

Student Name JoonHyuck

Grader Name TK

Score 90 (661's max points: 120, 461's max points: 105)

	Points Worth	Full Credit
README	5	<ul style="list-style-type: none"> <li>Submission contains a README with detailed explanation where necessary.</li> </ul>
P1	5	<ul style="list-style-type: none"> <li>Cleanly thresholds both two_objects.pgm and many_objects_2.pgm. A reasonable threshold value is 100~140.</li> </ul>
P2	20	<ul style="list-style-type: none"> <li>Correctly labeled mask generated. Test on two_objects.pgm and many_objects_2.pgm.</li> <li>Visualization is clear and correct.</li> </ul>
P3	15	<ul style="list-style-type: none"> <li>Correct database returned using the label mask. Test on two_objects.pgm and many_objects_2.pgm.</li> <li>Visualization is clear and correct (ie. Centroid and orientation look reasonable)</li> </ul>
P4	5	<ul style="list-style-type: none"> <li>Use two_objects.pgm to generate the database..</li> </ul> <p>Test recognition on many_objects_2.pgm. A correct algorithm should at least always be able to find the object on the right in two_objects.pgm in the many_objects_2.pgm</p> <ul style="list-style-type: none"> <li>Comparison criteria is reasonable</li> </ul>
P5	10	<ul style="list-style-type: none"> <li>Finds edges in the image using SG-op or Laplacian. Using built-in convolutions functions are ok.</li> </ul>
P6	20	<ul style="list-style-type: none"> <li>README clearly states the correct constraints to limit the size of accumulator array and the ranges of possible theta and rho values. (rho = bounded by image diagonal length, theta = -90~90 degrees)</li> <li>README clearly states the choices of accumulator resolution, voting scheme and edge threshold. These choices should be reasonable.</li> <li>Hough_image_out displays sinusoids with intersections.</li> </ul>
P7	10	<ul style="list-style-type: none"> <li>Visualization clearly picks out strong edges in both hough_simple_2.pgm and hough_complex_1.pgm.</li> </ul>
P8	10	<ul style="list-style-type: none"> <li>Lines are pruned into segments cleanly. Reasonable things to do here include looking at line intersections or finding segment end points using gradient information along</li> </ul>

**WRITTEN PORTION**

	Points Worth	Full Credit
1.a	5	Refer to the solution
1.b	5	
1.c	5	
2	5	Refer to the solution

Partial Credit	No Credit
	<ul style="list-style-type: none"> <li>No README included in the submission</li> </ul>
<ul style="list-style-type: none"> <li>(-3) Objects are lost or artifacts introduced due to a bad threshold value.</li> <li>(-2) Column major traversal</li> </ul>	<ul style="list-style-type: none"> <li>Uses matlab/python's built-in threshold functions (cv2.threshold, otsu's thr</li> <li>Crashes</li> </ul>
<ul style="list-style-type: none"> <li>(-5) Wrong number of labeled objects detected.</li> <li>(-5) Small edges of 'wrongfully labeled objects' exist. This is mostly due to buggy equivalence tables.</li> </ul>	<ul style="list-style-type: none"> <li>Built-in region-proposal functions used.</li> <li>Hard coded outputs</li> <li>Crashes</li> </ul>
<ul style="list-style-type: none"> <li>(-1) for each attribute not included. (object_label, (x,y) centroid, min_moment, orientation, roundness must be included)</li> <li>(-5) x and y (col and row) flipped.</li> <li>(-5) Reported values and visualization do not make mathematical/intuitive sense.</li> </ul>	<ul style="list-style-type: none"> <li>Built-in descriptors used.</li> <li>Hard coded outputs</li> <li>Crashes</li> </ul>
<ul style="list-style-type: none"> <li>(-2) Unstable recognition: algorithm does not always recognize the objects in the query image.</li> <li>(-3) Visualization does not display only the recognized objects.</li> </ul>	<ul style="list-style-type: none"> <li>Built-in matching functions used.</li> <li>Hard coded outputs</li> <li>Crashes</li> </ul>
	<ul style="list-style-type: none"> <li>Hard coded outputs</li> <li>Crashes</li> </ul>
<ul style="list-style-type: none"> <li>(-5) No discussion of theta/rho range</li> <li>(-5) No discussion of design choices for accumulator resolution, voting scheme and edge threshold.</li> <li>(-10) Detected lines are not in line with real lines in the image</li> </ul>	<ul style="list-style-type: none"> <li>Lines are not found consistently</li> <li>Hard coded</li> <li>Crashes</li> </ul>
<ul style="list-style-type: none"> <li>(-3) No thresholding which results in many noisy lines in the visualization</li> <li>(-2) Error in plotting even with the correct accumulator</li> </ul>	<ul style="list-style-type: none"> <li>Hard coded outputs</li> <li>Crashes</li> </ul>
<ul style="list-style-type: none"> <li>(-5) Inconsistency in pruning (some are pruned, some aren't)</li> <li>(-5) Pruned lines do not correspond to the actual line segments in the image.</li> </ul>	<ul style="list-style-type: none"> <li>Hard coded outputs</li> <li>Crashes</li> </ul>

Partial Credit	No Credit
Give partial credits where applicable.	

Points earned	Comments
5	
5	
20	
15	
5	
10	
20	
0	Error in visualization
0	

Points earned	Comments
5	
0	Should be $\frac{1}{4}$
5	
0	