CS 558: Homework Assignment 2 - Line Detection Due:
OCTOBER 26TH, 5:59pm

Enrique Dunn
Department of Computer Science
Stevens Institute of Technology
edunn@stevens.edu

Submission Format. Electronic submission on Canvas is mandatory. Submit a zip file containing:

- a pdf file describing your work, showing resulting images and a brief explanation of the implementation.
- the (runnable) code,
- the output images.

Problem 1: Pre-processing. Download the image provided on the course canvas website. You must implement two methods for detecting lines. Before extracting the lines, you need to detect potential points on them. These will be found as local maxima of the Hessian detector, implemented as in the 4^{th} set of notes. Apply a Gaussian filter first and use the Sobel filters as derivative operators. Threshold the determinant of the Hessian and then apply non-maximum suppression in 3×3 neighborhoods.

Ignore pixels for which any of the filters falls even partially out of the image boundaries.

Problem 2: RANSAC. Apply RANSAC on the points detected above to find four lines with strong support. Since it is intractable to guarantee that you found the four lines with the most support, use your best judgement to determine that the outputs are correct. One possible implementation is to run RANSAC until it finds a line with a sufficient number of inliers. Make sure that the same inliers are not used again.

To determine a line model hypothesis select two points at random and compute the parameters of the line joining them.

2a. Report your choices for inlier thresholds, total number of iterations and confidence values. **2b.** Plot the line segments in the image by connecting the two extreme inliers of each line. Also plot the inliers as 3×3 squares.

Problem 3: Hough Transform. Apply a Hough transform to detect the four lines with the strongest support using the polar parameterization of the line (not the point coordinates)
3a) Plot the results as above in (2b). Include also an image of the resulting 2D histograms used for voting, where black color corresponds to zero votes and white color corresponds to the maximum number of votes.
3b) Reduce the Bin size in each dimension by half w.r.t. to the value used in (3a) and report results.
3c) Double the Bin size in each dimension w.r.t. to the value used in (3a) and report results.

Note that the last two problems are independent.

Requirements and notes.

- You can use any programming language. You may have to explain the homework to me in person, if I am not familiar with your choice.
- You are allowed to use image reading and writing functions, as well as plotting functions, but you are not allowed to use filtering, edge detection or other image processing functions.
- Screenshots of the input with the lines and squares superimposed are acceptable.
- If you cannot implement the Hessian detector, using the local maxima of the gradient magnitude will allow you to continue, but will not receive maximum credit. (Non-maximum suppression should be in all directions here, as with the Hessian detector.)
- In Problem 1, the main parameter to be adjusted is the threshold on the determinant of the Hessian. The standard deviation of the Gaussian filter is not that important.
- In Problem 2, the main parameters are the required number of inliers and the distance threshold that determines which points are inliers given a line equation.
- In Problem 3, the main parameters are the dimensions of the bins of the accumulator.