SIMON FRASER UNIVERSITY

Image Segmentation

Johannes Merz, Ali Mahdavi Amiri

September 17, 2019

1 Active Contour

In the first part of the assignment, it is your task to implement the Active Contour model Snakes by Kass et al. (1988) [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.

$$E^*_{Snake} = \int_0^1 E_{snake}(v(s))ds = \int_0^1 E_{int}(v(s)) + \int_0^1 E_{ext}(v(s))ds \tag{1.1}$$

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

$$E^*_{Snake} = \sum_{i=1}^{n} E_{int}(i) + \sum_{i=1}^{n} E_{ext}(i) \tag{1.2}$$

For this assignment, you are not supposed to implement the method from scratch, a MATLAB framework is provided that helps you with many details. The following steps will guide you through the rest of the implementation.

1.1 Initialization (2 Point)

A discrete representation of a spline is used, which is a vector of points (x, y) that gives a coarse description of the snake. Active contours is an iterative method and needs to be initialized with a snake. Depending on the initialization, the results can vary significantly. The framework already loads a test image for you. Implement the function *[x, y] = initializeSnake(I)*. It should display the image and give the user the possibility to choose the initial points of the snake. To increase the accuracy, interpolate between the points you selected with a fine spacing. (Hint: You can use the matlab function $spline$.) Finally, make sure that the snake forms a closed loop.

1.2 External Energies (12 Points)

As introduced before, the energy of the snake that needs to be minimized is the sum of the external and internal energy at each sample point. You are supposed to implement the function *[Eext] = getExternalEnergy(I,Wline,Wedge,Wterm,Sigma)* to precompute the external energies. The output is a matrix of the same size as the input image, which is given by

$$E_{ext} = w_{line}E_{line} + w_{edge}E_{edge} + w_{term}E_{term}. \tag{1.3}$$

The weights w are parameters that change the behavior of the snake. You need to calculate the three energy functionals.

- E_{line}: This energy is equal to the intensities of the image (the image itself).
  (2 Point)
- E_{edge}: An edge in an image is a contour with a large image gradient. Thus, this energy functional is given by $E_{edge} = -|\nabla I(x,y)|^2$. This means, you are looking for the negative gradient magnitude of the image. (Hint: Lookup the matlab function $imgradient$.)
  (2 Point)
- E_{term}: The curvature of the snake has to be taken into account in order to find corners. This is done by using the first and second order derivatives of the image. The energy functional then calculates as follows:

  $$E_{term} = \frac{C_{yy}C_x^2 - 2C_{xy}C_xC_y + C_{xx}C_y^2}{\left(1 + C_x^2 + C_y^2\right)^{\frac{3}{2}}} \tag{1.4}$$

  (Hint: You can use $conv2$ with appropriate masks to calculate the derivatives C, e.g. using finite forward differences. Also, the equation above varies from the referenced paper for stability reasons. Please use the one provided here.)
  (8 Points)

1.3 Internal Energies and Iteration (6 Points)

In addition to the external energies, the internal energies of the snake determine the behavior of the snake. They are calculated using Euler equations and formulated in a pentadiagonal matrix. This step is already implemented in the function *[Ainv] = getInternalEnergyMatrix(nrPoints, alpha, beta, gamma)* of the framework for you. Here, the parameters α and β

control the behavior of the snake. While α makes the spline act like a flexible membrane, increasing β lets it behave more like a thin plate. It is your task to use your calculated external energies and the given matrix A^{-1} to update the positions of the snake. Do this by implementing the function *[points] = iterate(Ainv, points, Eext, gamma, kappa)*. In this function, the contour points should be updated according to the following formulas:

$$\begin{aligned} x_t &= A^{-1}\left(\gamma x_{t-1} - \kappa f_x(x_{t-1}, y_{t-1})\right) \\ y_t &= A^{-1}\left(\gamma y_{t-1} - \kappa f_y(x_{t-1}, y_{t-1})\right) \end{aligned} \tag{1.5}$$

The parameter γ is the step size and κ regulates the influence of the external forces f_x and f_y. The external forces can be calculated from the external energies E_{ext} as their gradients. Note that x and y are float values, while f_x and f_y are matrices of the same size as the image, thus only defined at integer pixel values. To evaluate them at float values, you have to use bilinear interpolation.

1.4 Bonus: Internal Energy Matrix (5 Points)

References

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