Variational Methods for Computer Vision: Exercise Sheet 8

Exercise: 9 January 2014

Part I: Theory

The following exercises should be **solved at home**. You do not have to hand in your solutions, however, writing it down will help you present your answer during the tutorials.

1. Let $I(x) : \Omega \to \mathbb{R}$ with $\Omega \subset \mathbb{R}^2$ be an image. Consider this generalized version of the Chan-Vese functional:

$$E(\phi) = \int_\Omega f_1(x) H(\phi(x)) dx + \int_\Omega f_2(x)(1 - H(\phi(x))) dx + \nu \int_\Omega |\nabla H(\phi(x))| dx \quad (1)$$

Where f_i is a data term which arises from a general gray value distribution p_i hence:

$$f_i(x) = -\log p_i(I(x))$$

$H(\phi(x))$ denotes the Heaviside step function:

$$H(\phi(x)) = \begin{cases} 1 & \text{if} \quad \phi(x) > 1 \\ 0 & \text{else} \end{cases}$$

Prove that the Euler-Lagrange equation of (2) can be written as follows:

$$\frac{dE}{d\phi} = \delta(\phi) \left[f_1 - f_2 - \nu \, \text{div} \left(\frac{\nabla \phi}{|\nabla \phi|} \right) \right] = 0$$

The delta distribution $\delta(\phi)$ can be considered as the derivative of H.

2. The level set formulation of geodesic active contours (Chapter 8, Slide 15) can be formulated as the following functional:

$$E(\phi) = \int_\Omega g(x) |\nabla H(\phi(x))| dx \quad (2)$$

where $H(\phi(x))$ is defined as in exercise 1. Compute the Euler-Lagrange equation of $E(\phi)$.

Part II: Practical Exercises

This exercise is to be solved **during the tutorial**.

In this practical exercise we are going to consider the the Chan-Vese functional :

$$E(\phi) = \int_\Omega (I(x) - \mu_1)^2 H(\phi(x))dx + \int_\Omega (I(x) - \mu_2)^2 (1 - H(\phi(x)))dx + \nu \int_\Omega |\nabla H(\phi(x))| dx \quad (3)$$

1. Download the archive file `vmcv_ex08.zip` from the homepage and unzip it in you home folder and complete the code in `chan_vese.m` as follows:
   (a) In order to minimize above functional an optimal μ_1 and μ_2 have to be obtained. For a given curve the optimal values for μ_1 and μ_2 are the mean values of the inner and outer region. Implement a function `[mu1,mu2]=approxRegions(Phi,I)` that returns for a given image I and a level set function Phi the mean values inside and outside the contour.
   (b) Further implement a function `dPhi=update(Phi,I)` which computes a gradient descent direction using the the result of exercise 3.
   (c) Implement an energy minimization of (4). Initialize the level set function with a circle of radius R in the center of the image.
2. Test your implementation on the image `image.png` with various Radii R.