Introduction to Image and Video Processing
Lab 5: Image restoration

Spring 2022

1 Removal of DC component

Read a $M \times N$ image and find its FT, centered around $(M/2, N/2)$. Remove its DC component and display the filtered image. What do you observe? Why?

2 Periodic Noise Removal - Notch Filter

In this exercise you have to create additive periodic noise and add it to an image. The you have to create a notch filter to remove the periodic noise. We make the assumption that the frequency of the periodic noise is **known**.

- Create periodic noise with dimensions equal to the image size and frequency f_0 in x and/or y directions. The periodicity can be created by using a 2D sin or cos. Frequency variations can be horizontal, vertical and/or diagonal. You can multiply the noise magnitude with a constant of your choice.
- Read an image and add periodic noise. Display the image and its FT before and after the addition of noise.
- Consider that the periodic noise frequency (or frequencies, if different in x and y directions) is known. Create an Ideal notch filter around the noise frequency. Explain how you create it based on properties of the FT. For both filters, do the following:
  - Display the filter in 2D and 1D plots.
  - Apply the filter to the image and display the result. What do you observe?

3 Additive Noise Removal - Wiener Filter

In this exercise you will create an image affected only by additive noise, i.e. with degradation function $H(u, v) = 1$: $G(u, v) = F(u, v) + N(u, v)$, where $G(u, v)$ is the FT of the degraded image and $N(u, v)$ of the additive noise.

- Create additive random noise, of the same size as the image, using the inbuilt Matlab function randn.
- Add the noise to the image to create a noisy image $g(x, y)$ with DFT $G(u, v)$.
- Assume we known the noise-to-signal power spectrum $S_n(u, v)/S_f(u, v)$, where $S_n(u, v) = |N(u, v)^2|$, $S_f(u, v) = |F(u, v)^2|$ (but not the power spectrum of the noise or the power spectrum of the original signal). Use this information to create a Wiener filter to restore an approximation of the original image. *In your program, use your own knowledge of S_n and S_f to create the ratio S_n/S_f, and use the ratio in the code to create the Wiener filter*
- Apply the filter(s) to the image and display the result. What do you observe?

4 Image degradation and restoration

4.1 Image degradation with motion blur and additive noise

Create the following motion blurring degradation function. For $sinc(x) = sin(x)/x$ use "sinc" in Matlab:

$$H(u, v) = sinc(\alpha \cdot u + \beta \cdot v) \cdot \exp(-j\pi(\alpha \cdot u + \beta \cdot v)), \text{ for } \alpha = 0.13, \beta = 0$$

Here (u, v) need to be have values from -1 to 1, and their length needs to be equal to the corresponding dimensions of the image n_1 and n_2. You need to create them with the "meshgrid" function, and make sure the spacing from -1 to 1 produces n_1 points for u and n_2 points for v.

1. Read the $n_1 \times n_2$ image "win1" and apply motion blurr to it, by multiplying its FT with H.
2. Apply additive Gaussian noise *to the motion blurred image* using the function "imnoise" with $\mu = 0$, $\sigma^2 = 0.002$. *Note: in imnoise, the image needs to be converted to uint with "uint8". Use the magnitude of the image intensity (with "abs") in the argument of uint8.*
3. Compute the inverse FT and display the image degraded by motion blur and additive noise.

4.2 Image restoration with Wiener filter

1. Create Wiener filters for the degraded image as follows:
   (a) Assume the original x and degraded images x_n are both known. From these, find an approximation of the noise:
   $$n \sim x - x_n$$
   Use this to approximate the noise power spectrum, given by
   $$S_n(u, v) = |N(u, v)|^2, \text{ where } |N(u, v)| \text{ is the FT of the noise } n(x, y)$$
   (b) Find the power spectrum of the original image from
   $$S_f(u, v) = |F(u, v)|^2, \text{ where } |F(u, v)| \text{ is the FT of the image } f(x, y)$$
   (c) Find the Wiener filter for noisy, motion blurred images by:
   $$H_W(u, v) = \left[\frac{1}{H(u, v)} \frac{|H(u, v)|^2}{|H(u, v)|^2 + S_n(u, v)/S_f(u, v)}\right]$$
2. Apply the Wiener filter to the image degraded by motion blur and noise.
3. Display the restored image and its FT.

4.3 Image denoising

1. Read the original image original.jpg and its noisy version noisy.tif provided in this folder. Compare the noisy image with the original using MSE as a baseline.
2. Restore the image trying to minimize the MSE baseline as much as possible, using the following methods:
   (a) Gaussian filter
   (b) Median Filter
   (c) Wiener filter (you can use the inbuilt function from scipy).