

Solve before: June 13, 2006

Remo Ziegler, Christian Voegeli, Daniel Cotting,
Christian Sigg, Jens Keuchel

Visual Computing (Non-)linear Diffusion

General Remarks

It is not necessary to hand in your results. You will find an exemplary solution on the lecture's web page.

Basics

For two dimensional diffusion processes, *Fick's law* states that a concentration gradient $\nabla u = (\partial_x u, \partial_y u)^\top$ (e.g. of mass/temperature) causes a flux $j = (j_x, j_y)^\top$ which compensates for this gradient ('equilibration of concentration differences'):

$$j = -\mathbf{D} \cdot \nabla u .$$

The relation between ∇u and j depends on some characteristics of the material and is specified by the positive diffusivity \mathbf{D} , which may vary depending on the local structure. In the following, we will only consider the so-called *isotropic* case by assuming that j and ∇u are parallel, which results in a scalar-valued diffusivity $\mathbf{D} \in \mathbb{R}^+$. (In the more general *anisotropic* case \mathbf{D} is a matrix.)

The flux results in a temporal change (the *diffusion*) of the concentration u during which no mass is destroyed or created, which is expressed by the *continuity equation*:

$$\partial_t u = -\operatorname{div} j = -(\partial_x j_x + \partial_y j_y) ,$$

where t denotes time. Combining both equations gives the important *diffusion equation*:

$$\partial_t u = \operatorname{div}(\mathbf{D} \cdot \nabla u) .$$

In image processing, the concentration u corresponds to image gray values. Depending on whether the diffusivity \mathbf{D} is adapted to local image structure, we distinguish between *linear* and *nonlinear* isotropic diffusion filtering.

1) Linear isotropic diffusion

Consider the case where the diffusivity is constant and normalized to one, $\mathbf{D} = 1$:

$$\partial_t u = \operatorname{div}(\nabla u) = \Delta u = \partial_{xx} u + \partial_{yy} u .$$

- a) As is shown in the lecture, the unique solution of this differential equation (under the boundary condition that u corresponds to the given image f for $t = 0$) is a convolution of f with a Gaussian. How is the standard deviation σ related to the diffusion time?

- b) In order to numerically implement the diffusion process, let u be discretized in space with step size h and in time with step size τ : $u_{ij}^{(n)} := u(hi, hj, n\tau)$. Approximate the derivatives in the diffusion equation by finite difference operators (Taylor's Formula), using 'forward differences' for the time derivatives $(\partial_t u)_{ij}^{(n)}$, and 'central differences' (involving the four nearest neighbors of $u_{ij}^{(n)}$) for the spatial derivatives $(\partial_{xx} u)_{ij}^{(n)}$ and $(\partial_{yy} u)_{ij}^{(n)}$.
- c) Derive the recursive definition of the diffusion process by computing $u_{ij}^{(n+1)}$ from $u^{(n)}$.
- d) Which time step sizes τ lead to a stable filtering process? What kind of filter is obtained in this case?

2) Nonlinear isotropic diffusion

Linear isotropic diffusion processes blur edges. In order to prevent this, the diffusivity can be adapted so that it decreases close to edges and is large only in homogeneous regions. For this purpose, let's assume that the diffusivity is a function of the magnitude of the gradient $|\nabla u|$. In the following, consider the one-dimensional case, where $\nabla u = \partial_x u$, which results in the diffusion equation

$$\partial_t u = \partial_x (\mathbf{D}(|\partial_x u|) \partial_x u).$$

- a) Sketch the shape of the diffusivity function $\mathbf{D}(|\partial_x u|)$ that models the desired behavior.
- b) Perona and Malik have suggested the diffusivity function

$$\mathbf{D}(|\nabla u|) = \frac{1}{1 + |\nabla u|^2 / \lambda^2},$$

where λ is a contrast parameter. Derive a formula for this case (in 1D) that describes the variation in time of an edge, which is given by $\partial_t \partial_x u$.

Hint: Substitute the mathematical flux function $\Phi(\partial_x u) = -j = \mathbf{D}(|\partial_x u|) \cdot \partial_x u$, and differentiate using Φ .

- c) Suppose that a step edge is oriented in such a way that $\partial_x u > 0$. Sketch u and its first three derivatives. How do points close to the edge vary in time in comparison to the first derivative of the flux $\Phi'(\partial_x u)$?
- d) Using the results of the previous questions, describe which edges are smoothed and which are enhanced by this nonlinear diffusion process.

Hint: Sketch the mathematical flux function Φ for the Perona-Malik diffusivity and observe the behavior of $\Phi'(\partial_x u)$.

3) Diffusion in Matlab

To investigate the effects of diffusion for image filtering, we use functions from the nonlinear diffusion toolbox, which is available on the course web page.

- a) The toolbox contains several demos for nonlinear diffusion that are based on a diffusivity function

$$\mathbf{D}(|\nabla u|) = \begin{cases} 1 - \exp\left(-\frac{c_m}{(|\nabla u|/\lambda)^m}\right), & \nabla u > 0 \\ 1, & \nabla u \leq 0 \end{cases}$$

where the constant c_m is chosen such that the flux is monotonously increasing for $\nabla u \leq \lambda$ ($c_m = 3.31488$ for $m = 8$). Basically, this diffusivity has the same properties as the Perona-Malik diffusivity discussed in the previous question.

Try out the demos `nldif1demo.m`, `nldifdemo1.m`, `nldifdemo2.m` and `nldifdemo3.m` to explore the capabilities of nonlinear diffusion.

- b)** The command 'ldif' applies linear diffusion to an image with the spatial step size being fixed to 1. Fixing the temporal step size to 0.25, linear diffusion is applied for `nsteps` time steps in the following way:

```
t_stepsize = 0.25; nsteps = 10;
ima_ldif = ldif(ima,t_stepsize,nsteps);
```

Using the Lenna image from the previous exercise sheet, apply linear diffusion for 4,20,40,80,200 steps, and display the results. Can you obtain the same results with a Gaussian low pass filter?

- c)** In order to apply nonlinear diffusion with the Perona-Malik diffusivity, you can use the command 'pmdif':

```
lambda = 0.05;
t_stepsize = 0.25; nsteps = 10;
ima_pmdif = pmdif(ima,lambda,0,t_stepsize,nsteps);
```

Additionally to the time step size and the number of steps, you need to specify the contrast parameter λ which specifies the minimum strength of an edge to be sharpened.

Using a fixed step size of 0.25, apply Perona-Malik nonlinear diffusion to the Lenna image for different contrast parameters $\lambda = 0.01, 0.02, 0.05, 0.1, 0.2$ for 20 steps, and display the results. How does an increasing λ -value influence the result?

- d)** Now keep $\lambda = 0.05$ fixed, and apply nonlinear diffusion for 5,10,20,50,100 steps. How does the image change?
- e)** Add Gaussian noise with $\sigma = 0.02$ to the image, and apply nonlinear diffusion with $\lambda = 0.05$ for 10 steps. How do you explain the result?
- f)** Apply nonlinear diffusion for step sizes 0.5 and 1. How do you explain the results?