

EE 528 - Digital Image Processing: Project 2

Monte Carlo Analysis of a Measurement System and Neighborhood Operators

Monte Carlo Analysis is a simple and useful technique for studying the accuracy of a measurement system. Monte Carlo is very helpful when considering "what if..." scenarios during system design. This is especially so when processing includes nonlinear operations, which make it difficult to analyze via linear techniques (as with the median filtering, used here).

In this project we will work with a system that measures the orientation of a line. You will generate your own input images. These will contain 3 bright spots (light bulbs on the side of a space ship) and your program should determine the best-fit line through these points, via least-squares, and then report the orientation (of the space ship) in degrees. Your images will also contain a band of impulse noise. Use median filtering to remove the impulse noise, to improve accuracy.

The Monte Carlo method estimates accuracy via simulations. Here, your program will generate a simulated image and then process it. Because the input is simulated you will know what the exact orientation measurement should be. You can then compare it against your measured value. These values will differ because of noise and other limitations (like finite resolution of images). Generating randomized input conditions and averaging over many trials can yield reasonable estimates of accuracy.

This project also involves neighborhood processing, where a small group of pixels in an input image are used to compute each pixel in the output image. Here, you will use neighborhood processing to remove shot noise. Many examples of real-time systems exist that use neighborhood processing. It is popular for hardware-based implementations because the image neighborhood can be accessed using delay buffers. High performance hardware platforms exist that perform neighborhood operations in real-time, for example with an 8x8 neighborhood and 512x512 images at 30Hz frame rates.

Project Requirements:

Write a program that performs Monte Carlo Analysis on the line-orientation measurement described above. [Lectures in class will elaborate on this process]. Estimate the accuracy of the orientation measurement (in degrees). Define your own quantitative measure of accuracy. Also, after your program finds the best-fit line, it should draw that line in the image (use a gray level of 128, to make the line distinguishable from the white spots and black background). You may restrict the range of orientation for the line to +/- 30 degrees. Use the following expressions to determine the radius of your bulbs and their separation:

$$\text{Radius} = 10 + F/2.6 \text{ (pixels)} \qquad \text{Separation} = 60 + L/1.3 \text{ (pixels)}$$

Where F and L are the indices of the first letter of your first and last name. (A==0,B==1,C==2...). Everyone should use a height for the horizontal band of noise of 25 pixels and a probability of occurrence of 0.02.

Suggestion for creative extension to project, use Monte Carlo Analysis to find an optimum value for a system parameter such as bulb radius (in pixels). Search through a range of possible bulb radii and report the best.

Report

- 1) A listing of your program, with comments, and file header giving your name, course and project number.
- 2) Brief report including:
 - a) Your name, the course title, and number of this project.
 - b) Example images.
 - c) Define your metric of accuracy and report your accuracy estimate. Also report conditions used to determine this estimate: bulb radius and separation, and the parameters associated with the noise.
 - d) Brief discussion of procedure and comments on results.
- 3) Answers to questions, below.
- 4) An executable version of your program, with input file(s) on a floppy or CD.

Questions

- 1) Median filters are useful neighborhood operations that remove impulse noise. These can be implemented by sorting. Explain why the worst case amount of effort needed for a bubble sort requires $N(N-1)/2$ pair-wise comparisons, for a length-N list.
- 2) Find the cutoff frequency (in radians/pixel) for the following highpass filter. This filter is used to approximate a 1st derivative, sometimes for edge detection. This filter is referred to as a 'simple difference'.
$$h(x) = \{ 0.5, -0.5 \}, \quad x = 0, 1$$
What is the magnitude of the frequency response of an ideal differentiator? Sketch it and sketch $|H(w)|$ on the same graph. What spatial period (a distance in pixels) corresponds to the cutoff frequency? If the filter is used for edge detection, and if we desire a response in the passband for an edge, then what is the maximum width of edge (measured from light to dark) that can be found with this filter?
- 3) N data points (x,y) are obtained from the surface of a round welding bead. The center location of the bead (x, horizontal) needs to be found in order to guide a robot using machine vision. Derive a least-squares formulation, that can be used to solve for an optimal quadratic fit of the data. Your model of height (y) versus horizontal position (x), should be:
$$y = a * x^2 + b * x + c$$
Given N data points, find simplified expressions for the optimal parameters (a,b,c). Then solve for the center coordinate, x, (horizontal) using your expressions for a,b,c.