CMPT 732: Practices in visual computing I
Assignment 1 - part 1
Active contours

Total points: 30 + 2 points

Due: Wednesday, 12 October, 3 PM

Introduction

In the first part of the assignment, it is your task to implement the Snakes: Active Contour model by (1). It was already introduced in class, but the most important details will be reiterated. The method uses an energy-minimizing discrete curve, which adapts to a contour in an image. It is guided by internal forces that change the behavior of the curve and image forces to pull it towards features like lines and edges.

$$E^*_{snake} = \int_0^1 E_{snake}(v(s))ds = \int_0^1 E_{internal}(v(s))ds + \int_0^1 E_{external}(v(s))ds \tag{1}$$

Using an approximate discrete representation, the energy calculation can be simplified to:

$$E^*_{snake} = \sum_i^n E_{snake}(i) = \sum_i^n E_{internal}(i) + \sum_i^n E_{external}(i) \tag{2}$$

In this assignment, you will implement this method from scratch. A framework is provided along with this document, which will help you implement the method and run experiments on different images. In the following, different parts of the assignment and their respective grading are explained

Part 1: Initialization (2 pts)

In the `main.py` file, There is a code which opens an image and gives the user the ability to select the initial points of the contour. However, to increase accuracy, you should write a code that interpolates between the user-selected initial points with n evenly spaced fine points. Here, n is a parameter which can affect the quality of the final output.

Part 2: External energies (12 pts)

As we discussed in class, External energies are used to drive the contour towards the salient features of the image like edges, corners, and lines. For this part, you should complete the file `external_energy.py` which contains four functions:

1. E_{line}: This energy is equal to the intensity of the image.

2. E_{edge}: This energy pulls the contour toward the edges of the image and it is equal to the magnitude of the gradient of the image.

3. E_{term}: This energy pulls the contour toward corners and it is calculated as follows:

$$E_{term} = \frac{C_{xx}C_y^2 - 2C_{xy}C_xC_y + C_{yy}C_x^2}{(C_x^2 + C_y^2)^{\frac{3}{2}}} \tag{3}$$

   In this equation, C_x and C_y are the first order derivatives of the image in the x and y directions, and C_{xx}, C_{yy}, and C_{xy} are the second order derivatives of the image.

   bonus question (2 pts): prove that image curvature can be calculated with the equation above, and write it in your report.

4. $E_{external}$: The external energy is a weighted sum of all the above energies.

$$E_{external} = w_{line}E_{line} + w_{edge}E_{edge} + w_{term}E_{term} \tag{4}$$

   the weights specify the importance of each feature, and can vary based on application. The output of all the functions in this part are matrices in the size of the original image.

1 Part 3: Internal energy (8 pts)

The internal energy's job in the active contours' framework is to control the characteristics of the curve, by controlling the first and second derivatives of the curve. This control is further handled with two parameters α and β. The former, is related to the first order derivative of the curve and forces the curve to act like an elastic band. The latter, controls the smoothness of the curve and forces it to act like a metal sheet. As mentioned in the class, the optimization of the internal energy can be done using a pentadiagonal matrix A. In this part you should complete the file `internal_energy_matrix.py` which contains a function that takes α, β, and the changing step size γ, and returns the matrix $M = (A + \gamma I)^-1$, where, I is the identity matrix.

2 Part 4: Optimization loop (8 pts)

Back in the main.py file, you should finally complete the optimization loop. As discussed before, the optimization is done separately for x coordinates and y coordinates as follows:

$$x_t = M(\gamma x_{t-1} - \kappa f_x)$$
$$y_t = M(\gamma y_{t-1} - \kappa f_y)$$

Here, κ is an extra parameter that controls the importance of the external energy, compared to the internal energy. f_x and f_y are the gradients of the external energy in the x and y directions. Notice that because the external energy is computed on the image grid, it is only defined on integer coordinates. Thus, you have to do bilinear interpolation in order to get f_x and f_y. Furthermore, blurring the image at the beginning, might help with the quality of the output.

3 Report

For the demo session, please prepare parameters that work for each example image in the images folder. Additionally, you should submit a written report that includes:

1. visual results on the provided images. include at least the following scenarios:
   - For the binary images (circle, square, star, shape) and the vase, segment the objects
   - For the dental image, segment the row of teeth
   - For the brain image, segment the outer layer of the skull, the inner contour of the brain matter
   - The right eye hole
2. proof of the bonus question (optional).

References

[1] Michael Kass, Andrew Witkin, and Demetri Terzopoulos. Snakes: Active contour models. *International journal of computer vision*, page 1(4):321–331, 1988.