Computer Vision cs682 **Due: Apr. 22, 2019**
Spring 2019
Dr. Duric

Homework #4 (70p)

In your fourth homework you will explore methods for detecting lines in images from Homeworks #2 and #3. First, you will explore the *Radon transform* and compare it to the *Hough transform*. Second, you will find the intersections of the detected lines to estimate the vanishing points of the images. It is important that you make your *Python* code efficient, i.e. you should avoid loops over image points whenever possible. You may have to explore "fancy indexing in numpy" and similar terms to figure out how it can be done. It is OK to share the ideas on piazza with your classmates.

Midterm related: 50% of your grade on Parts i. and ii. will be added to your midterm grade. In all your work you should start by using edge detection to detect significant edge points in the image. You may use Canny edge detector for that, or alternatively you may use Sobel masks with some suitable threshold which detects important edges in the image.

The homework has the following parts:

i. (20p) **Radon Transform**: If you are looking for lines at some angle α_0 then for each feature (edge) point (x, y) we have $d = x\cos\alpha_0 + y\sin\alpha_0$. Note that the accumulator array becomes 1d and finding lines reduces to finding peaks in d. If you find a peak at some $d = d_i$ then the line equation becomes $d_i = x\cos\alpha_0 + y\sin\alpha_0$. Implement a *Python* function *FindLines(α)* which finds lines at angle α using this method. You should use appropriate thresholds to find only significant lines. Justify your choice. You should also find points which contribute to the selected lines and display the results (10p for this step). Demonstrate your results on 2 different images for two different angles. You should find some significant lines by hand and then show that your algorithm finds those lines. You should also demonstrate how the algorithm works for two randomly selected angles.

ii. (20p) The method described above assumes that you know the line orientations. In this part you will explore ways to find significant lines. First you will use edge histograms to detect likely lines. You will use 36-bin histogram, with 5 degrees per bin. Note, that the edges whose orientation is opposite—i.e. differ by π—should be binned together. The peaks α_i of the histogram should be used as a guess for significant lines. Note that you should probably search in range $\alpha_0 \pm 3^\circ$ in increments of 1°. Demonstrate your results on 3 images.

iii. (10p) Use the *Hough transform* and the *Probabilistic Hough transform* from *OpenCV* to find lines in images from Part ii. Compare the results.

iv. (20p) Read about the vanishing points at the Wikipedia. Use the lines found in Parts ii. and iii. to detect them. Show the results on 3 images.

Submitting your homework

Post the results and your programs/scripts on your webpage; write a report describing your work. Your report must be clear and as brief as possible without compromising comprehension.

Post your report on the blackboard. If would be nice if you could use password protection and include the login and the password in your report on the blackboard.