CSE344: Computer Vision
Homework - 2

Nitika Saran
#2014068

I used python and OpenCV for this assignment.

Q1. a) The two images blurred using 11 x 11 gaussian filters with varying sigma are shown below:

road1.png:

Sigma = 1

Sigma = 3

Sigma = 7

road2.png:

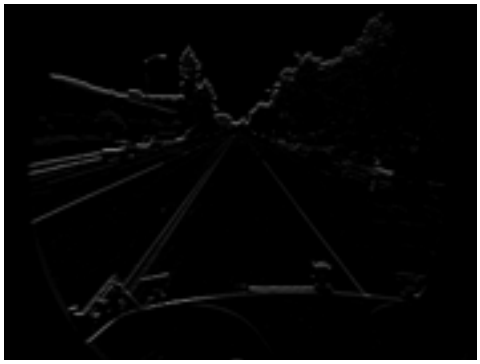
Sigma = 1

Sigma = 3

Sigma = 7

b) The sobel derivatives for the two images with varying kernel sizes are shown below. Each series is in the order - x gradient, y gradient, gradient magnitude.

road1; kernel size = 3



road2; kernel size = 3

Results for other kernel sizes are include in the submission folder but not in this report.

c) To sharpen both images in their grayscale format, I multiplied them by scalar value 1.6 and subtracted 0.6 * gaussian blurred version of them (11x11 kernel with sigma 7).
Results below show that this brings out the light-coloured lanes more and distinguishes them from the rest of the road.

road1:

road2:

Q2. a) To highlight the lanes, I used a simple contrast enhancement technique wherein I normalised all pixel values (dividing them by max intensity 255); squared these normalised values and then multiplied them by 255. This way, the pixels that were dark become much darker, and those that were light become only slightly darker, making the yellow and white lanes stand out more.

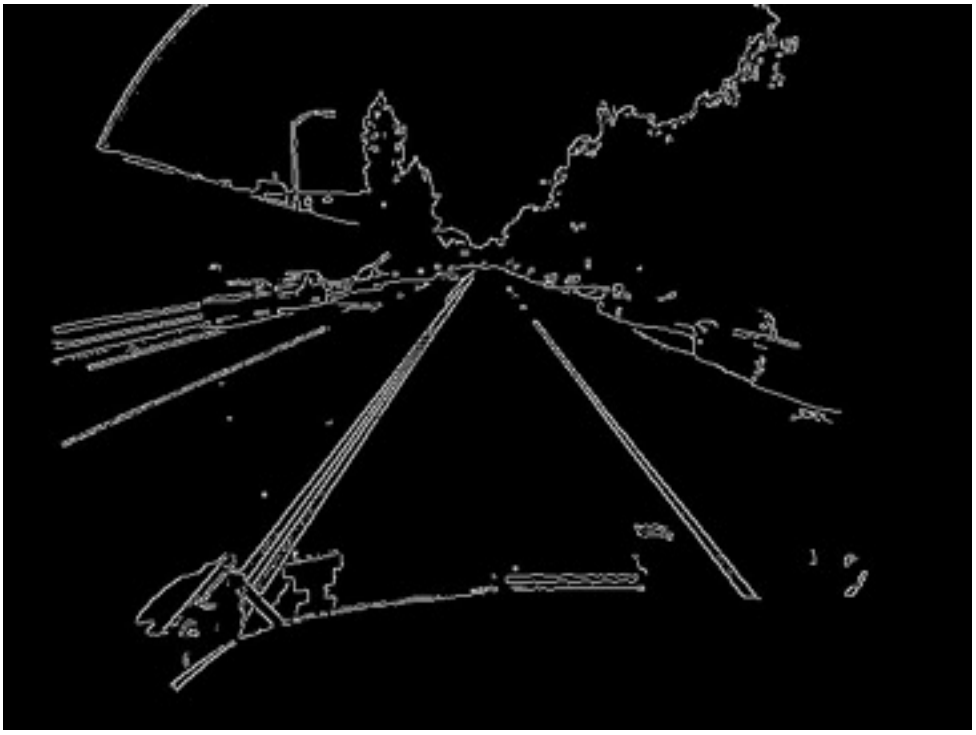
A comparison is shown below: (left=original; right=enhanced)

road1:

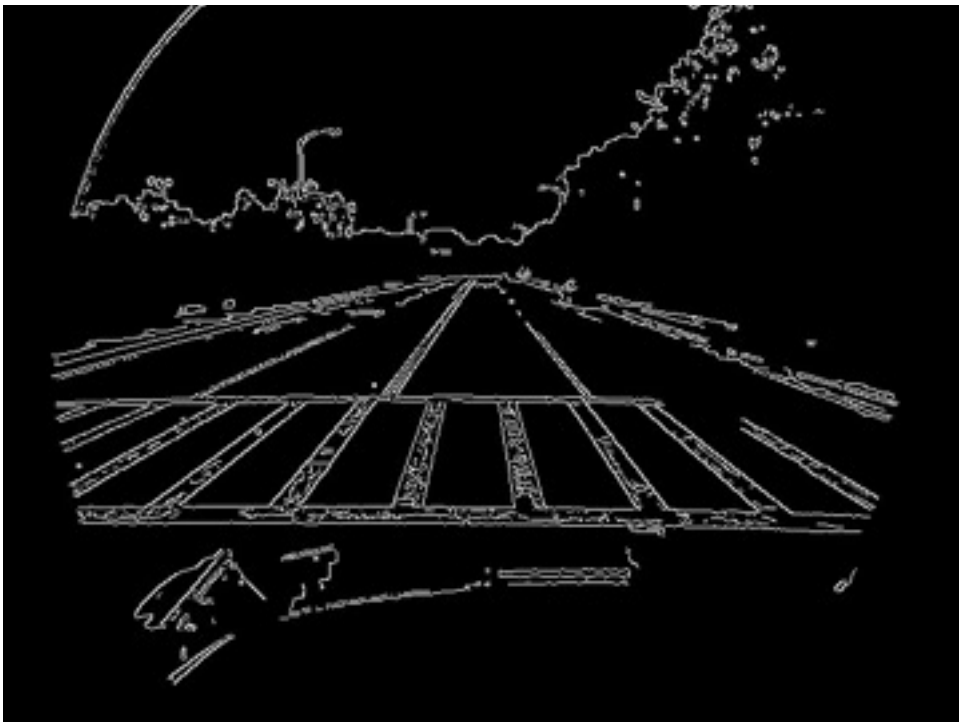

road2:


b) I chose the two canny thresholds for hysteresis edge detection as lower = 60, and upper = 160. The recommended ratio of upper to lower is between 2:1 and 3:1. I set the upper threshold slightly higher than default value to remove unnecessary clutter from tree leaves etc. These edges were not required for lane detection. The lower threshold was a little more than one-third of the upper one to detect lines well. The two canny images are shown below.

road1.png:



road2.png:

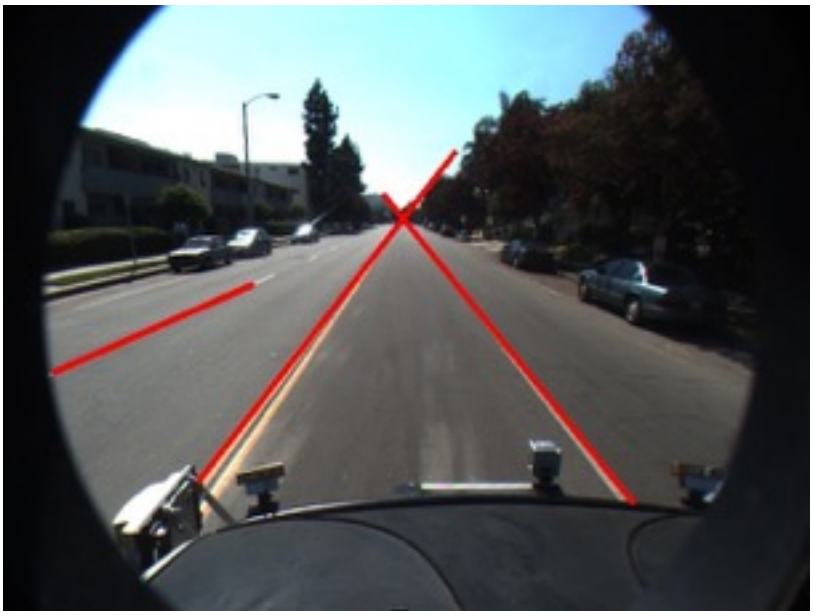


c) I chose the hough parameters as follows:

Distance resolution of the accumulator = 2 pixels
Angle resolution of the accumulator = 1 degree (pi/180 radians)
Threshold for votes- lines with votes >threshold get selected = 180 votes
Minimum line length- line segments shorter than that are rejected = 120
maxLineGap – Maximum allowed gap between points on the same line to link them = 17

These parameters detected multiple lines per lane, and I grouped lines with an angle difference of less than ~7.5 degrees. I also allowed a maximum of 2-3 lanes per side (left and right lanes) with preference to lines with higher number of votes. This same process and set of parameters were used for both images. Results as follows:

road1.png



road2.png

Q3. I set the parameters of EDISON to the following:

SpatialBandwidth : Specifies size of spatial search window during the mean shift computation = 23
RangeBandwidth: bandwidth of the search window in the range subspace = 10
MinimumRegionArea: minimum allowable region area (in pixels) in segmented image = 200 pixels

This gave me the following segmented image with each product having 4-6 components: