TNM087 – Image Processing and Analysis
Lab 3 – Filtering in the frequency domain

TASK 2 - MATLAB code for filtering in the frequency domain

Write a code that performs filtering of an image in the frequency domain by a Gaussian lowpass and highpass filter using the given MATLAB template $FilterFreq.m$. As the input you have the original image and the cut-off frequency of the Gaussian lowpass filter and the outputs are the lowpass and highpass versions of the original image.

As discussed in the book and lecture notes for Chapter 4 (**FÖ** $\boxed{5}$ **and** $\boxed{6}$ **in the course**); **pages 96, 97 and the figure on page 98**, in order to do the filtering in the frequency domain you need to follow the following steps:

- Given the image f of size $M \times N$. Pad this image to another image of size $P \times Q$, with $P = 2M$ and $Q = 2N$ and call it f_p. **Important notice:** As shown in the figure on page 98 in the lecture notes, the input image should be located at the top-left corner of the zero-padded image (and not in the middle of it). Consider that when doing the zero-padding.
- Compute the DFT of f_p and use *fftshift* afterwards, call it F.
- Construct a real, symmetric Gaussian filter transfer function, $H(u, v)$ of size $P \times Q$ with center at $(\frac{P}{2}, \frac{Q}{2})$. **See the lecture notes** for Chapter 4 (**FÖ** $\boxed{5}$ **and** $\boxed{6}$ **in the course**) on **page 108** to learn how to construct such a filter transfer function. **Notice that** in the lecture notes, it was shown how to construct a Butterworth filter transfer function in MATLAB. Constructing the **Gaussian** filter transfer function will be done similarly, just use the appropriate equation discussed in the lecture notes on **page 104**.
- Form the product $G(u, v) = H(u, v)F(u, v)$ using elementwise multiplication.
- Obtain the filtered image (of size $P \times Q$) by computing the IDFT of $G(u, v)$:
$$g_p(x, y) = Real\{\Im^{-1}[ifftshift(G(u, v))]\}$$
- Obtain the final filtered result, $g(x, y)$, of the same size as the input image, by extracting the $M \times N$ region from the top, left quadrant of $g_p(x, y)$.

NOTICE: Observe that in this task, the size of the filter transfer function and the padded images, i.e. P and Q, are both even integers and therefore $P/2$ and $Q/2$ are both integers and won't cause any problem. Otherwise, you should use $floor(\frac{P}{2})$ and $floor(\frac{Q}{2})$ instead.

NOTICE: Since P and Q are even integers, the $ifftshift$ at the next to the last step above could be replaced by $fftshift$, without affecting the result. However, if they were odd integers, $fftshift$ wouldn't work.

At the end of this task, you are supposed to find the result of highpass filtering the image. The highpass transfer function, as discussed in the lectures, can easily be obtained by subtracting the lowpass filter transfer function from 1. This means that the Fourier transform of the highpass filtered image is obtained by:

$$F_{highpass} = H_{highpass}F_{original} = (1 - H_{lowpass})F_{original} = F_{original} - H_{lowpass}F_{original}$$

Taking the inverse Fourier transform of the above equation, gives,

$$f_{highpass} = f_{original} - f_{lowpass}$$

The above equation implies that the highpass version of the original image can **easily** be obtained by subtracting the lowpass filtered image from the original image.

Test your code on different images (for example *Einstein1.jpg*, *Einstein2.jpg* or *characterTestPattern.tif*) using different cutoff frequencies and look at the result of lowpass and highpass filtering the original image. Ask yourself, what happens if you increase the cutoff frequency (the second input argument in your function). Will the result be blurrier or less blurry? Test it and see whether or not you are correct. Hopefully, you are not surprised by the result, if you are, discuss it with your classmates or the teachers.