Prof. Dr.-Ing. A. Bruhn
Institute for Visualization and Interactive Systems
Department Intelligent Systems
University of Stuttgart

Homework Assignment 3

$F(x, u, u')$

H 3.1 (Isotropic flow-driven optical flow) — 12 Points

$u = v + \varepsilon h$

$\frac{d}{d\varepsilon}\int F(x, v+\varepsilon h, v' + \varepsilon h')\,dx$

Consider the following energy functional for optical flow computation

$= \int \frac{\partial F}{\partial x}\frac{\partial x}{\partial \varepsilon} + \frac{\partial F}{\partial u}\frac{\partial u}{\partial \varepsilon} + \frac{\partial F}{\partial u'}\frac{\partial u'}{\partial \varepsilon}\,dx = 0$ (0, h, h')

$F_v - \frac{\partial F_{vx}}{\partial x} - \frac{\partial F_{vy}}{\partial y} = 0$

$F_u - \frac{\partial F_{ux}}{\partial x} - \frac{\partial F_{uy}}{\partial y} = 0$

$$E(u, v) = \int_\Omega (f_x u + f_y v + f_t)^2 + \alpha\Psi\left(|\nabla u|^2 + |\nabla v|^2\right)\,dx\,dy \qquad (1)$$

$= \int F_u \cdot h + F_{u'} \cdot h'\,dx = 0$

$(uv)' = u'v + uv'$

(a) Compute the Euler-Lagrange equations.

$= \int F_u \cdot h - \frac{\partial F_{u'}}{\partial x}\cdot h\,dx + F_{u'} h\Big|_a^b$

(b) Compute an analytical expression for the arising derivative $\Psi'(s^2)$ for

$= \int \left(F_u - \frac{\partial F_{u'}}{\partial x}\right) h\,dx + F_{u'} h\Big|_a^b \overset{!}{=} 0$

$$\Psi(s^2) = \lambda^2 \log\left(1 + \frac{s^2}{\lambda^2}\right)$$

$h(a) = h(b) = 0$

$F_u - \frac{\partial F_{u'}}{\partial x} = 0.$

(c) What is the relation to the isotropic nonlinear diffusion studied in Lecture 13?

$F_{u'} \cdot h = 0$ $F_{u'} = 0$ $F_{u'} \cdot = 0$

(d) What is the effect of the function Ψ considering flow edges?

$\vec{F} \cdot \vec{n} = 0.$

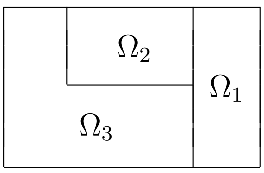
(e) How would the Euler-Lagrange equations change, if Ψ was applied to the data term instead?

$\begin{pmatrix} F_{ux} \\ F_{uy} \end{pmatrix}^T \cdot n = 0$

(f) What could be the impact of this modification?

H 3.2 (Segmentation) — 8 Points

Let the following 4×4 piecewise constant image with three regions be given:



$\Leftrightarrow$

14	6	6	2
14	6	6	2
14	14	14	2
14	14	14	2

where the size of each pixel is 1×1. Use the algorithm from Lecture 15 that approximates the solution of the *Mumford-Shah Cartoon model* for different scale parameters λ by successively increasing λ and merging adjacent regions. Specify all merging events, the corresponding scale parameters as well as the corresponding segmentations.

P 3.1 (Coherence-Enhancing Diffusion Filtering)

Please download the required file `cv20_ex03.tgz` from ILIAS. To unpack the data, use `tar xvfz cv20_ex03.tgz` .

(a) Supplement the file `diff_tensor.c` with the missing code. You may use the included routines for principle axis transformation and backtransformation. Compile the programme with

```
gcc -O3 -o ced ced32.o diff_tensor.c -lm  (on 32-bit machines),
gcc -O3 -o ced ced64.o diff_tensor.c -lm  (on 64-bit machines).
```

(b) Use the programme `ced` for enhancing the fingerprint image `finger_scaled.pgm` with the paramters $C = 1$, $\sigma = 0.5$, $\rho = 4$, $\alpha = 0.001$, $\tau = 0.2$, 40 iterations. You will observe that the extremum principle is violated by the standard discretisation that is used in this algorithm.

(c) Use `ced` for creating your own Christmas postcards. Its easy: just take `xmas.pgm` and filter it with the same parameters as for the fingerprint.

(d) Use `ced` to visualise all stripes of `fabric.pgm` at different scales. Use the standard parameters and increase the number of iterations.

P 3.2 (PDE-based morphology - Dilation and Erosion)

(a) In the file `pde_morphology.c`, supplement the routines `dilation_point` and `erosion_point` with the missing code such that they implement one iteration step for the pixel (i, j).

```
gcc -O3 -o pde_morphology pde_morphology.c -lm
```

(b) Use the program `pde_morphology` to perform dilation and erosion on the images `head.pgm` and `bank.pgm`.

(c) Why is it not possible to reconstruct the results from the lecture for `bank.pgm` with this implementation?

Remark: The program will be able to handle both color and gray images.

P 3.3 (PDE-based morphology - Shock Filter)

(a) In the same code file, supplement the routines `hessian` and `shock_filter` such that an implementation of the shock-filter is created. Since the program should be able to handle color images as well as gray value images, we extend our theoretical knowledge:

For color images, one could perform coherence enhancing shock filtering for each channel seperately. However, this would create shocks at different locations in each channel. Thus, Weickert `[http://www.mia.uni-saarland.de/Publications/weickert-dagm03.pdf]` proposed to synchronize the processes as follows:

i) Compute the joint structure tensor

$$J_{f_1,f_2,f_3} = \sum_{c=1}^{3} J_{f_c} = \sum_{c=1}^{3} K_\rho * \nabla f_c \nabla f_c^\top$$

ii) Compute the dominant direction η as dominant eigenvector of J_ρ

iii) Average second order derivatives in η direction: $v_{\eta\eta} = \frac{1}{3}\sum_{c=1}^{3} v_{c\eta\eta}$

iv) Evolve the channels according to

$$\partial_t u_c = -\mathrm{sgn}(v_{\eta\eta})\,|\nabla u_c| \quad c = 1, 2, 3$$

Remark: We are interested in the sign of $v_{\eta\eta}$, so the factor $\frac{1}{3}$ is neglectible.
Remark: The second order derivatives should be computed using the Hessian matrix:

$$v_{c\eta\eta} = \eta^\top H_{v_c}\eta; \qquad H_{v_c} = \begin{pmatrix} \partial_{xx}v_c & \partial_{xy}v_c \\ \partial_{xy}v_c & \partial_{yy}v_c \end{pmatrix}$$

Remark: Instead of averaging derivatives, one may average the Hessians (linearity).

(b) Check your implementation with the color image `baboon.ppm` and the gray image `finger.pgm`. You should be able to reproduce the results from the lecture slides.

(c) Take a photo of yourself and perform shock filtering with parameters at your taste. Most image formats can be converted to `ppm` using `convert image.xyz image.ppm`

Submission:

The theoretical problem(s) have to be submitted in handwritten form before the deadline.

Deadline for Submission is: Friday, January 25th,11:59 pm

Prof. Dr.-Ing. A. Bruhn
Institute for Visualization and Interactive Systems
Department Intelligent Systems
University of Stuttgart

Classroom Assignment 3

C 3.1 (Mumford-Shah Cartoon Model)

Let $\Omega_i, \Omega_j \subset \Omega$ denote two segments with mean u_i resp. u_j. Furthermore, let $\partial(\Omega_i, \Omega_j)$ denote the common boundary between Ω_i and Ω_j.

Show that for the Mumford-Shah cartoon model, merging these two regions results in the following change of energy:

$$E(K \setminus \partial(\Omega_i, \Omega_j)) - E(K) = \frac{|\Omega_i| \cdot |\Omega_j|}{|\Omega_i| + |\Omega_j|} \cdot (u_i - u_j)^2 - \lambda \, l(\partial(\Omega_i, \Omega_j)) \ .$$