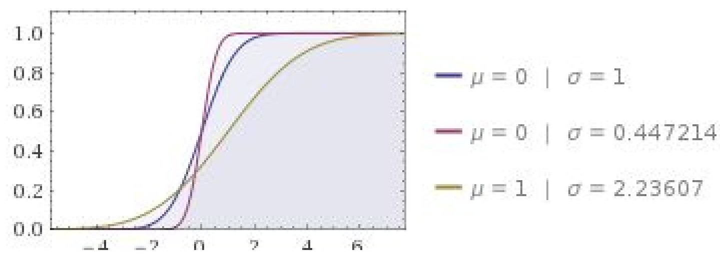
1.0000	0.0000
0.0000	1.0000

v4 =

10.0000   0.0000
0.0000   10.0000

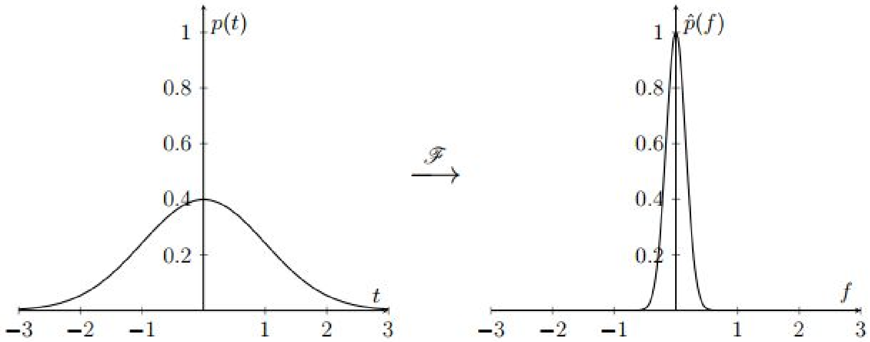
v5 =
100.0000   0.0000
0.0000 100.0000

Q15 : 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.



For variance <1 we get a shape looking less like a gaussian distribution and more like a square. The ideal filter has a shape of a rectangle. When we have values very close to 0 we can get a shape looking more like a rectangle and less like a gaussian distribution. That is why our estimates of the variance differ from the true value when t < 1.

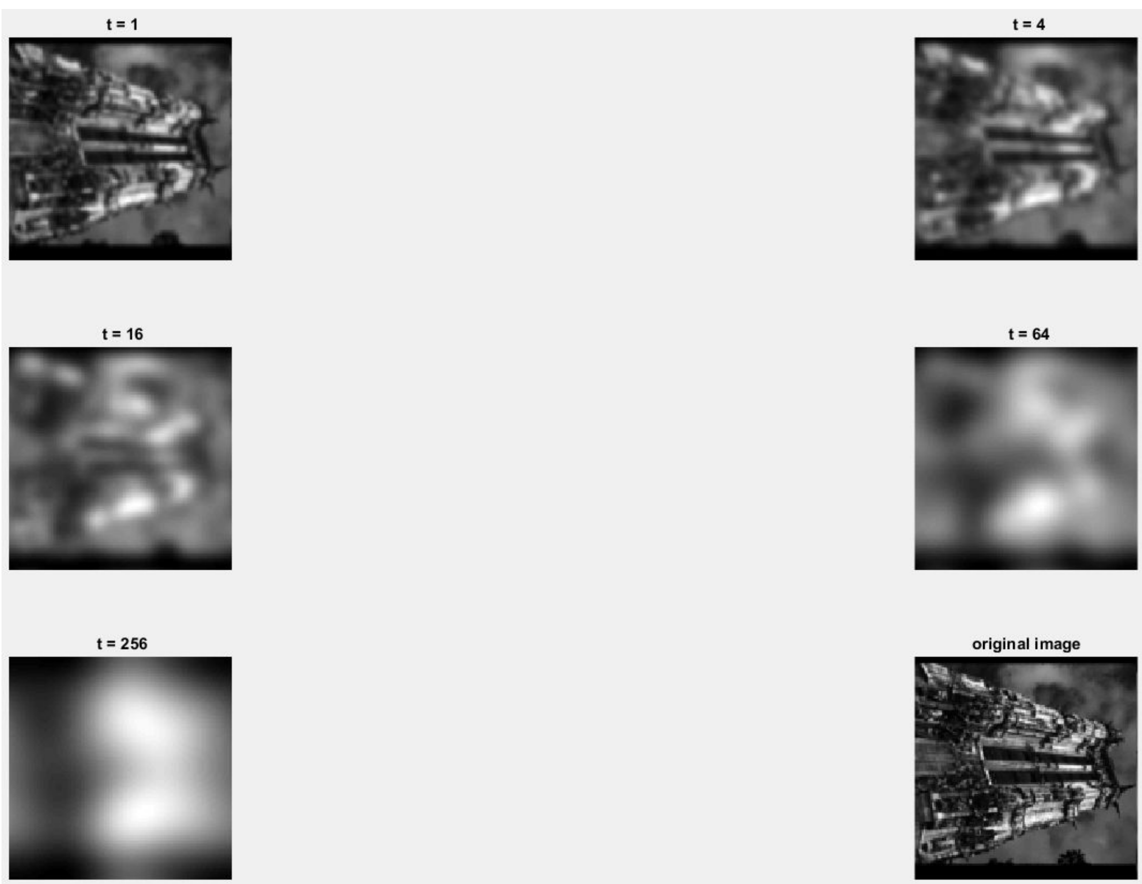
We remember that if we take the fourier transformation of a gaussian distribution we still get a gaussian distribution in the Fourier domain. With low values on variance(t) our filter in the spatial domain is more compressed to the center which means the opposite relation in the Fourier Domain. I.e the fourier transform of low variance filters will be more spread out from the center. In other words have a higher cut-off frequency That means they include more frequencies (higher frequencies). When multiplying with the fourier transform of the image more data is preserved. That is why our images are more blurry with high values on variance t, because more high are cut-off that are representing edges. And Vice versa it is less blurry for low values of t.



The Fourier transform of a Gaussian is just another Gaussian, stretched and scaled!

Q16: 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?

See Q15 for explanation and below for results.



Smoothing

A smooth image, e.g. a photograph often contains large areas of slowly varying intensity. These images have a dominant part of their energy concentrated to a few low-frequency components in the Fourier-domain. Edges in images imply rapid changes in the intensity, and will therefore be represented by the high-frequency components of the image's spectrum. This is important for the following questions.

Q17: 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).

Observations and results:

- Median filter

    Positive:
    - Removes low levels of noise effectively
    - Removes salt-and-pepper noise
    - Edge preservation
    - Extreme values are effectively removed (as in the sap-noise)

    Negative:
    - Computationally expensive because of the convolution especially in the case of larger images and window sizes

- Gaussian filter

    Positive:
    - Because it has a gaussian distribution the Fourier transform is still of a gaussian shape (just stretched or compressed)
    - Result in the smoothest transition from low frequencies to high frequencies in the mask
    - Good smoothing effect on images.

    Disadvantage:
    - Blurs the image for large variance values, information is lost, edges are not preserved because we remove higher frequencies that the edges represent in the frequency domain

- Ideal lowpass filter

Positive**:**

- You effectively remove all unwanted frequencies.(in theory)

Negative:

- Removes all high frequencies over the cut-off frequency, the mask has the shape of a box, therefore no edges are preserved for low values of the cut-off frequency
- Ringing artifacts appear in the image

Q18: What conclusions can you draw from comparing the results of the respective methods?

- Median filter

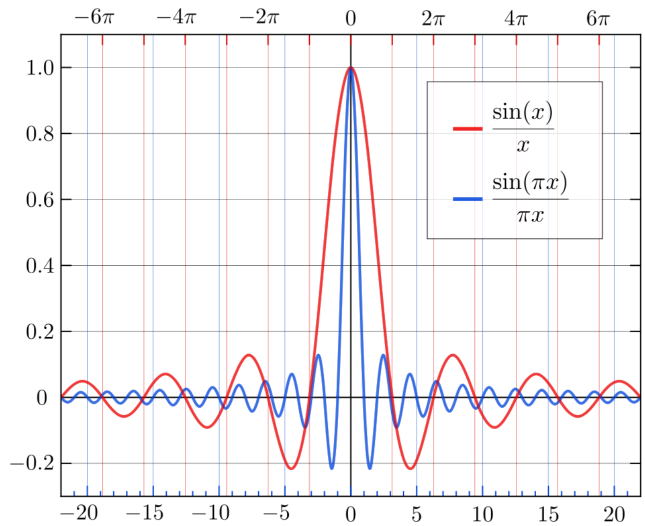
In the median filter the parameters define the size of the mask in which you calculate the median of the intensity, called the window. In the case of salt-and-pepper noise, the median filter is particularly effective because for every pixel the median intensity is calculated within the window area and replaces the pixel intensity value, removing any extreme values.
The filter is nonlinear and performs a convolution. Because the filter must process every entry in the signal, for large signals it requires large computational power.

- Gaussian filter

A linear filter , performs a multiplication in the fourier domain. If the variance increases it means more blur in the image. Fourier transform of the filter is still a Gaussian.

- Ideal lowpass filter

The Ideal lowpass filter is a brick-wall filter meaning it has an abrupt transition (i.e. perfectly rectangular), This however is not possible to achieve in reality but can be achieved by approximate implementations. The impulse response of such a filter has a sinc(x) shape.



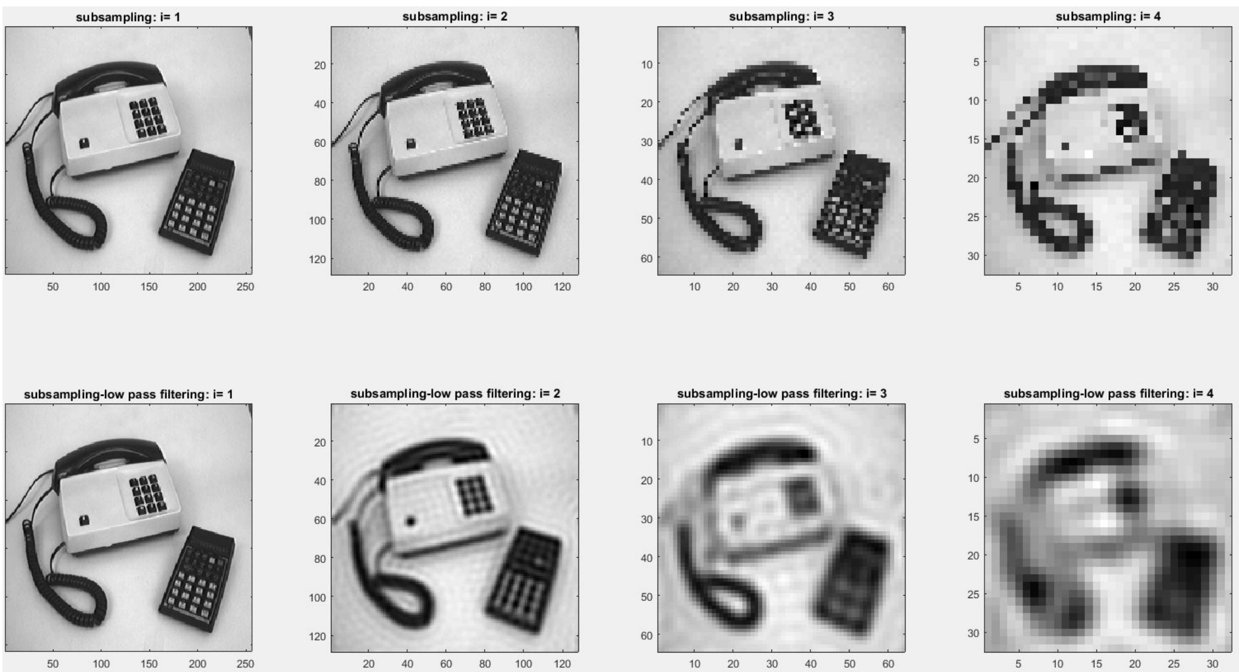
The function decays very slowly as you move away from the main lobe. This is not a desirable property; when you convolve a signal with a sinc, the effects of the slowly-decaying sidelobes will be apparent in the spatial-domain resulting in ringing artifacts.

Subsampling

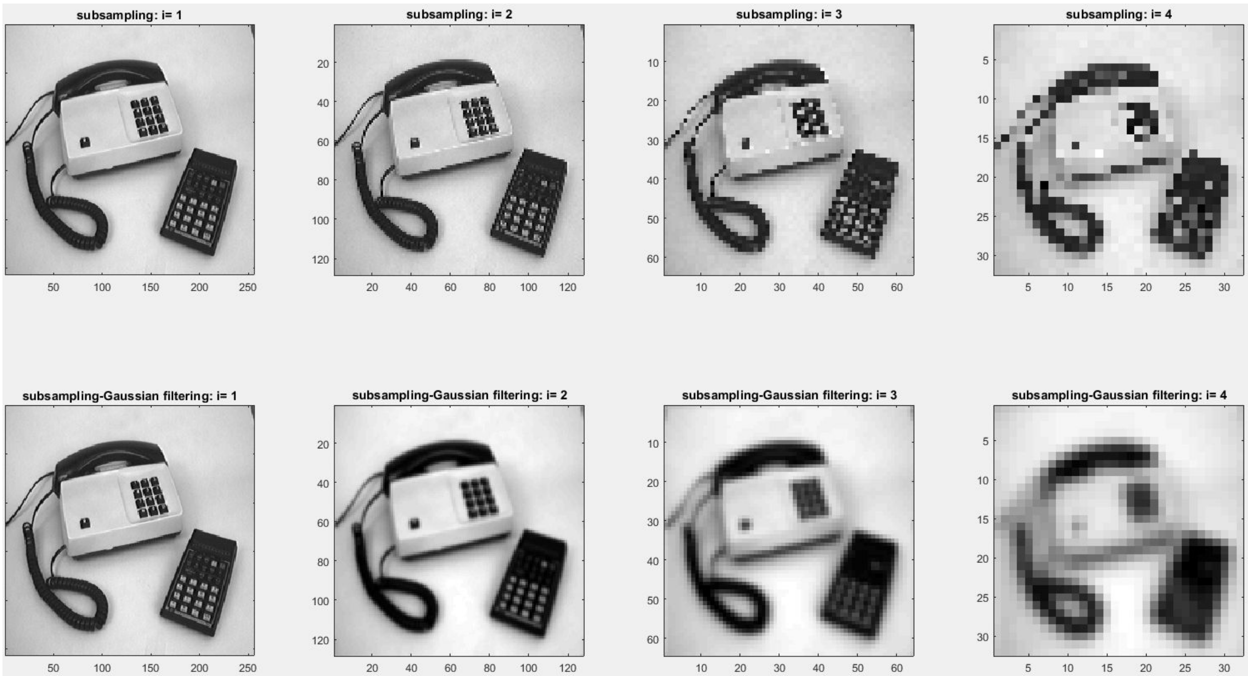
Q19: What effects do you observe when subsampling the original image and the smoothed variants? Illustrate both filters with the best results found for iteration i = 4.

We create a lowpass pyramid and gaussian pyramid by smoothing the image with the filter and then subsampling the smoothed image, with a factor of 2 along both axis. The resulting image is then subjected to the same procedure, we repeat the process 3 times. In this case we get worse looking image than if we smooth the image before subsampling.

- Lowpass pre-filtering



- Gaussian pre-filtering



Observations

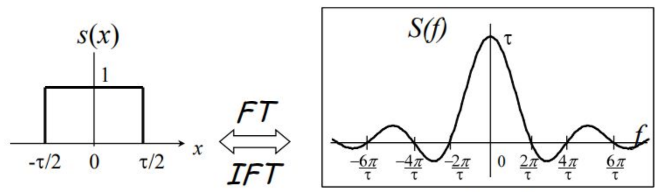
- For each iteration we see a image with poorer resolution.
  Worse in the case where we introduced smoothing after subsampling.(code template)
  Better in the case where we introduced smoothing before subsampling.
- Generally better result with gaussian filter
- We see ringing effects wíth the ideal lowpass filter and more information loss than the gaussian filter.
- Smoother images in both cases than with only subsampling without filtering

Q20: What conclusions can you draw regarding the effects of smoothing when combined with subsampling? Hint: think in terms of frequencies and side effects.

Each cycle of this process results in a smaller image with increased smoothing, but with decreased spatial sampling density (that is, decreased image resolution).
We use Prefiltering in order to avoid aliasing. Known as anti aliasing before sampling.

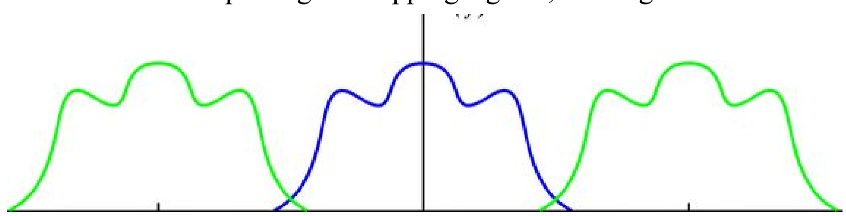
This is done by filtering (smoothing) of the signal to remove details above the sampling frequency. Ideally eliminates aliasing completely. Smoothing the image with a lowpass filter removes high frequencies so that the signal will be accurately represented by its samples. Pre-filtering is applied to ignore the frequencies above a limit we set.

- Prefiltering
  - Multiplication of the fourier transform with a box function corresponds to convolution of the signal with a sinc-function (sin(*x*)/*x*)



sinc-function

Repeating overlapping signals, aliasing



,

Frequencies above the Nyquist limit are 'folded back' corrupting the signal in the acceptable range.

The information in these frequencies is not correctly reconstructed.