Snake Active Contours – Final Project

# Introduction

‘Snakes’ is an active contour based on parametric representation approach which minimizes the energy function by image forces from the gradient vector field. This force field acts on the contour such that the contours converge iteratively and eventually segment the object of interest. We implemented the Snakes in Matlab and reviewed the segmentation results for 4 images.

# Accomplishments

We are able to achieve very good performance in that the contours match the boundaries of the object very well for all the images.

We first computed the **magnitude of gradient** force field on the edge map, and thereafter utilized the snakeA4e to compute the inverse matrix A which is part of the iterative snake equation. We solve for the snake equation iteratively to get the contours to move closer to the object boundary with each iteration.

The numbers of iterations were chosen empirically. For Lena image, we needed to run for considerably larger number of iterations because it’s a more complex boundary. Additionally, in solving the snake equation, we also reset the contour points after 1000 iterations to evenly space them.

To get to these good results, we needed to continually vary the parameters in the snake model.

Finally, we got the edge map using no thresholding, and the 11σX11σ Gaussian with σ=5. The Gaussian helps to smooth edges and also remove any noise. We did not perform any other pre-processing. Thereafter, for the edge map, we used α as 0.5, β as 0, γ as 0.6, and the number of iterations as 2000 for 3 of the 4 images other than Lena. For Lena, we used the number of iterations as 20,000.

The corresponding python code was also ran to compare the performance, and the performance of our Matlab code was similar to the Python code.

# Improvements

The Lena image is quite complex with a lot of edges, but still we are able to get most of Lena’s face and hat in the contour. We could try and improve by varying the parameters, like using a smaller sized and deviation Gaussian filter, or even a superior filter like the Canny filter.

Alternatively, we could have used the **gradient vector flow** field in lieu of the magnitude of gradient field to get the edge map. Gonzalez textbook mentions that the gradient vector flow field converges in lesser number of iterations. Additionally, we could normalize the force fields Fx and Fy to see if the performance improves.

# Conclusions

The ‘snake’ algorithm can enclose simple boundaries with a rubber-band like stretch. It works well for objects with rounded boundaries which do not have sharp edges. But, this ‘snakes’ algorithm cannot detect multiple regions in an image. The magnitude of gradient force field works efficiently for simple objects but not for more complex objects like a human face with complex boundaries. Python code takes longer, with Lena image taking 6 hours. Comparatively, our Matlab code is very fast and gave better results than Python.