E5ADSB Exercises 4 – Image processing

KPL, 2018-09-05

Exercise 1

Image smoothing with 2D low-pass filters

1. Load the test image “test_pattern_blurring_orig.tif”.
2. Try smoothing the image with mean-filters size 3x3, 5x5, 9x9, 15x15 and 35x35. Use `imfilter`.
3. How does the different filters affect the image?
4. Explain what happens at the border, and try out the different methods for dealing with borders as described in Marques p. 212 (“In Matlab”).
5. Try smoothing the image with the Gaussian-Blur filter size 3x3, 5x5, 9x9, 15x15 and 35x35. Use different values for σ in the range $[1; 10]$ and compare to the mean-filter. The function `gauss_mask.m` returns the Gaussian mask coefficients.
6. Load the image “noisyimage1.tif” and reduce the noise using smoothing.

Exercise 2

Image sharpening using “the Laplacian” and “unsharp masking”

1. Load the image “moon.tif”.
2. Implement the mask used to find the Laplacian in Marques chap. 10.4.1 p. 219 (try both, what’s the difference?).
3. Sharpen the moon-image by adding the Laplacian of the image of the image to the original image, like this: $g(x,y) = f(x,y) + c \cdot \nabla^2(x,y)$, where $c = \pm 1$ (Marques formula 10.16).
4. Show the result.
5. Try sharpening the moon-image using “unsharp masking”. The technique of unsharp masking is:
   a. Blur the original image
   b. Subtract the blurred image from the original image. The resulting difference is called the *mask:* $g_{mask}(x,y) = f(x,y) - f_{blurred}(x,y)$
   c. Add the mask to the original: $g(x,y) = f(x,y) + k \cdot g_{mask}(x,y)$, where k is a weighting factor. First try with $k = 1$.
6. Show the result. Try different values for k.

Exercise 3

Non-linear filtering – the Median Filter

1. Implement a 3x3 or 5x5 (or MxM) median filter as a function in Matlab.
2. Use the median filter to reduce the “salt & pepper noise” in the image “noisyimage2.tif”
3. Compare the median filter to the smoothing filter for reducing salt & pepper noise.