In this assignment, the goal was to segment the region of interest(ROI) that is provided by the given InitialMask.png.

The method I have used to segment the ROI is active contour, also called snake. The active contour is an energy-minimizing spline guided by external constraint forced and influenced by image forces that pull it toward features such as lines and edges. This method has some advantages and drawbacks.

Advantage:

- They autonomously and adaptively search for the minimum state.
- It could also be used to track dynamic objects

Drawbacks:

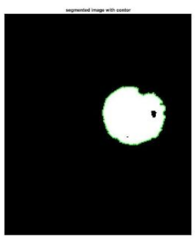
- Sensitive to local minima states.
- It is semi-automatic. We need to manually choose the ROI.

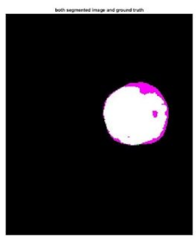
A simple elastic snake is defined by a set of n points v(s)=(x(s),y(s)). The energy function of the snake is the sum of its external energy and internal energy. The external energy is usually a combination of the forces due to the image itself E_{image} , and the constraint forces introduced by the user E_{con}. The energy function of the snake is the equation (1).

$$E^{*}_{snake} = \int_{0}^{1} E_{snake}(v(s))ds = \int_{0}^{1} E_{int}(v(s)) + E_{image}(v(s)) + E_{con}(v(s))ds \quad (1)$$

where E_{in} = internal elastic energy, E_{image} = image forces pushing the snake toward image features, E_{con} = external constraints are responsible for putting the snake near the desired local minimum.

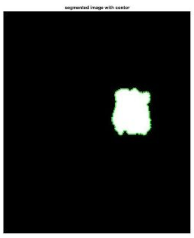
After applying the active contour to the test image, The final result is acceptable. The iteration of the active contour I have used is 70.

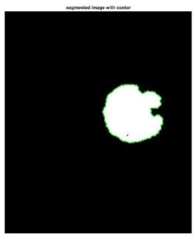


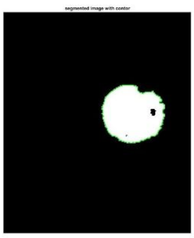


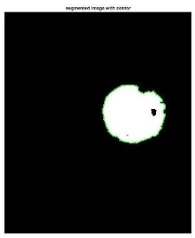
The image on the left is the segmented image with its contour. And the image on the right is the comparison between the segmented image and the ground truth.

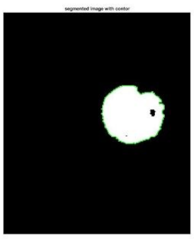
To find out the best iteration numbers, I have tried 10, 30, 50, 70, 100 iterations for the active contour. The result are shown in following image.











From the images, it is easy to tell that 10 iterations has the worst the result, and with increasing of the iteration number, the segmentation result gets better. And the visual result become stable when the iteration number reaches 50. The results of 50, 70, and 100 iterations are too similar to tell the difference. So I have used the four metrics to see the subtle changes in the results shown in the following table. The four metrics are(tp = true positive, tn = true negative, fp = false positive, fn = false negative):

- Precision: $\frac{tp}{tp+fp}$
- Recall: $\frac{tp}{tp+fn}$
- Dice: $\frac{2*tp}{2*tp+fp+fn}$
- Hausdorff: $HD(X, Y) = max(max(d_Y(x)),\ max(d_X(y))),$
  $d_Y(x) = min(d(x, y))\ is\ the\ distance\ if\ point\ x\ on\ X\ from\ Y$

Number of iteration	10	30	50	70	100
precision	1	1	1	0.9992	0.9995
recall	0.4715	0.8090	0.8535	0.8546	0.8526
dice	0.6408	0.8944	0.9208	0.9213	0.9202
hausdorff	20	18.9737	13.1529	13	13.8924

As shown in the table the precision remaining to 1 from 10 iterations to 100 iterations. The recall and dice score are increasing when the number of iteration increases. And the hausdorff distance drops when the number of iteration increases. This situation matches with the images. At the beginning, the segmentation was small but it is inside the ground truth, therefore the precision is very high, but the recall is low. After increasing the number of iterations, the segmented region grows, and the shape of segmented region becomes similar to the ground truth al well. I think the best result is when the number of iterations is 70. Even though the precision drops a little, but all other metrics are the best. When the number of iterations reaches 100, all metrics drops a little bit. I believe the region got over segmented.