Tutorial: Active contours

Note

This tutorials aims at introducing the active contours (a.k.a. snakes) method as originally presented in [1].

1 Definition

A snake is a parametric curve $v(s)$ with $s \in [0;1[$. The energy functional is represented by:

$$E_{snake} = \int_0^1 E_{\text{int}}(v(s))\mathrm{d}s + \int_0^1 E_{\text{ext}}(v(s))\mathrm{d}s.$$

The internal energy is detailed in Eq.1. The first order derivation constrains the length of the curve, the second order derivation constrains the curvature. The parameters α and β a priori depend on s, but for simplicity, constant values will be taken.

$$E_{\text{int}}(v(s)) = \frac{1}{2}\left(\underbrace{\alpha(s)|v'(s)|^2}_{length} + \underbrace{\beta(s)|v''(s)|^2}_{curvature}\right) \tag{1}$$

The snake will be attracted by some edges points from the external energy (the image to be segmented). The, the energy functional will be in a local minimum: it is shown that the Euler-Lagrange equation is satisfied:

$$-\alpha v^{(2)} + \beta v^{(4)} + \nabla E_{\text{ext}} = 0$$

with $v^{(2)}$ and $v^{(4)}$ denoting the 2nd and 4th order derivatives. To solve this equation, the gradient descent method is employed: the snake is now transformed into a function of the position s and the time t, with $F_{\text{ext}} = -\nabla E_{\text{ext}}$

$$\frac{\partial s}{\partial t} = \alpha v^{(2)} - \beta v^{(4)} + F_{\text{ext}}$$

2 Numerical resolution

The spatial derivatives are approximated with the finite difference method, and the snake is now composed of n points, $i \in [1;n]$:

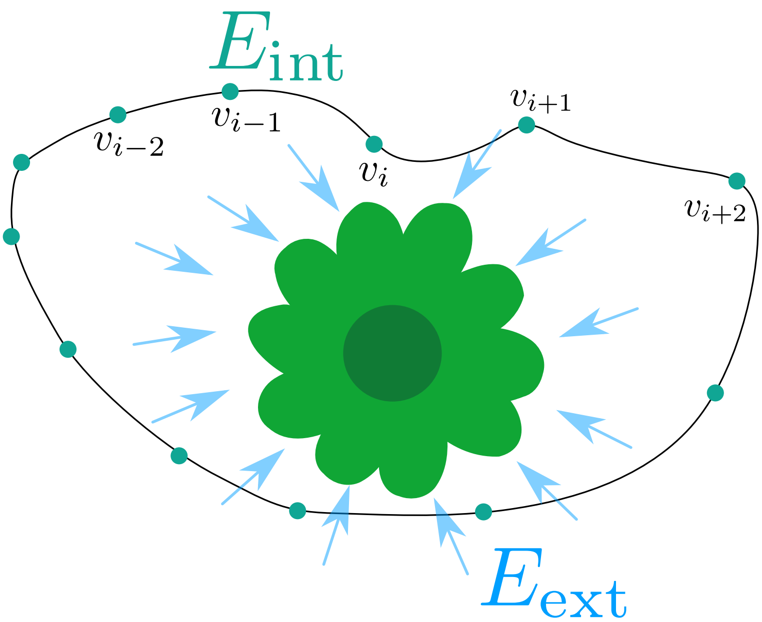

Figure 1: Illustration of the active contours segmentation method. Two energies are at stake: internal energies depend only on the snake shape and control points, external energies are related to the image properties.

$$
\begin{aligned}
x^{(2)} &= x_{i-1} - 2x_i + x_{i+1} \\
x^{(4)} &= x_{i-2} - 4x_{i-1} + 6x_i - 4x_{i+1} + x_{i+2}
\end{aligned}
$$

The gradient descent method can be written as, with γ being the time step and controls the convergence speed:

$$
\begin{aligned}
\frac{x_t - x_{t-1}}{\gamma} &= A \cdot x_t + f_x(x_{t-1}, y_{t-1}) \\
\frac{y_t - y_{t-1}}{\gamma} &= A \cdot y_t + f_y(x_{t-1}, y_{t-1})
\end{aligned}
$$

with

$$
A = \begin{bmatrix}
-2\alpha-6\beta & \alpha+4\beta & -\beta & 0 & \cdots & 0 & -\beta & \alpha+4\beta \\
\alpha+4\beta & -2\alpha-6\beta & \alpha+4\beta & -\beta & 0 & \cdots & 0 & -\beta \\
-\beta & \alpha+4\beta & -2\alpha-6\beta & \alpha+4\beta & -\beta & 0 & \cdots & 0 \\
0 & -\beta & \alpha+4\beta & -2\alpha-6\beta & \alpha+4\beta & -\beta & 0 & \cdots \\
\vdots & & \ddots & \ddots & \ddots & \ddots & & \vdots \\
0 & \cdots & -\beta & \alpha+4\beta & -2\alpha-6\beta & \alpha+4\beta & -\beta & 0 \\
0 & \cdots & 0 & -\beta & \alpha+4\beta & -2\alpha-6\beta & \alpha+4\beta & -\beta \\
-\beta & 0 & \cdots & 0 & -\beta & \alpha+4\beta & -2\alpha-6\beta & \alpha+4\beta \\
\alpha+4\beta & -\beta & 0 & \cdots & 0 & -\beta & \alpha+4\beta & -2\alpha-6\beta
\end{bmatrix}
$$

The forces f_x and f_y are the components of F_{ext}. For example for an image I, if $*$ denotes the convolution and G_σ a gaussian kernel of standard deviation σ:

$$F_{\text{ext}} = -\nabla(\|\nabla(G_\sigma * I)\|)$$

- Generate a binary image (of size $n \times n$ pixels, with values 0 and 1) containing a disk (of a radius R).
- Generate the initial contour as an ellipse, with the same center as the disk.
- Generate the pentadiagonal matrix A. The different parameters can be $\alpha = 10^{-5}$ and $\beta = 0.05$.
- Program the iterations and visualize the results. For example, $\gamma = 200$ and 1000 iterations may give an idea of the parameters to use.

References

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