Edge Detection

Exercise 6 in advanced image analysis

Henrik Aanæs and Karl Skoglund

23rd March 2004

The aim of this exercise is to give you an introductory practical experience with taking the derivative of images, taking scale space theory into consideration.

Task1: Make a function that takes an image and a scale parameter t as input and returns extracted edges according to: Tony Lindeberg, "Scale-Space: A framework for handling image structures at multiple scales". Use Equation (14)

Task 2: Extract edges from the image found in `CircIm.mat`.

Task3: Extract edges from the image found in `Pap.mat` at different scales. Explain.

Hints

It is advised that you derive the Guassian kernel and its derivatives *analytically* and use this for your implementation. Note that the Gaussian is separable, i.e. the 2D kernel can be formed from 2 1D kernels. The formula for the 1D gaussian is:

$$\frac{1}{\sqrt{2\pi t}}\exp(-\frac{x^2}{2t})$$

Where $t = \sigma^2$ is the variance.

An example of how to convolve the image with a gaussian is:

```
im=filter2(Gauss,Pap,'same');
```

Note that the size of the kernel should reflect the size of the scale parameter t. I.e. the larger the t the larger the kernel should be. A rule of thumb is that the kernel *radius* should equal $3\sqrt{t}$. Why?

A possible way of detecting zero-crossings of `Lvv` is:

```
LvvP=(Lvv>0);
Lvv0=xor(LvvP,circshift(LvvP,[0,1])) | xor(LvvP,circshift(LvvP,[1,0]));
```

Why? Note that this does not find the zero-crossings with sub-pixel accuracy.