Pattern Recognition
Lab 3

Useful matlab functions:
`edge`, `find`, `imshow`, `imwrite`, `imagesc`, `imdilate`, `line`, `hough`, `houghpeaks`, `houghlines`, `adapthisteq`, `im2double`, `imfindcircles`, `viscircles`

Guidelines for lab reports:

- Always give a (short) explanation of what you are doing.
- Do not forget to include your Matlab programs. Present and discuss the results of your programs, be it a number, a matrix or an image.
- Put large pieces of Matlab code in an appendix.
- One should be able to understand plots independently, be sure to label axes, add a legend for colors, etc.
- Refer to all plots, tables, code blocks, etc. in your report.
- If you print gray-scale make sure the colors used in the plots are distinguishable.

Assignment 1:

In this exercise you will apply the MATLAB function `hough`.

1. Read and show the image `cameraman.tif`. This image is supplied by MATLAB.
2. Compute the edge map of the image usign the Canny algorithm with the MATLAB default parameters. Show this edge map.
3. Compute the hough transform accumulator of the edge map.
4. Show the accumulator array.
5. Threshold the image to keep only the strongest responses of the accumulator array. Define your own definition of what the strongest responses are.
6. Find the local maxima in the thresholded accumulator.
7. Show the accumulator array and mark the five strongest local maxima points on it.

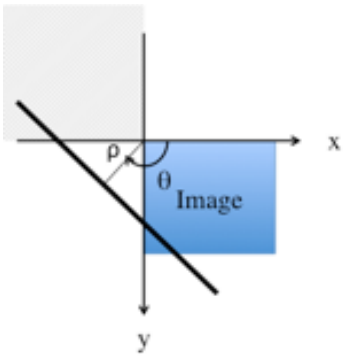

Figure 1: Image in spatial coordinates (image space), with the associated ρ and θ shown.

8. Show the original image, `cameraman.tif`, and overlaid the strongest line. Create a new function `myhoughline`, that takes as inputs an image, a ρ and a θ and that draws a line at perpendicular distance ρ from the upper left corner of the figure, with angle θ from the top horizontal edge of the image to the perperdicular.

hint: Create a spacial case for vertical lines.

hint: Note the change between Cartesian and Space coordinates.

Assignment 2:

In this exercise you will write your own MATLAB function `myhough` that uses the Hough transform based on parameterizations of the type

$$\rho = x \cos \theta + y \sin \theta, \tag{1}$$

where ρ and θ are respectively the orientation and distance to the center of a line.

1. Decide on a discrete set of values of θ and ρ to use. The usual convention is to take values of θ in the range $\frac{\pi}{2} < \theta \leq \frac{\pi}{2}$ and let ρ have both positive and negative values. However for the shake of simplicity, we are going to use the values of θ and ρ as indices in the accumulator array by restricting ρ to non-negative values. Figure 1 represents an image in spatial coordinates (image space). Restricting ρ to positive values, which values of θ should be considered to represent lines in an image?

2. The function `myhough` should accept an edge map as input and output the accumulator array. Take the steps below in `myhough`:
   a) Determine which values of θ you are going to use.
   b) Find the foreground pixels of the edge map.
   c) For each foreground pixel (x, y) in the edge map calculate the values of ρ according to (1), for all of our chosen values of θ. Round ρ values to the nearest integers less than or equal to the obtained ρ value.
   d) Initialize a two-dimensional accumulator array. The size of the array is determined by the number of angles θ, and values ρ.

   hint: We can first find the largest positive value of the computed ρ values, $\rho_{\max}$, and use $\rho_{\max} + 1$ as one dimension of the array. We add 1 to allow for $\rho = 0$.

e) Step through all of the values of ρ updating the accumulator array as we go. Keep in mind we are only considering positive values of ρ.

Assignment 3:

In this exercise we will use the implemented functions of MATLAB to understand the Hough transform for lines.

1. Create a 50×50 black image with a single white pixel. Compute the Hough transform of the image. Show both the image and the Hough space of the image.
2. Create a 50×50 black image with a three non-aligned white pixels. Compute the Hough transform of the image. Show both the image and the Hough space of the image.
3. Create a 50×50 black image with a three aligned white pixels. Compute the Hough transform of the image. Show both the image and the Hough space of the image.
4. Compare and explain the results obtained in exercises 1, 2 and 3.
5. Use `houghpeaks` to find the maxima in the Hough space from the image with the three aligned points. Show the maximum peak in the Hough space.
6. Extract the line from the `houghspace` of the image with tree aligned points using `houghlines` and show it in the original image, marking the beginning and end of the line.
7. Repeat steps 5 and 6 with the figure `chess.jpg`.

hint: Do not forget to compute the edge map of the image.

8. Show the 15 strongest lines in the image `chess.jpg`.

Assignment 4:

In this exercise we will use the implemented functions of MATLAB to understand the Hough transform for circles. Our aim is to find the circular screws of an industrial image of a milling machine.

1. Read the image `HeadTool10002.bmp` and convert it to double precision.
2. Apply the contrast-limited adaptive histogram equalization method to enhance the contrast of the image.
3. Find at least 6 circles, if two circles belong to the same screw, keep only one of them.

hint: Try radii between 20 and 40 pixels, and different values of sensitivity with `imfindcircles`.

4. Show the enhanced image from step 2 with the circles found in exercise 3.
5. Show the enhanced image from step 2 with the two strongest circles found in exercise 3.
6. Show the accumulator array.