Computer Vision 1: Homework 7

Important: Mark the homeworks you solved in the homework sheet and bring your solutions with you to the exercise class. For each homework problem, one student will be chosen at random to present their solution.

Programming tasks.

Figure 1 is an aerial view of an airport. We will use Hough transform to detect the principal runway in the image. You can find this image `airport.tif` on Moodle.

Figure 1: Aerial view of an airport. Source: Gonzalez/Woods' DIP book

- Read the image and generate the edge image using Canny Edge Detector with $\sigma = 2$. Choose the lower threshold high enough to get rid of the edges around the principal runway.
- Using `skimage.transform.hough_line`, compute the Hough transform of the image. Note that the input to the Hough transform is the binary edge image.
- Using `skimage.transform.hough_line_peaks`, extract the first 3 peaks in the Hough transform.
- Visualize the edge image, the Hough transform and the lines corresponding to the 3 peaks together. The result should be similar to Figure 2.

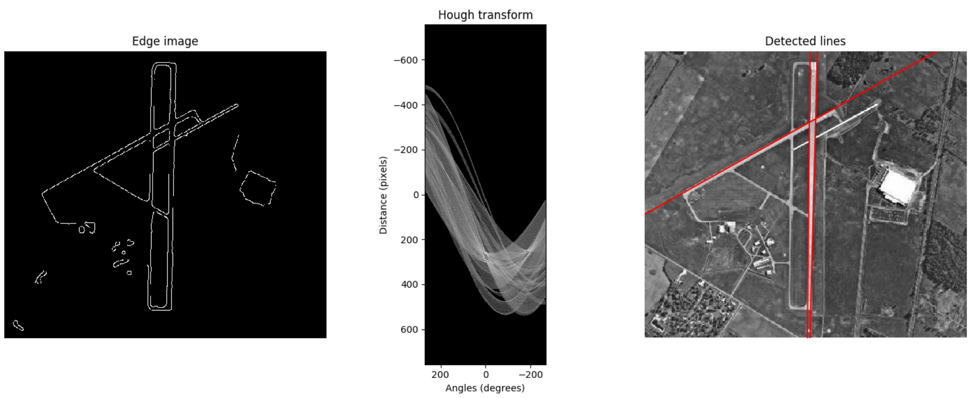

Figure 2: Hough transform result with 3 peaks.

Hint: see an example of plotting Hough transform at `http://scikit-image.org/docs/dev/auto_examples/edges/plot_line_hough_transform.html`.

- Add some Gaussian noise into the original image and repeat the process above. Try out different amount of noise. Do the first 3 peaks still correspond to the 3 edges of the principle runway? What can we conclude about the robustness of the process against noise?

Other tasks.

1. Given a point $P(x, y)$ in a 2D xy-plane and a line L runs through P. The point $H(x_0, y_0) \in L$ is the projection of the origin $O(0, 0)$ on L, the length of OH is ρ and the angle between OH and the x-axis is θ. See Figure 3 for a visualization. Prove the normal representation of L:

$$x \cos\theta + y \sin\theta = \rho$$

2. Recall that 2-dimensional Hough space is a 2D grid divided into accumulator cells. Suppose the size of the grid is $P \times Q$ where P is the number of θ's and Q is the number of ρ's, the number of edge points is N. Compute the time complexity of the Hough transform for 2D line detection as a function of P, Q and N. If not all possible θ but only the gradient of each pixel is considered, how does that complexity change?

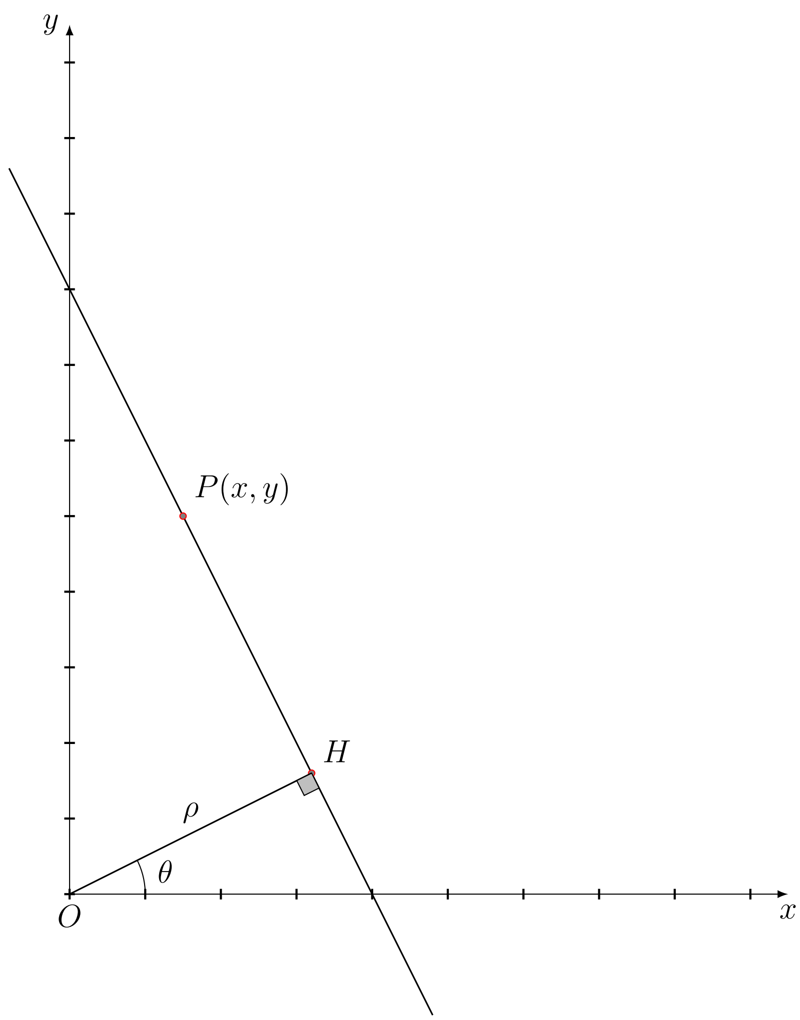

Figure 3: (ρ, θ) parameterization of a line in the xy-plane.