Variational Methods for Computer Vision: Exercise Sheet 6

Exercise: 9 December 2011

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 $u \in C^2(\Omega; \mathbb{R})$ be a twice differentiable real-valued function and $\Omega \subset \mathbb{R}^2$. And let

$$E(u) = \int_\Omega \mathcal{L}(u(x), \nabla u, Hu(x)) \, \mathrm{dx}$$

be a real valued Gâteaux differentiable functional which depends on:

$$u(x) \quad , \quad \nabla(u) = \left(\frac{\partial u}{\partial x}, \frac{\partial u}{\partial y}\right)^T \quad , \quad \text{and} \quad (Hu(x))_{i,j} = \frac{\partial^2}{\partial x_i \partial x_j} u(x) \quad .$$

The 2×2 matrix $Hu(x)$ denotes the Hessian of u. Calculate the Gâteaux derivative of $E(u)$.

2. Calculate the Euler-Lagrange equation of the following energy functional:

$$E(u) = \frac{1}{2\lambda} \int_\Omega (h * u - f)^2 + \int_\Omega |\nabla u| \, \mathrm{dx}$$

Where $\Omega \subset \mathbb{R}^2$ represents the image domain, $u : \Omega \to \mathbb{R}$ denotes the optimization variable, $f : \Omega \to \mathbb{R}$ stands for the input image and h denotes a convolution kernel (not necessarily symmetrical).

Part II: Practical Exercises

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

1. In the lecture we encountered the concept of super resolution from video. The key idea of super resolution is to exploit redundancy available in multiple frames of a video. Assuming that each input frame is a blurred and downsampled version of a higher resolved image u the high-resolution image can be recovered as the minimum of the following energy funcitonal:

$$E_(u) = \sum_{i=1}^{n} \int_\Omega (ABW_i u(x) - Uf_i)^2 \, \mathrm{dx} + \lambda \int_\Omega |\nabla u(x)| \, \mathrm{dx} \tag{1}$$

The Linear Operator B denotes a Gaussian Blurring. The up-sampling operator U simply replaces every pixel with four pixels of the same color. In order to be able to compare image u with the up-sampled version of f_i which is constant block-wise we apply the linear averaging operator A on u which assigns every block of pixels the mean values of the pixels in that block. The linear operator W_i accounts for the coordinate shift by motion w_i hence:

$$W_i u = u(x + w_i(x))$$

(a) In the following we are going to construct a toy example for super resolution by executing the following steps:

i. Download the archive vmcv_ex06.zip and unzip it on your home folder. In there should be a file named `scene.jpg`.

ii. Create from the unzipped image image 6 versions shifted in x direction by exactly one pixel hence:

$$f_i = f(x + i, y)$$

for $i = 1...6$. In order to account for the boundary consider taking cropped images from the interior of the original image.

iii. In order to simulate blurring convolve the shifted images with a Gaussian Kernel. Next down-sample the images f_i by factor 2 by using the `imresize` function in Matlab with nearest neighbor interpolation.

(b) In what follows we are going to minimize the above functional in order to obtain a super resolved image from our input images f_i.

i. Compute the matrix representations of the linear operators A, B , W and U. Since these matrices are huge , consider using sparse data structure in Matlab (`spdiags speye`) in order to obtain a sparse representation.

ii. Compute $\operatorname{argmin}_u E(u)$ by means of Gradient descent using matrix vector representation after stacking the function u in a vector using the matlab command `reshape`.