CS 675 – Computer Vision – Fall 2007
Instructor: Marc Pomplun

Assignment #2

Posted on October 8
Programming questions (Q1 and Q2) due by October 31
Other questions due by October 16 before class

Question 1: Adding the Canny Edge Detector

Now you are ready to add an "Edge Detection" button. When the user presses this button, the following operations will be carried out on the grayscale image on the left side:

- Gaussian smoothing with a filter of size s×s using the method from Assignment 1,
- Gradient computation with two 2×2 filters,
- Edge thinning using the sector method,
- Binary thresholding of edge values using a threshold t (for receiving bonus points, allow users to choose whether they want to perform thresholding or not).

The result is displayed on the right side, where above-threshold edge values are indicated by black pixels and all other values by white pixels. Basically, the edges in the original image should appear as dark lines on white background. Use several test images to determine values for s and t that work best for most images.

Question 2: Implementing the Hough Transform

Now your nightmares are coming true: It is time to implement the Hough transform for finding straight lines in edge images. When the user presses a button labeled "Hough Transform," a window pops up asking for the number of straight lines to de detected. The user enters this number, and then the Canny edge detection is carried out on the input image on the left side. On the result of the Canny edge detector the Hough transform is computed, using the length and the angle of the normal (= perpendicular connection to the origin) of a straight line as its two parameters as discussed in class.

The Hough transform uses a 450×450 array of counters. The y-axis corresponds to the length *d* of the normal. The maximal length for our 450×450 pixel images is the length of its diagonal ($\sqrt{2} \cdot 450$ = 636 pixels). In order to represent the interval [0,636] on the y-axis with 450 different values, we multiply the length of the normal by 449, divide it by

636 and apply the floor function to obtain a value in the desired interval [0, 449]. The x-axis corresponds to the angle α of the normal ranging from 0 to 360 degrees (or from 0 to 2π, or from $-\pi$ to π). We use the same method as for the y-axis to map these values onto the interval [0, 449].

Once the Hough transform is computed, your algorithm determines the maximum counter value in the array. Afterwards, the values of all 450×450 counters are divided by this maximum so that the value of each counter is normalized to the real-valued interval [0, 1]. The result is shown in the program's right window, where 0 is displayed as white, 1 as black, and other values as shades of gray.

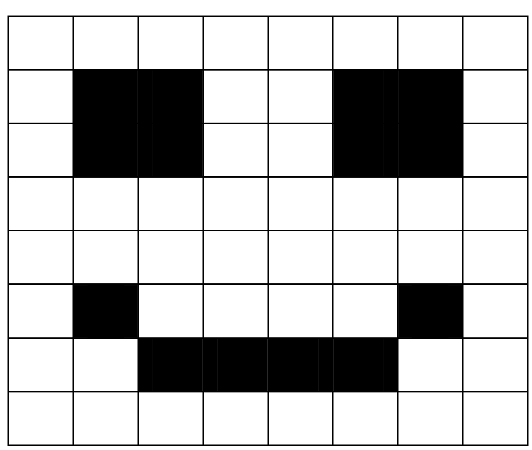
Then the local maxima in the counter array are determined. We define a local maximum as a counter whose value is greater than or equal to all its neighbors of chessboard distance smaller than six. Then, depending on the number n of lines the user chose to be detected, determine the n local maxima with the largest counter values. The parameters (p, α) at these points are the parameters of the n clearest lines in the image that the user wants to see. So the last step is to visualize these lines in the original image on the left side. You could use black pixels to show the lines, or even better, you could use a white pixel whenever the original pixel is dark (< 128) and a black pixel otherwise.

Question 3: Find the Circle

Imagine that you found the three edge points (1, 1), (-2, 4), and (0, 6). Determine the center and radius of the circle that passes through all of these points. Please write down your complete computation.

Question 4: Growing Trees

Write down the quad tree representing the following binary image:



Question 5: Image Filtering

a) Apply a 3×3 median filter to the 5×5 image below and enter your result in the empty 3×3 matrix on the left. Then apply a uniform 3×3 smoothing filter (i.e., one that has the same value in every cell) to the original 5×5 image and enter your result in the empty 3×3 matrix on the right.

0	150	0	150	0
0	150	0	150	0
0	150	0	150	0
0	150	0	150	0
0	150	0	150	0

result of median filter

result of smoothing filter

b) If you want to blur an image (i.e., make it look like it were out of focus), which of the two filters would you use? Explain why.

Question 6: Texture Comparison

Consider the following four black-and-white textures (1) to (4):

1	1	0	1	1	0
0	1	1	0	1	1
1	0	1	1	0	1
1	1	0	1	1	0
0	1	1	0	1	1
1	0	1	1	0	1

(1)

0	1	1	0	1	0
1	0	1	0	0	1
0	1	0	0	1	1
0	0	1	0	1	0
1	1	0	1	0	1
1	0	1	0	0	1

(2)

1	0	1	0	1	0
1	0	1	0	1	0
1	0	1	0	1	0
1	0	1	0	1	0
1	0	1	0	1	0
1	0	1	0	1	0

(3)

1	0	1	0	1	0
0	1	0	1	0	1
1	0	1	0	1	0
0	1	0	1	0	1
1	0	1	0	1	0
0	1	0	1	0	1

(4)

a) For each texture, determine two 2×2 co-occurrence matrices, namely one for the displacement vector (1, 1) and another one for the displacement vector (1, -1).

b) Based on these matrices, define a suitable pairwise distance (or difference) measure for black-and-white textures. This measure should increase with greater differences between two textures. Identical textures should yield zero. Enter the pairwise distances, according to your measure, between textures (1) to (4) into a symmetrical matrix:

	(1)	(2)	(3)	(4)
(1)				
(2)				
(3)				
(4)				

According to your results, which are the two most similar textures among these four?

Question 7 (Bonus Question): Here Comes the Math

Assume that you found a contour that is precisely shaped like a parabola defined by the equation $y = x^2$. Determine the radius of its osculating circle in the point (1, 1), and the osculating circle in the point (2, 4). Describe how you compute the result.