

Segmentation and image analysis - Practical work 3

Segmentation based on active contours

1 2D active contours

Active contours (or *snakes*) are an image segmentation technique based on the boundaries of the objects. A contour is initialized in the image near the shape to be extracted. It is then deformed according to the following constraints : resistance to stretching, resistance to bending, edges in the image.

You have to implement the basic 2D snake algorithm by computing the successive locations of an active contour until it fits the contour of the wanted object.

A "snake" (parametric curve $v(s) = (x(s), y(s)), s \in [0; 1]$) tries to minimize its energy which is defined by :

$$E = E_{internal} + E_{image}$$

$$\text{with } E_{internal} = \int_0^1 \alpha \left| \frac{\partial v(s)}{\partial s} \right|^2 + \beta \left| \frac{\partial^2 v(s)}{\partial s^2} \right|^2 ds$$

$$\text{and } E_{image} = \lambda \int_0^1 -|\nabla I(v(s))|^2 ds$$

The parameters α , β and λ allow to weight the effects of the two internal energies which are linked to stretching and to curvature and of the external energy which is linked to the image gradient. Here the snake will be digitized as a polygonal curve, i.e. a list of N points defined by their coordinates (arrays **X** and **Y** of the class **Snake**). The energy minimization is then reformulated as partial differential equations :

$$-\alpha x''(s) + \beta x^{(4)}(s) = \lambda \frac{\partial |\nabla I(v)|^2}{\partial x}(x(s), y(s)) \tag{1}$$

$$-\alpha y''(s) + \beta y^{(4)}(s) = \lambda \frac{\partial |\nabla I(v)|^2}{\partial y}(x(s), y(s)) \tag{2}$$

The derivatives are approximated with finite differences so as to write (1) and (2) as the product of a matrix A with **X** or **Y** :

$$AX = \lambda f_x(X, Y)$$

$$AY = \lambda f_y(X, Y)$$

The matrix A is the following :

$$\begin{bmatrix} 6\beta + 2\alpha & -4\beta - \alpha & \beta & 0 & \dots \\ -4\beta - \alpha & 6\beta + 2\alpha & -4\beta - \alpha & \ddots & \ddots \\ \beta & -4\beta - \alpha & 6\beta + 2\alpha & \ddots & \ddots \\ 0 & \ddots & \ddots & \ddots & \ddots \\ \vdots & \ddots & \ddots & \ddots & \ddots \end{bmatrix}.$$

$f_x(X, Y)$ (resp. $f_y(X, Y)$) is a vector composed of N values representing the horizontal (resp. vertical) component of the image forces applied to the snake vertices. This system being not

invertible, the solution is computed iteratively thanks to the following formulation (γ is a friction coefficient) :

$$\begin{aligned} X_t &= (A + \gamma I)^{-1}(\gamma X_{t-1} + \lambda f_x(X_{t-1}, Y_{t-1})) \\ Y_t &= (A + \gamma I)^{-1}(\gamma Y_{t-1} + \lambda f_y(X_{t-1}, Y_{t-1})) \end{aligned}$$

A starting point to implement this segmentation method is given in the archive (<http://dept-info.labri.fr/~vialard/Image3D/src/files3.tgz>) which contains :

- a directory `images` with test images,
- a directory `src` with the source file of the `ImageJ` plugin `Snake_Viewer.java`. It uses the class `Snake` (to be completed). The image gradient is computed with the Deriche filter (see `Deriche.java`).

How to compute the deformation

Fill in the method `initParameters` of the class `Snake` by adding the computation of the matrix $M = (A + \gamma I)^{-1}$. The linear algebra package `Jama` allows to define matrices of real numbers, to apply operations such as addition and inversion. This package and its documentation can be found at <http://math.nist.gov/javanumerics/jama/>.

Write the method `iteration` of the class `Snake`, i.e. translate the following formulas :

$$\begin{aligned} X_t &= (A + \gamma I)^{-1}(\gamma X_{t-1} + \lambda f_x(X_{t-1}, Y_{t-1})) \\ Y_t &= (A + \gamma I)^{-1}(\gamma Y_{t-1} + \lambda f_y(X_{t-1}, Y_{t-1})) \end{aligned}$$

The image forces f_x and f_y at a point of the snake are obtained by differentiating the squared gradient norm of the image with respect to x and to y . These partial derivatives are computed by finite differences. In practice, you will have to avoid too important movements of points : the images forces will be limited to the range $[-1.0 : 1.0]$.

Tests

If the snake evolution is computed in an image with a uniform color, what is the value of the image energy ? What is the final shape of the snake ? Why ?

Test your algorithm on the image `empty.png` with the parameters $\alpha = 0.01$, $\beta = 0.5$, $\gamma = 2$. Vary progressively these parameters to understand their influence.

Test your algorithm on the other images by carefully choosing the parameters α , β , λ and γ .

2 3D active contours : ITK-Snap

`ITK-Snap` is a software dedicated to the interactive segmentation of medical images. The user can carry out a manual segmentation by following the object contours slice by slice (label edition, edition of a polygonal contour). He can also segment an organ in the 3D image by using active surfaces.

Download `ITK-Snap` from <http://www.itksnap.org/> (Downloads / Linux Binary). Also download the test images `MRICrop-orig.gipl` and `MRICrop-seg.gipl` (archive `MRICrop.zip` - 3D brain MRI).

Visualization of a 3D image / associated segmentation

Open the image `MRICrop-orig.gipl`. The interface allows to visualize the image according to 3 orthogonal slices. Test the tools for exploring the image : browse the slices, zoom. Test the contrast tuning (Tools/ImageContrast).

Open the image `MRICrop-seg.gipl` which contains a partial labeling of the previous image (Segmentation / Open Segmentation). Visualize the segmented structures in 3D (Update button in the 4th window). Edit the labels so as to see only the ventricle which corresponds to the red label (Segmentation/Label Editor)..

Semi-automatic segmentation based on the image edges

The aim of this part is to obtain a segmentation of the ventricle in the image `MRICrop-orig.gipl` by using active surfaces which will be stopped by the image edges. The successive steps are described in section 7 of the ITK-snap tutorial (<http://www.itksnap.org/pmwiki/pmwiki.php?n=Documentation.TutorialSectionEdgeBasedSegmentation>).

The segmentation is started by the 5th tool of the Main Tool Bar. Choose *Edge Attraction* for the presegmentation step. You can first define the parameters of the function g_I which is based on a gradient computation to take into account the image data in the surface evolution. This function has a near 0 value on the edges and a near 1 value in the uniform zones. You have then to initialize the surface by a set of spheres. Finally you have to set the evolution parameters called α , β and γ in the mathematical mode.

The user interface gives the evolution equation, which corresponds to the level-set method. To understand it in details refer to the article by Yushkevich published in NeuroImage in 2006 (given in files3.tgz). Precise the role of each parameter. How is exactly defined g_I ?

Endocranium segmentation

Use the same process to segment the endocranium (surface of the bone inside the head) in the image `head.img`.