

Computer Aided Medial Procedures II

Exercise on Registration based on Free-form Deformations

14.07.2011 - Loren Schwarz, Nassir Navab

In this exercise, you will implement a simple deformable registration algorithm for 2D images based on B-spline free-form deformations (FFDs). To get started easily, you are provided with some code skeletons as described below.

1. Open the Matlab file `spline_reg.m`. This is the main program skeleton for your registration algorithm. Familiarize yourself with the structure of the file and read the comments. In particular, pay attention to how control points and images are represented in the code.

Also take a look at the function `spline_basis.m` that you can use in your code. It implements the cubic B-spline basis functions.

2. Write a Matlab function `grad_energy` with the following signature (the function stub is already given - read the comments):

```
function [g, diff] = grad_energy(phi, R, T, d_ctrl, n_ctrl)
```

The function is supposed to compute the gradient of the registration energy E . To simplify the exercise, please only use the SSD-based dissimilarity term and disregard the regularization. You can either implement a finite difference approximation of the gradient, or the analytical derivative.

The returned vector g should have $2d$ entries, where d is the total number of control points. Entry i of the vector should represent the partial derivative of E with respect to the i -th control point - in the x-direction. Entry $i+d$ of the vector should contain the corresponding y component.

3. Write a Matlab function `get_displacement` with the following signature:

```
function [u_x, u_y] = get_displacement(phi, d_ctrl, n_ctrl, n_img)
```

The function is supposed to compute a dense displacement field (i.e., its two components u_x and u_y for each of the two dimensions), based on a given control point configuration ϕ .

4. Play with the parameters of the algorithm, and in particular with the step size parameter `lambda` and the control point grid spacing `d_ctrl`. Try to achieve the best registration of the two images (in terms of dissimilarity).
5. Currently, the iterations of the algorithm are simply governed by the number of iterations, `n_iter`. Think of some other, better, convergence criteria and try to implement them.