BMI/CS 567 Medical Image Analysis
University of Wisconsin-Madison
Midterm Project

Instructor: Jeanette Mumford
Due March 28th *by* 2:30PM

You must work *independently* on the midterm. Please turn in your solution via Canvas, using the same format you've used for homework assignments.

Jigsaw Puzzle Problem

For this midterm you will solve a simplified version of the jigsaw puzzle problem: finding the proper placement of a puzzle piece. You have been given the two images in Figure 1, an almost complete jigsaw puzzle and a remaining piece of the puzzle. The piece will need to be shifted and rotated into position and you will do this using the generalized Hough transform and image registration. You can only use the approved built-in MATLAB image processing tools. See the Home Page on Canvas for the list or ask if you are not sure about using a specific function. Use of an unapproved function, such as imrotate, will result in losing points.

Figure 1: Two images that were distributed. The main puzzle (left) and the missing piece (right).

General instructions

You should avoid hard coding any numbers in your code. For example, in a grid search you cannot speed up the algorithm by eyeballing where the search should be and limiting the search to that

range. For thresholding or filtering it is understandable that you may need to hard code values, but you should explain your choices. Your code should also perform well if I was to run it on the other missing piece without altering your code.

Make your code as efficient as possible. Really think through what you're doing and try to think of ways to avoid looping, limit the number of loops or amount of processing within each loop. **How clever your code is will be reflected in your grade.** I say more about this below.

For this problem you can assume the shape information is sufficient to solve the problem. I do not expect you to match up the color or picture on the piece.

You will want to change the resolution of the image. Resize the pixels to lower the resolution by a factor of 5 (images will be 600×720 **and** 81×53**).**

As you'll see, the piece will need to be rotated to fit into the puzzle. **I only expect you to check rotations in increments of 45 degrees, for a total of 8 rotations.**

Hint: If you binarize your images, make sure the empty piece spot and the piece are both coded by the same values.

Generalized Hough Transform (20 points)

My goal is to see if you can implement the generalized Hough transformation based on what I've taught in lecture. I've specifically made sure the lecture contains all of the information you need to code the algorithm on your own. In the end, you should have two images: boundaries of the main puzzle and an image of the same dimensions with the piece boundary in the proper spot. Use `imfuse` to combine these two images to get the solution in Figure 2. It may not look exactly like my solution, but it will be close. I'm unsure how much room there is for sophistication in your code on this portion of the midterm as the algorithm already has a speedup built in. My version probably takes less than half a minute. Your grade will reflect how well you implemented the algorithm, how clean your code is and whether you avoided hard coding values. If I enter the red piece into your algorithm, I should still get the correct solution.

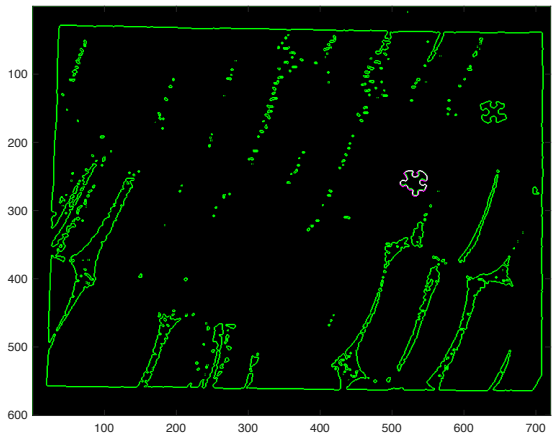

Figure 2: What your final answer should look like for the Generalized Hough. Green = boundaries on main puzzle image, pink = boundaries of my piece solution (that don't overlap) and white is the overlap of the two.

Image Registration (20 points)

The second solution will be obtained using image registration. If you simply run a grid search over all x, y and rotation values, it will likely take an hour or more. Although you would receive a grade at the AB level for this solution, speeding up your solution will result in a higher grade. I have thought of two ways to do this, one is around 20 minutes and the other is about 15 seconds! The full credit for this problem will only be given if you find the fastest solution. Your solution will look something like Figure 3 where you'll use imfuse to fuse the puzzle image with another image, the same dimension as the puzzle, but with the piece in the correct spot.

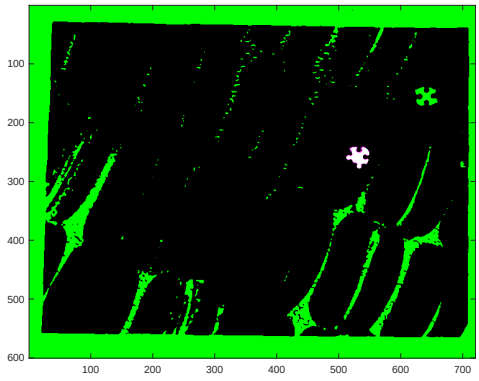

Figure 3: What your final answer should look like for the Image Registration approach. White = overlap of my piece solution with the piece slot in the puzzle. In the puzzle image 1=background and 0=puzzle area. In piece image 1=piece and 0=background.