IMAGE PROCESSING (RRY025)

STUDIO EXERCISES

Lecture 3, Image Enhancement -2

On the windows system to use m-files written for the course you must in windows after starting MATLAB type

```
addpath \\mfil.me.chalmers.se\course-err041
```

If you are using the linux operating system after starting MATLAB you should type

```
addpath /chalmers/groups/course-err041
```

EX 1. Image smoothing example

Image smoothing can be used to detect structure in noisy images. Load the noisy image (**load('noisy.mat')**) which is an image an astronomer took of unknown object close to a nearby star, which is dominated by gaussian probability noise. Display with **imshow(noisy,[])**, is any structure detectable to the eye? Calculations suggest that the signal to noise on the object should be around 0.2, try to recover the object by image smoothing. Create a 2D gaussian filter function of size nn pixels and width ('sigma') of mm pixels using $\mathbf{g} = \mathbf{fspecial}('\mathbf{gaussian}', \mathbf{nn}, \mathbf{mm})$ and then filter with $\mathbf{filtim} = \mathbf{filter2}(\mathbf{g}, \mathbf{noisy})$. Try gaussian filters of various sizes, what is the minimum size to detect anything? What is the best gaussian size, what sets the limit for the maximum size its useful to use?

EX 2. Unsharp masking - High pass imaging

Load and convert to double precision the flower image **flower = double(imread('flower.tif')**. Display using **imshow**, using the mouse to resize the window to make the image bigger. Try to sharpen this image by first smoothing it and then removing that smoothed image from the original ('unsharp masking'). Use again **fspecial('average',3)** to produce an 3x3 smoothing filter. Next make a smoothed version using **filter2** and display that. Finally remove the smoothed version from the original and display it. This image is a so called 'high pass' image, because it only contains high spatial frequencies. Which regions have large amplitude in this image and why? What is the average pixel value in this image?

EX 3. High Boost Filtering

Although the above sharpened image highlight regions where the image changes we lose general information about the object (its hard to recognise as a flower). If our aim is to make simply a sharpened version of the original to publish in a magazine our best strategy is to combine with the original ('high boost filtering').Create (original image) + (A x High pass image)) where A is an adjustable parameter (start with A=1). Display this final image alongside the original image, is it significantly improved?. Vary the A parameter to find best results.

EX 4. High pass filtering - directly computed

A high pass image can be made in one go without the need to remove a smoothed version from the original if we choose a suitable filter (equal to a delta function minus a smoothing function) to correlate with.

A doctor is checking an X-ray to see if there is any fractures in the spine of a patient - and she uses high pass filtering to enhance her image. High pass filtering can be done in one step by using a suitable convolving function which is a combination of a delta function minus an averaging function.

In matlab load **load('spine4.mat')**, display with **imshow(spine4,** []**)**, enter a highpass filter **hp = (1./9).*** [**-1 -1 -1; -1 8 -1; -1 -1 -1**], convolve using command **hpspine =filter2(hp,spine4** and display the result (type **zoom** and click on image to look closer at different parts of image). Do you see any small fractures near the centre of the spine?.

Apply a similar highpass filter to the camera image (**cam = imread('cameraman.tif');** Experiment using the **fspecial** command to create some other Laplacian or highboost filter.