UNIVERSITY OF WISCONSIN-MADISON, COMPUTER SCIENCE DEPARTMENT

CS767 - Assignment 2

Zhicheng Gu
Email: zgu58@wisc.edu
Student ID: 9073696370

October 23, 2016

1 PROBLEM 1

The java.util.PriorityQueue can't store a value and a index (value for sort and index is the point's x and y coordinate). I use the following package: http://www.mathworks.com/matlabcentral/fileexchange/24238-priority-queue–mex-c++- .

I use Dijkstra's algorithm to get the shortest path from seed point to destination point. The algorithm is shown in the following list.

```

Initialize the priority queue pq.
Initialize Visited matrix, set all to False.
Initialize Dist matrix, set all to a very large number.
Initialize Parent matrix, use to remember the path.

put seed into pq

while current point != dist point:
  current point, cost = pq.pop()
  Visited[current point] = True

  if Visited[current point]:
    continue

  for n in current point's neighbor:
    if cost + path(current -> n) < Dist[n]:
      Dist[n] = cost + path(current -> n)
      Parent[n] = current point
      pq.push(n)

Use Parent matrix to get the path.
```

The results are shown in Figure 1.1 and Figure 1.2.

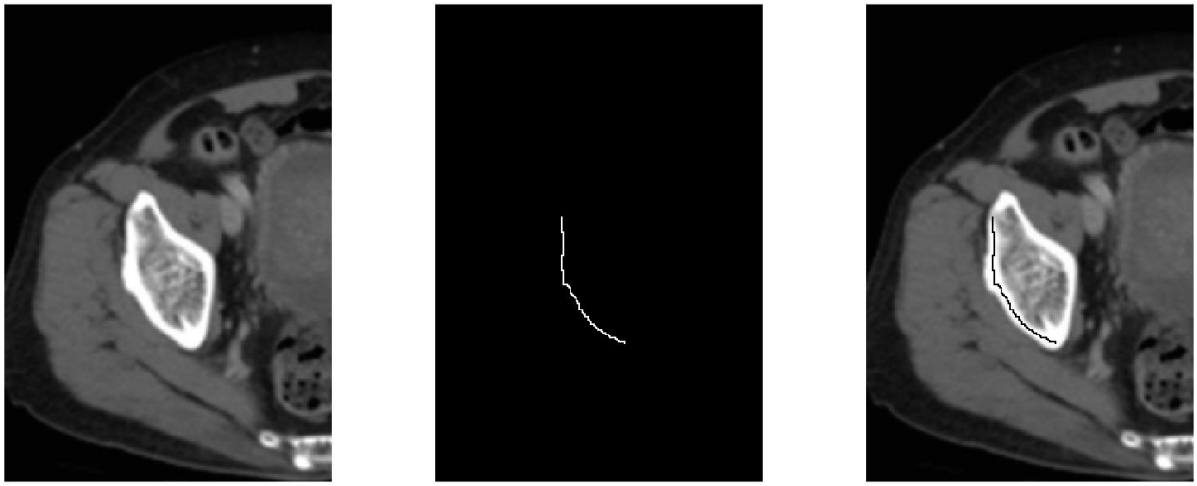

Figure 1.1: Result for scissors algorithm

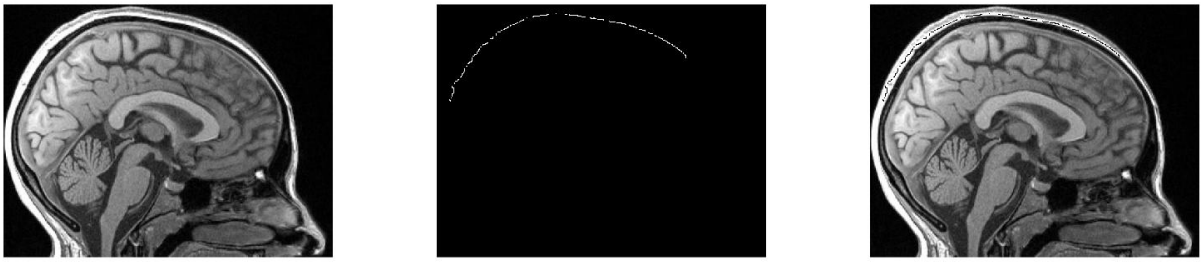

Figure 1.2: Result for scissors algorithm

2 PROBLEM 2

1 PART 1

I use the hough transform of with paramets d and θ: $xcos\theta - ysin\theta = d$. The resolution for d is 1 and 1 degree for θ. Some results are shown in Figure 2.1 - 2.3.

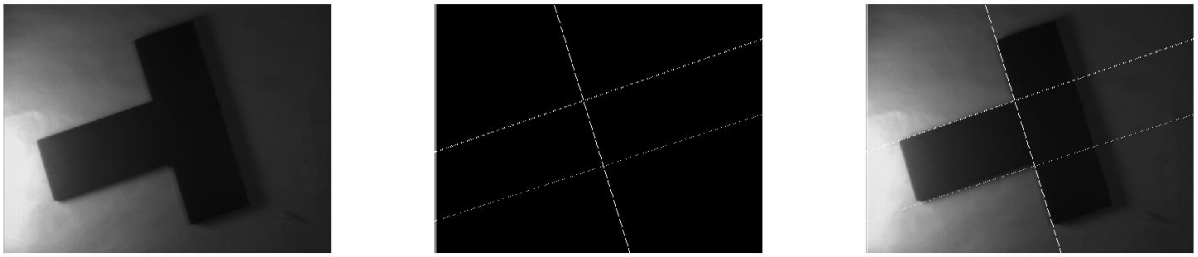

Figure 2.1: Result for hough transform

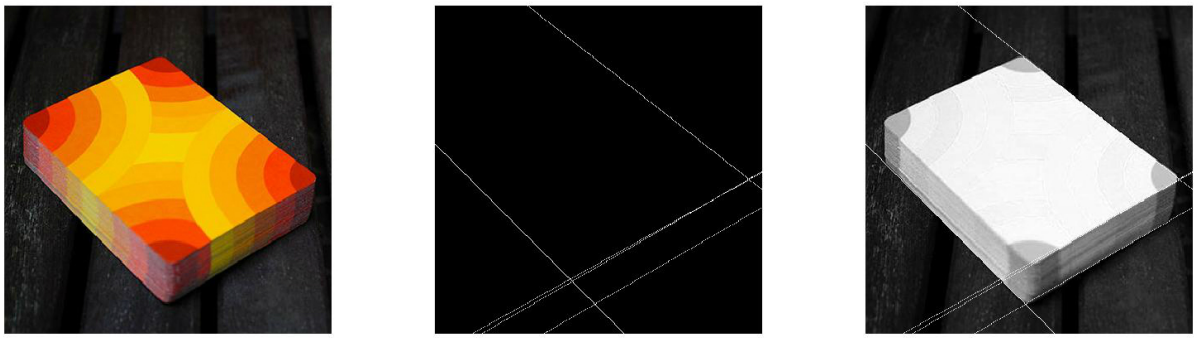

Figure 2.2: Result for hough transform

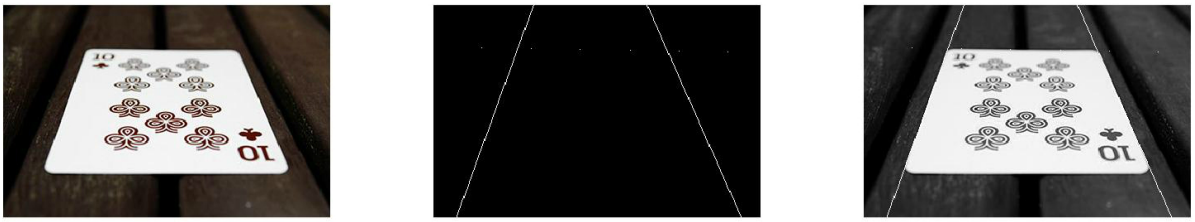

Figure 2.3: Result for hough transform

2 PART 2

Two reference points are (21, 118) and (32, 11).

In myHoughCircleTrain part, I store the relative position between center point and boundary point. That is, (center x - boundary x, center y - boundary y). Then in myHoughCircleTest method, for every boundary points, I mark the all possible center points by (relative x + boundary x, relative y + boundary y). Select the max two points in the hough space as the reference points.

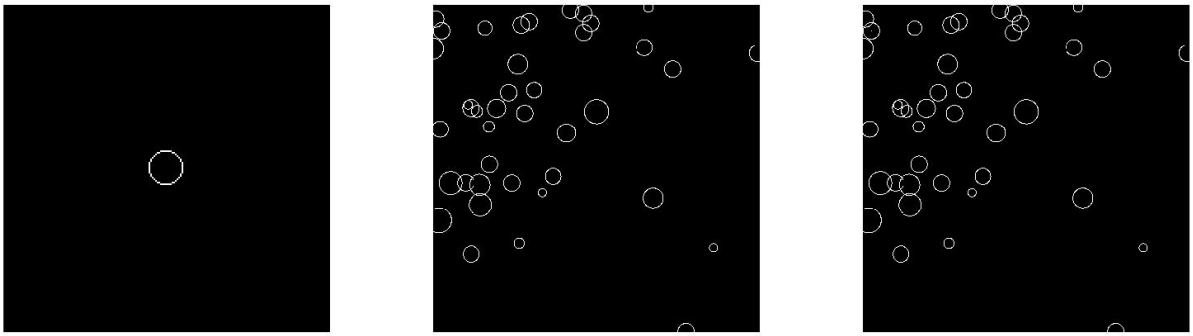

Figure 2.4: Result for circle finding, the reference points are marked in the third figure

3 PROBLEM 3

I use the dynamic programming method to get minimum energy for each iteration. I can get the best result with alpha=0.3 and beta=0.5. In gerneral, I found the more points I use, the better result I can get. I use about 50 points for the following results. The best result is shown in Figure 3.1 and Figure 3.2.

If the alpha is a little larger, say alpha=1, The result may be not optimal. From Figure 3.3, we can see that the snake shrink too much on the top part of the region.

If the alpha and beta is too large, such as alpha=10 and beta=10, the snake will shrink to a cluster. This is shown in Figure 3.4.

If the alpha and beta is too small, the snake will go along with the other boundary in the figure. This is shown in Figure 3.5.

If the number of points is too small, the result will be very bad. This is shown in Figure 3.6, with alpha=0.5 ,beta=1 and 10 points.

Figure 3.1: Final result with alpha=0.3 and beta=0.5

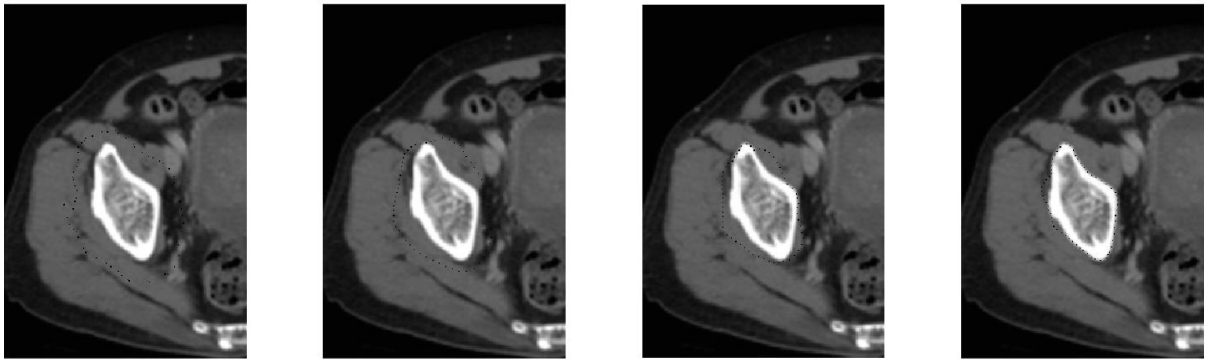

Figure 3.2: Result after every 10 steps with alpha=0.5 and beta=1

Figure 3.3: Final result with alpha=1 and beta=1, shrink too much on the top part of the region

Figure 3.4: Final result with alpha=10 and beta=10, shrink to a cluster

Figure 3.5: Final result with alpha=0.05 and beta=0.05, snake get on the other boudary

Figure 3.6: Final result with alpha=0.5 ,beta=1 and only 10 points