CSE166 – Image Processing – Homework #5
Instructor: Prof. Serge Belongie
`http://www-cse.ucsd.edu/classes/fa10/cse166`
Due (in class) 11:00am Friday Nov. 12, 2010.

General Homework Guidelines

- Use the Cover Sheet provided.
- Please attach all code that you use. Attach code at end of submission.
- In general try to keep your answers concise. Use as many words as you need and no more. Also work on your presentation skills. This means organize your plots and displays. Always use titles and add captions when appropriate. *Points will be awarded for clarity and presentation.*

Written exercises

1. GW Second and Third Edition, Problem 8.1.
2. GW Second Edition, Problem 8.12.
   or
   GW Third Edition, Problem 8.8
3. GW Second Edition, Problem 8.14.
   or
   GW Third Edition, Problem 8.18
4. Consider the symmetric 2×2 matrix

$$A = \begin{bmatrix} a & b \\ b & c \end{bmatrix}.$$

   By finding the roots of the characteristic equation,

$$\det(\lambda I - A) = 0,$$

   show that the eigenvalues of A are given by

$$\lambda = \frac{\mathrm{tr}(A) \pm \sqrt{\mathrm{tr}(A)^2 - 4\det(A)}}{2}$$

Matlab exercises

1. Binary image processing.

   Before doing this problem, study the `bwmorph`, `imerode`, `imdilate`, `bwlabel`, and `regionprops` functions in Matlab.

   (a) Reproduce GW Second Edition, Figure 9.5(a,c).
   or
   Reproduce GW Third Edition, Figure 9.7(a,c).

   (b) Reproduce GW Second and Third Edition, Figure 9.14(a,b).

   (c) Perform connected components labelling on the particles image for GW Second Edition Problem 9.27 or GW Third Edition Problem 9.36.. Based on the area of each connected component, produce a new image containing only the isolated (nonoverlapping) particles. Assume all particles are approximately the same size.

Things to turn in:

- Printouts of output and code listings for steps 1a, 1b, and 1c.

2. Shape and the scatter matrix.

   This problem makes use of the result in written exercise 4 above and the additional fact that the angle of the principal eigenvector of A is given by

   $$\phi = \frac{1}{2}\tan^{-1}\left(\frac{2b}{a-c}\right)$$

   The angle of the other eigenvector is $\phi + \pi/2$. Note: in Matlab, use `atan2` to compute the inverse tangent. Do not use `imfeature` for this problem.

   (a) Load in GW Second Edition Figure 11.10 or GW Third Edition Figure 11.16, call it `I`, and binarize using the command `BW=I>128`. Use `find` to obtain the (x, y) coordinates of the nonzero pixels. Plot the resulting pointset using the `axis('image')` and `axis('ij')` options and the `'b.'` pointmarker.

   (b) Compute the centroid $\mathbf{m}$ and plot it in the preceding figure (turn `hold` on) using the `'rx'` pointmarker.

   (c) Compute and display the scatter matrix C. Find its eigenvalues and eigenvectors, first using the above shortcuts, then using the Matlab function `eig`, and demonstrate that both methods give you the same result.

   (d) Compute the aspect ratio of this shape using the formula $(\lambda_{max}/\lambda_{min})^{1/2}$.

   (e) Center the pointset so that its centroid lies on the origin. By visual inspection, estimate the rotation (in degrees) of the shape with respect to horizontal. Compare this to the estimate of the rotation provided by ϕ. Now derotate the coordinates so that the shape is oriented along the x-axis, and make a plot of the result.

   Things to turn in:

   - Code listing for all steps.
   - Plot for steps 2a and 2b.
   - Program output for steps 2c and 2d.
   - Written comments and plot for step 2e.

3. Estimating Affine Transforms.

   This exercise makes use of the pointset data contained in `coords.txt`, which is available on the course web page. The data file consists of two lists of six 2D coordinates, arranged in corresponding order.

   (a) Plot the two pointsets on the same set of axes. Use the `'x-'` pointmarker/linetype for the first set and `'o-'` for the second.

   (b) Solve for the least-squares affine transform (consisting of a 2×2 matrix A and a 2×1 translation vector $\mathbf{t}$) that maps the first pointset onto the second. Display the values you obtain for A and $\mathbf{t}$.

   (c) Use the estimated affine transform to align the two pointsets, and make a plot to show the alignment. Compute and display the sum of the squared Euclidean distances between the corresponding point pairs before and after alignment.

   Things to turn in:

   - Code listing, plots, and program output for each step.