Problem Set #3

Due on Thursday, March 5 2009

General Instructions

Images are available at `http://zoo.cs.yale.edu/classes/cs475/image.html`.

If you would like to submit your answers electronically, please leave a copy in you zoo `cs475` directory. Create a subdirectory called `hw3` and name the file `hw3-yourname.pdf` or `hw3-yourname.doc`. If you would prefer, images can be saved into the same folder. *If you do this you must use a sane naming scheme*

1 Edge Detection

(a) Define what you believe to be a "good edge".

(b) Compute the gradient magnitude of `Paulina.tiff` as in the first assignment. Suppose the values are in the variable `gmag`. Display `gmag` > `thresh`. Can you find a value of `thresh` so that all the "good edges" show? Can you find one where no "bad edges" show? Is it possible to detect all "good edges" while suppressing "bad edges"? Show your best example.

(c) Repeat (b) for the canny edge detector. Use `edge(image, 'canny', thresh)`. Show your best example.

(d) Repeat (b) with your DoG filter from problem set 1. Look for the zero crossings of your filter. Use `edge(image 'zerocross',filter, threshold);`

(e) Which edge detector did best? Did any of them successfully detect all "good" edges without finding "bad" edges?

2 Blurring

(a) Look at the figures below. Run the best edge detector from part 1 on the clean images. Can a threshold be chosen such that a clean "good" edge can be found? Show your results.

(b) Repeat part (a) for the noisy images.

(c) Blur the noisy images by convolving them with a Gaussian. Use `fspecial('gaussian',hsize,sigma);` or your own filter. Find a sigma filter such that the edge detector no longer detects "bad" edges. Does the filter still detect "good" edges? Show your best combination of sigma and threshold for each blurry image.

(d) For each noisy image for what sigma does the noise effectively disappear? When do the circles connect according to your edge filter? When do the internal edges disappear?

(e) Does your edge detector meet your criteria for a "good" edge detector in this case? Why or why not?

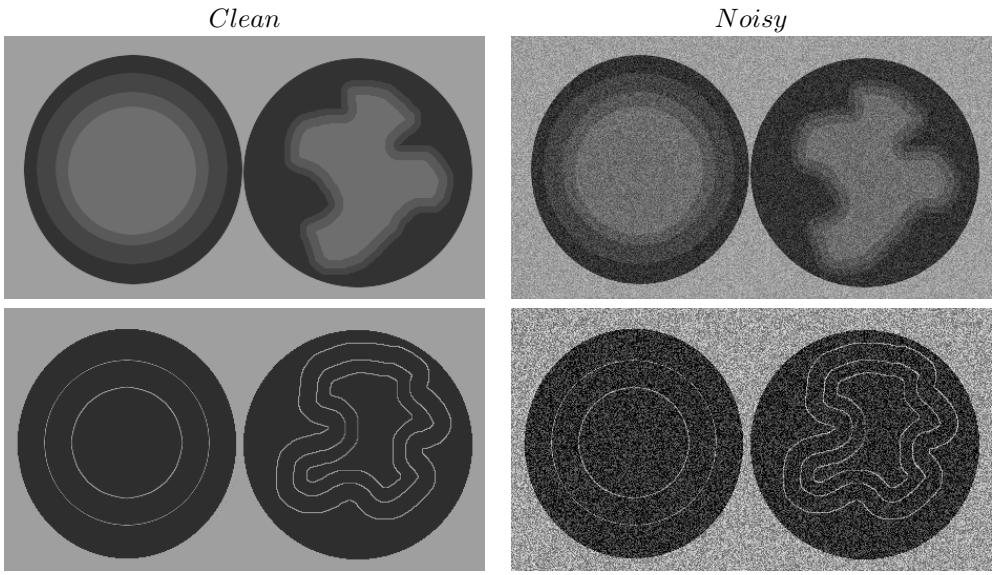

Figure 1: Images for problem 2

3 Orientation Sensitivity

(a) Build an oriented edge detection filter. It should respond positively to vertical edges when high brightness is to the right of low brightness. Convolve your filter with the clean circle images. Over what range of orientations is the response positive?

(b) Is it possible to decrease the range of orientations that your filter responds to *without* using a threshold? How?

HINT: You can change the size of your filter if it helps. Should the filter be square?

(c) Build two more vertical edge filters each tuned more precisely in orientation to a vertical edge. Show the results.

(d) Are there any trade-offs for your scheme? In other words, by increasing sensitivity to orientation, are you decreasing sensitivity to any other important information? How well will a very precise orientation filter respond to the vertical part of a small circle? How will the response change as you vary the radius?