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

Assignment C4 – Classroom Work

Problem 1 (Nonlinear Diffusion)

In the explicit scheme for 1-D nonlinear diffusion filtering, the new value u_i^{k+1} at pixel i can be computed from the known values of the old time step k via

$$u_i^{k+1} = u_i^k + \frac{\tau}{h}\left(g_{i+1/2}^k \frac{u_{i+1}^k - u_i^k}{h} - g_{i-1/2}^k \frac{u_i^k - u_{i-1}^k}{h} \right)$$

where τ is the time step size, and $g_{i+1/2}^k$ approximates the diffusivity in the intermediate location $i + 1/2$ at time level k.

(a) How does the corresponding stencil look like?

(b) For the whole signal, this iteration step can be formulated as a single matrix-vector product

$$\boldsymbol{u}^{k+1} = Q(\boldsymbol{u}^k)\, \boldsymbol{u}^k \;.$$

Here, $\boldsymbol{u}^k$ and $\boldsymbol{u}^{k+1}$ are vectors of size N and $Q(\boldsymbol{u}^k)$ is an $N \times N$ matrix. Which structure has the matrix $Q(\boldsymbol{u}^k)$? How it is related to the previous stencil?
Hint: You can assume that the signal is reflected at the boundaries, i.e. one may use dummy values $u_0^k := u_1^k$ and $u_{N+1}^k := u_N^k$.

(c) Compute the row and column sums of $Q(\boldsymbol{u}^k)$. What are your findings considering the rules proven in Problem 1(a) and 1(c)?

(*This assignment connects the stencil notation for a single pixel to the overall explicit step.*)

Assignment H4 – Homework

Problem 1 (Hard Wavelet Shrinkage) (6 points)

Compute the discrete Haar wavelet transform of the noisy step signal

$$\boldsymbol{f} := (5, 4, 5, 6, 7, 6, 5, 4)^\top \ .$$

Perform hard wavelet shrinkage with $T = \frac{1}{\sqrt{2}}$, and compute the inverse wavelet transform.

Problem 2 (Continuous Nonquadratic Variational Methods) (6 points)

Let $f(x)$ be a noisy 1-D signal on the interval $[a, b]$. For discontinuity-preserving denoising, we seek a twice continuous differentiable function $u(x)$ that minimises the energy functional

$$E(u) \ := \int_a^b \left(\frac{1}{2} (u - f)^2 + \alpha \lambda^2 \sqrt{1 + u_x^2 / \lambda^2} \right) dx \ .$$

(a) Write down the Euler-Lagrange equations of this energy functional.

(b) What is the meaning of the parameter λ?

Problem 3 (Discrete Variational Methods) (6 points)

(a) Write down a discrete analogue to the energy functional from Problem 1.

(b) Which nonlinear system of equations has to be satisfied necessarily in its minimum?

Problem 4 (NL-Means) (6 points)

Please download the required files from ILIAS into your own directory. You can unpack them with the command `tar xvzf is20_ex04.tgz`.

The image `barbara-cr-n20.pgm` has been created by adding Gaussian noise of standard deviation 20.

(a) The programme `nlm` is an implementation of the NL-means filter. There are three parameters to choose:

- The patch radius defines the size of the disk-shaped patch.
- The search-window-size gives the size of the restricted, disk-shaped space around pixel i.
- The filter parameter σ is used in the weight function g defined as $g(x) = \exp\left(\frac{-x^2}{2\sigma^2}\right)$

Denoise `barbara-cr-n20.pgm` using the programme `nlm`.

Hint: σ^2 should be chosen to be roughly 5 times the noise standard deviation.

(b) Now try to denoise `barbara-cr-n20.pgm` by a Gaussian convolution using the precompiled programme `gauss_conv`.

(c) So-called method noise is given by the difference between noisy image and filtered version. The programme `difference-shifted`, calculates the difference between two images and shifts the result by 127.5. Apply it to the original image and the filtered results obtained in (a) and (b).

What do you observe? Why is the result shifted by 127.5? How should the method noise look like in the optimal case? Justify your answers.

Problem 5 (Deconvolution with Wiener Filtering) (12 points)

The code `deconv.c` is an almost complete deblurring programme based on manipulations in the Fourier domain. It only requires to specify the deblurring function in the subroutine `deconv-filter.c`.

(a) Supplement the missing code for Wiener filtering. However, do not use C/C++ libraries such as `complex.h` that offer computations with complex data types. You can compile the programme via

```
gcc -O2 -o deconv deconv.c -lm .
```

(b) The image `bus1.pgm` has been blurred with the small Gaussian `k-gauss1.pgm`. Use the compiled programme to perform a deblurring. What is a good value for K?

(c) Do the same experiment with the image `bus2.pgm` and its corresponding blurring kernel `k-gauss2.pgm`. What is a good value of K in this case?

(d) Finally, consider the image `hogblur.pgm` which is a digital photo with simulated motion blur corresponding to the kernel `k-motion.pgm`. Try to compensate for the motion blur by finding a suitable value for K. Why does deblurring take so long in the case of this image?

Submission

Please remember that up to three people from the same tutorial group can work and submit their results together. Note that in order to submit results as a group you have to create a submission group in ILIAS. A submission result will only be accepted for the submitting group as created in ILIAS!

The solutions have to be submitted in two parts:

1. The solutions to the theoretical problems 1 – 3 have to be submitted in a single `pdf` file, which can be either digitally created or contain a scanned document.
2. The solutions to the practical problems 4 and 5 has to be submitted in a single archive file:

- Rename the main directory `Ex04` to `Ex04_<your_name>` and use the command

  ```
  tar czvf Ex04_<your_name>.tgz Ex04_<your_name>
  ```

  to pack the data. The directory that you pack and submit should contain the following files:

  - One reasonably denoised image of problem 4(a);
  - One reasonably denoised image of problem 4(b);
  - The method noise images of problem 4(c);
  - the source code for the Wiener filter in `deconv-filter.c`.
  - the three deblurred images from the tasks 5(b), (c), and (d)
  - a text file README that
    * states the parameters which are used for problem 4(a) and (b),
    * contains the answers to the questions of problem 4(c),
    * states the used values for the parameter K in the tasks 5(b), (c), and (d),
    * answers the question w.r.t. the runtime in task 5(d),
    * contains information on all people working together for this assignment.

- The file format can be a gzipped tar archive (`.tgz`) or a zip archive (`.zip`). No other file formats are accepted.
- Please make sure that only your final version of the programmes and images are included.

Submit the two files via ILIAS.
(Remark: Please do **not** use the button “Upload Multiple Files as Zip-Archive”).

Deadline for submission: Friday, July 24th, 23:59