Objective

The objective of this homework is to gain experience with 2D object extraction from an image that can be thresholded into a binary image. A secondary objective is for the student to become familiar with working with .pgm image files and simple programs that process them.

Problem 1

Modify and extend an existing connected components program to perform the following operations. Some resources are given below.

1. The program should read a P2 .pgm input file as described in Ch 2 (Fig. 2.12). You may assume that no image will be larger than 512 x 512.

2. The program should threshold the input image into background (0) and objects (1). The student can find an appropriate threshold by trial and error or by using the tool xv or gimp; it is not necessary to do it automatically.

3. The program should detect and summarize each separate object (blob) by giving its (a) area, (b) centroid, and (c) three second moments (rr, cc, rc). Clean output should be produced for the report.

4. Report the results for the images hw2F05-1A and hw2F05-1B.

Problem 2

Match up two objects from image 1A to similar objects in image 1B. Compute the rotation and translation that maps the coordinates of the centroids in image 1A to the corresponding centroids in image 1B. You can find an example of how to do this at the bottom of page 335 of the text. (You may do this by hand, calculator, or tiny program.) Then, map all of the centroids for regions that have correspondences and note the error for each (difference between the extracted centroid in image B and the result of mapping the centroid from image A).

Problem 3

Repeat Problems 1 and 2 using images 2A and 2B.
Resources

Students may use the following resources.

1. Students may use any of the C++ code referenced from the course or instructor web pages. If you use this code, you will have to make small changes and perhaps add new functions. Depending on your local compiler settings, you may not have enough program stack to recurse on large regions, so beware.

2. We will use this code again in a future homework, so take care not to hack it together too badly.

3. Students may use MATLAB, but may not use an image library function to perform connected components (coloring).

Notes

1. The due date had to be changed to give students a fair amount of time to complete the work. The original calendar had a 15/16 Sep due date.

2. Program code or tools used that are developed by others should be documented. This is not a programming project; the objective is not to produce a beautiful program, but rather to learn to use an algorithm and to get correct results.

3. Your report must show the use of the program, the results, and some discussion of the results. Review the report format given on the course web pages.

4. Partial credit will be given for partial completion of the tasks

5. Intermediate output beyond the requirements is allowed, but please do not submit verbose output in your report.

Remember: Academic Integrity

The critical language from the MSU code is as follows. “...all academic work will be done by the student to whom it is assigned, without unauthorized aid of any kind. ... If any instance of academic dishonesty is discovered by an instructor, it is his or her responsibility to take appropriate action.” Possible actions include assigning a failing grade for the assignment or course.

CSE 803 assignments are NOT to be done in groups. Discussion of course material relating to an assignment with fellow students is a valuable learning technique and is encouraged. Discussion of computer tools and clarification of assigned problems is also encouraged. However, group writing of programs or group solutions of assigned problems is NOT ACCEPTABLE. Reviewing a fellow student’s work AFTER submission is encouraged as a learning experience for both students.