Matthew England
2/15/2016

Computer Vision CS8690/ECE8690
Assignment 1: Linear Hough Transform

Introduction:

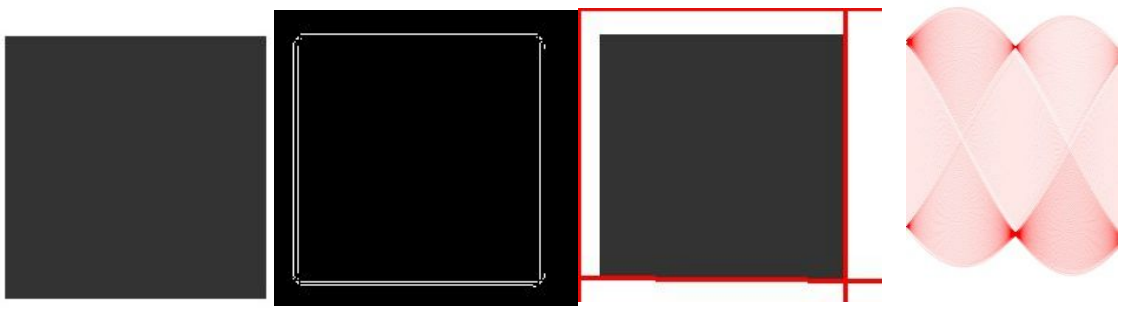
The hough transform allows researchers to find linear line segments in an image. The hough transform can be a feature extractor to help detect, roads, walls, and straight contours in images. A advantage of using hough transform is the ability to transform into a polar coordinate space (a.ka. Hough Space). This allows for the bucketing of overlapping polar coordinate lines to determine line segments.

Implementation:

I used the provided pseudo code look into other methods of searching the image space for lines within the confines of the hough transform. I convert the incoming image into grayscale to simply edge detection. I blur the image using a gaussian blur, then I apply the canny edge detector over the image. After that I begin to find points in the image. First I must convert every point into polar coordinates using the function: r = x * cos(theta) + y * sin(theta). Using the found polar coordinates, you store them into a grid histogram (i.e. accumulator matrix). After traversing the image, you then must find the local maximums of which a max window filtering over the accumulator matrix. If you find that your current point is not a local maximum, you must skip that point and move onto the next polar coordinate. After finding all the local maximums, I begin to add them together to create lines. I then convert the polar coordinates back into cartesian coordinates (2D image coordinates).

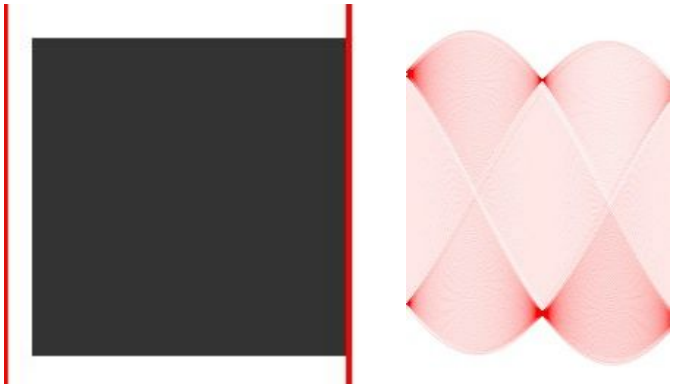
Results:

Square Image (Threshold = 25):

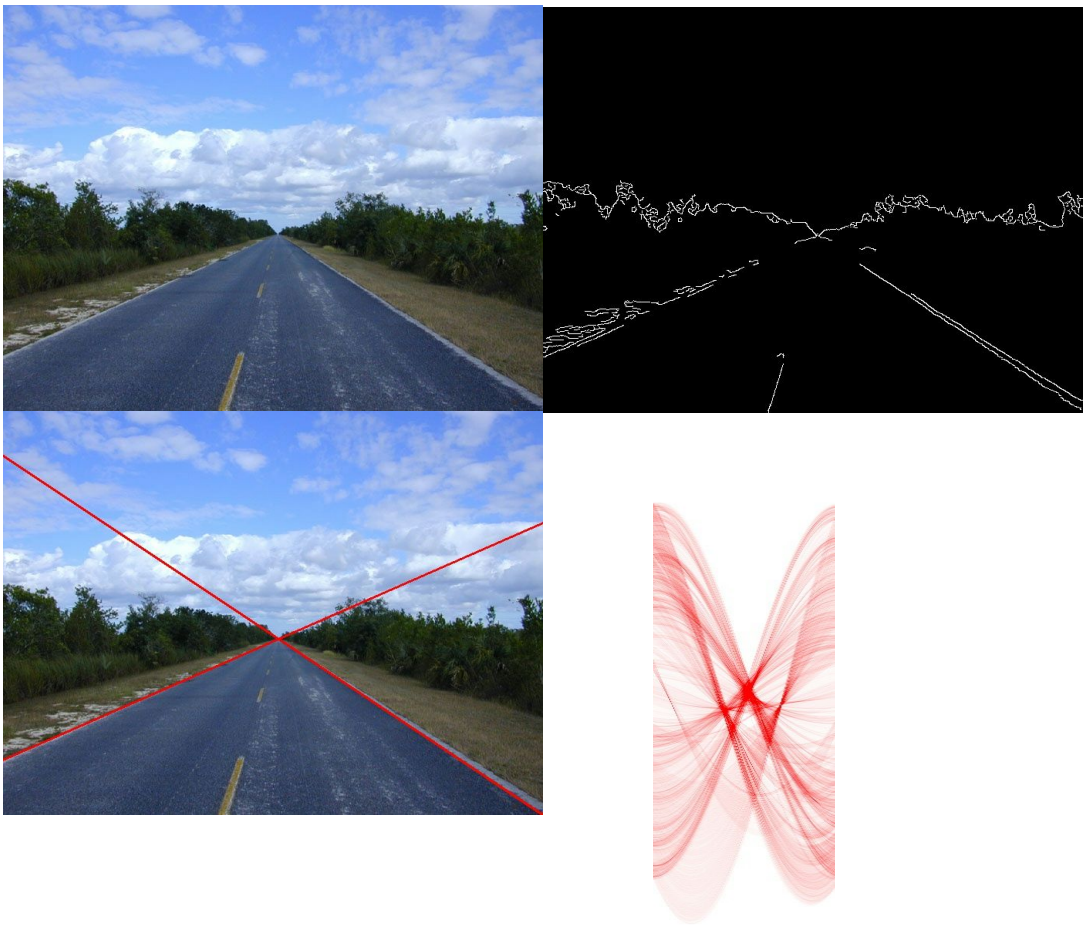


Matthew England
2/15/2016

Square Image (Threshold = 120):

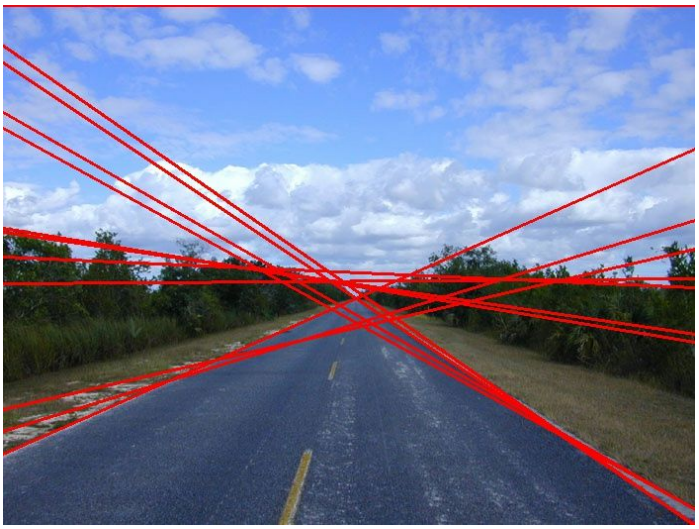


Road (Threshold =120): Lines 2

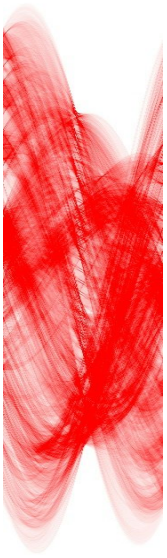


Matthew England
2/15/2016

Road (Threshold = 60): Lines 14



Building (threshold =):



Matthew England
2/15/2016

Conclusion:

As you decrease the threshold for the hough transform, you will always get an increase found lines. However each image requires different thresholds for line detection. For example the square image decreases the accuracy using a 120 threshold. As you increase the threshold, the hough transforms provides us with valuable information, which lines are the strongest in the image. If you need to find distinct lines, such as the shoulder of a road, then hough transform can provide valuable assistance.

I do have some issues with creating the lines with most of the the images. Even the easy square I get the width and height lines correct, but I miss the initial x and y corner edge. In my solution I believe I have the issue with selecting the the points that deal with 0,0 becuase I want the line to span the entire image, thus in the rare instance it fails by selecting the top corner. Another idea for the possible error comes from canny edge detector from opencv not providing positive edges, because of the magnitudes doing magnitudes. I figured there solution was similar to ours, but that is not the case.

However, I do have great results on road line detection. As one can see I get the edges of the road until it becomes a vanishing point and my lines cross. I will work to find the issue in my hough transform code.