Adam Pickeral

Active Contours – Implementing A Rubber Band & Balloon Model

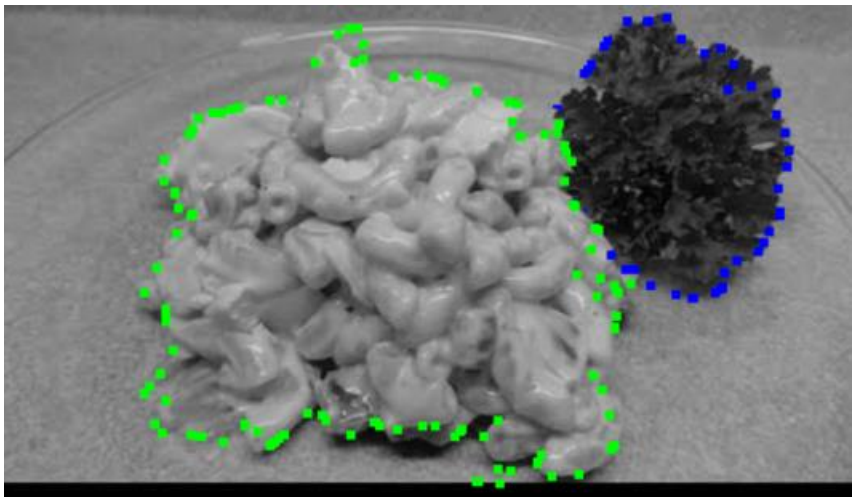
I wanted to implement a solution that can trace contours on PNM images. The end goal was to be able to implement 3 types of contours: a rubber band model activated by drawing a contour around an object by left clicking, a balloon model using a circular contour activated when right clicking, and a small active contour that adjusts a few nearby points when a drawn contour point is shift clicked and dragged. The backbone for this project is WIN32 C programming – the GUI and all implementations of code are written in this way. After the pnm images were loaded into the program, they were size to fit the window and converted to grayscale PPM format.

Rubber Band Model

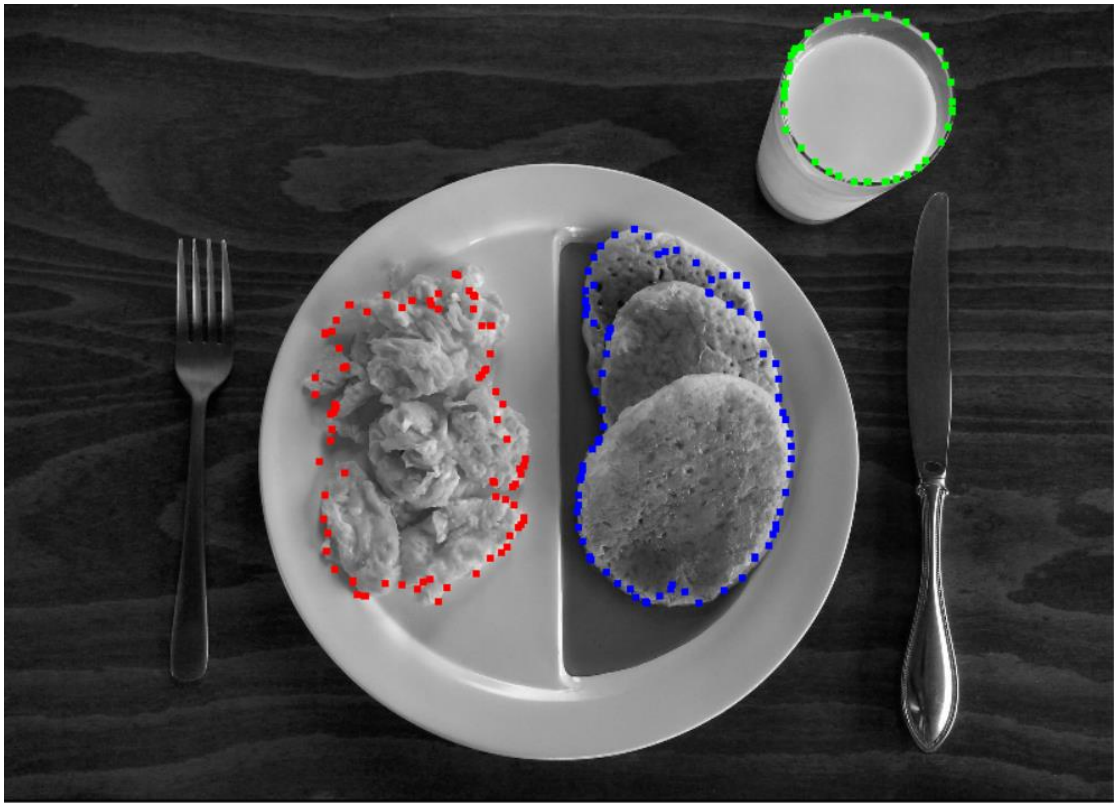
For the implementation of the rubber band model, 3 energy terms were used. The two internal energy terms were 1) the distance squared between the current contour point and the next, and 2_ the deviation from the average distance between points. The external energy came from the negative of the normalized sobel image to encourage the point to move toward edges. In order to put emphasis on certain terms, I used the following weights when computing the final energy as a sum of all 3 terms.

- Distance Energy : 1.2
- Deviation Energy: 1.2
- External Energy: 1.0

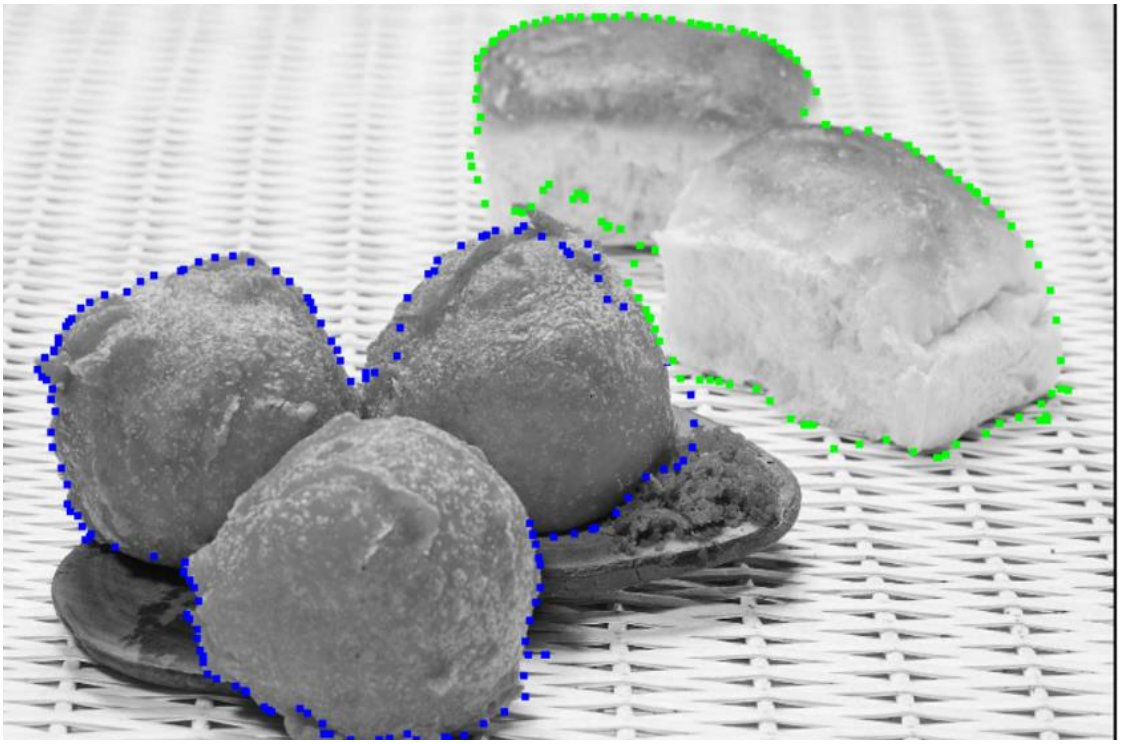
I ran the algorithm with a 15x15 window size for 20 iterations. The results are shown below.



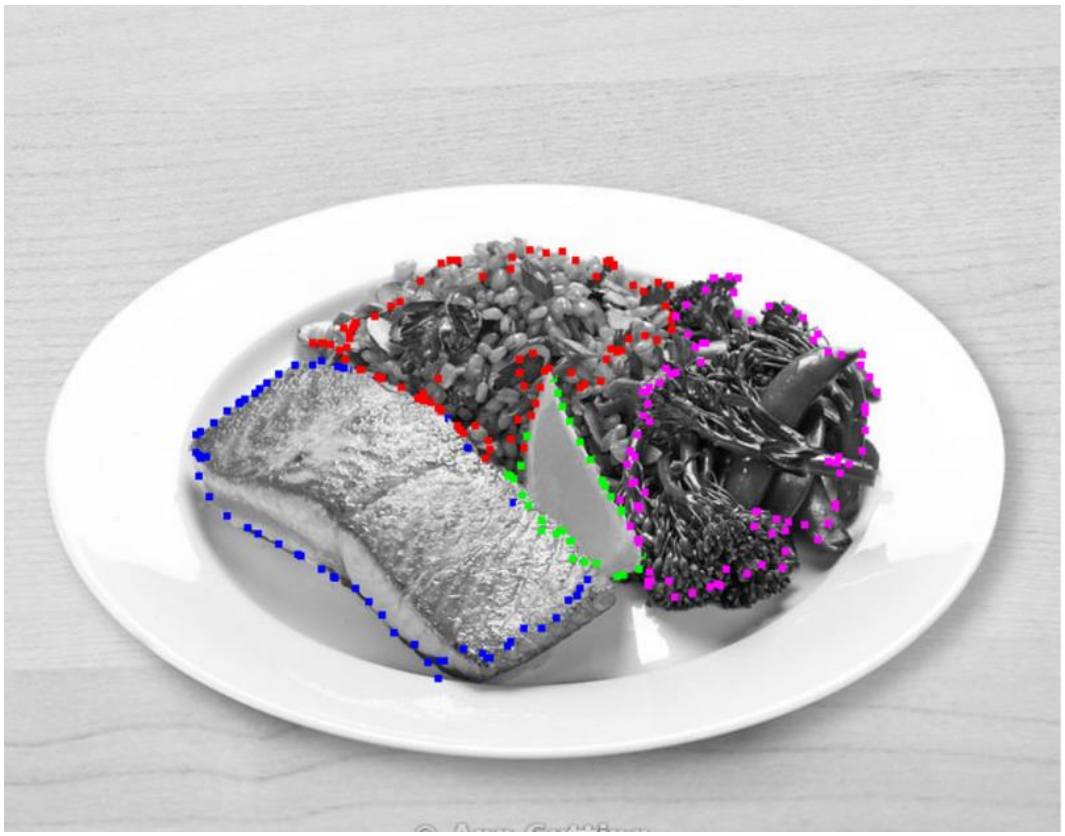
“macaroni-kale.pnm” Segmentation



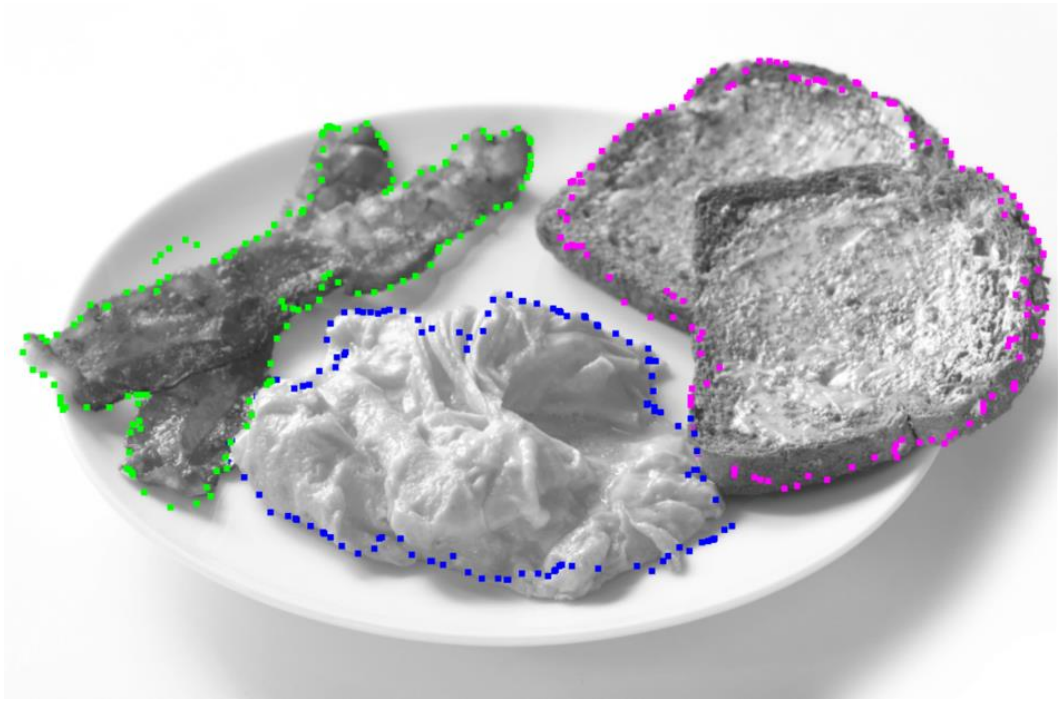
“eggs-pancakes-milk.pnm” Segmentation



“hushpuppies-biscuits.pnm” Segmentation



“fish-lemon-rice-greens.pmn” Segmentation



“bacon-eggs-toast.pmn” Segmentation

The results show that the contour is accurate in determining where each region is relatively. The main problem it had was with foods that had edges within them. For example, eggs and macaroni have a lot of texture within them, which could cause the contour to move inward unnecessarily. The pancakes, like the bottom of the first biscuit, have an outer edge that is lighter with a darker inside edge, causing the contour to move slightly inward.

Balloon Model

For more circular shapes like the milk glass inside and the hushpuppies, it is possible to implement a balloon model that expands outwards to fit a shape. For this project, the initial contour for the balloon model was a circle with radius 10. To accomplish this, I implemented Bresenham's circle drawing algorithm in my code, which draws a circle with 64 points. The initial contour is shown below.

Initial Rubber Band Contour, Circle of Radius 10 px

After drawing the initial contour, I down sampled it and only kept every third point.

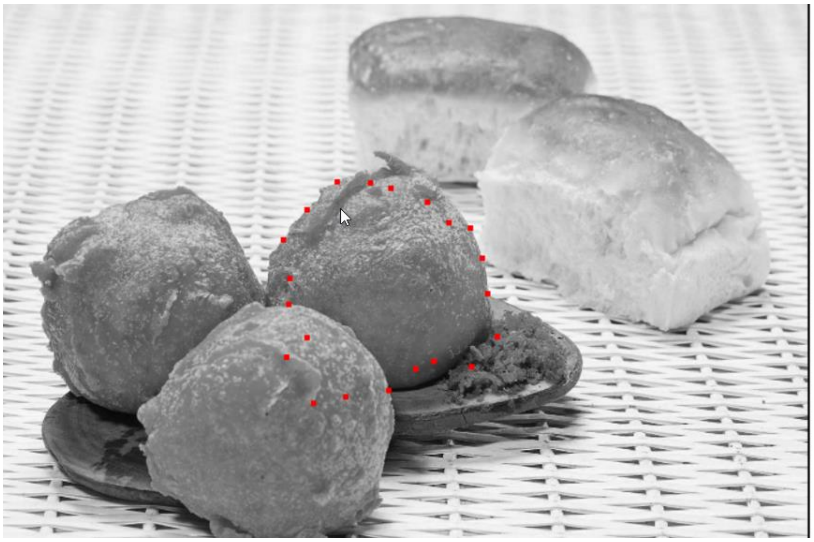
For the implementation of the rubber band model, 3 energy terms were used. One of the two internal energy terms were different. One of the internal energy terms was still the deviation

from the average distance between the current contour point and the next, but the other internal energy term was set to the distance of the current point from the initial centroid of the circle. This encouraged the contour to expand. The external energy came from the negative of the normalized sobel image to encourage the point to move toward edges. Since I needed the shape to remain stable and expand outwards at the beginning, I left the sobel energy out of the process for the first 5 iterations of the algorithm, allowing the contour to expand freely. After the first 5 iterations, I used these weights to computer the final energy.

- Distance Energy : 1.2
- Deviation Energy: 1.2
- External Energy: 2.25

This process allowed the contour to expand outwards, and then snap to its target. I also used a 15x15 window to calculate the energies and ran the algorithm for 20 iterations again too. The results for the milk and hushpuppies are shown below.

Milk Balloon Contour Result



Hushpuppy Contour Results

The results for the milk glass are fantastic. The contour snapped well and held its shape. For the hushpuppies, it isn't as great. The algorithm doesn't give it as many expansions as it needs to get to where it needs to snap, so it doesn't hold its shape as well. The hushpuppies are also significantly larger than the milk glass, and in the future more contour points could be added to make the contour more accurate.

Moving a Contour Point

The final small algorithm consists of creating a new contour when a contour's point that is already drawn is shift clicked and dragged to a new location. When this occurs, I implemented an algorithm that takes two energy terms to find the new locations for each contour point on either side of the point that was dragged and moved. I noticed when a group or bunch of points was off to the side, if you moved the middle point to a better spot, we mostly want the points on either side to follow. So, my first energy term was the used to minimize the distance between the points on either side and the moved point. The second energy term was the sobel image term, so that a point wouldn't move much if it was already on the edge we wanted it to be on. This algorithm ran for 3 iterations with a window size of 7x7 when activated. Both energy terms were weighted equally. A result is shown below.

Initial Contour with Displacement at Bottom

Contour After Shift Click Movement

The red points shown above are the ones that were moved. The middle red point was dragged from the bottom of the screen to the edge of the eggs, and as a result the point on its right moved away from its bunch and towards the moved point. The point on the left moved closer slightly towards the eggs. This contour isn't massive, but it is useful to fix small corners of a contour that may be misplaced.

Conclusion

In conclusion, this project taught me about the different types of active contours and when one, say balloon, may be useful over another, say rubber band. Going forwards, if I wanted to improve the contours, I could experiment with more models for the shift click model that would include moving more points near the point moved. For the balloon model, I could implement a model that has more calibration for expansion with larger versus smaller regions.