Problem 1 (Linear Parabolic Stuff)

We begin our analysis by considering the discretised diffusion equation

$$u_t = \nabla \cdot (D(u)\nabla u) \tag{1}$$

by taking into account the four pixel boundary segments $l_{ij,1}, \ldots, l_{ij,4}$ as in (12.3):

$$\frac{\mathrm{d}}{\mathrm{d}t}\overline{u}_{ij}(t) = \frac{1}{|\sigma_i|}\sum_{k=1}^{4}\int\limits_{l_{ij,k}} (D(u)\nabla u)\cdot\vec{n}\ \mathrm{d}s. \tag{2}$$

As a note, in the following we denote $D \equiv D(u)$. Furthermore, we keep this part as general as possible, as we can use this part also for exercise 2. Now we are doing the time integration

$$\underbrace{\int\limits_{t}^{t+\delta t}\left[\frac{\mathrm{d}}{\mathrm{d}t}\overline{u}_{ij}(t)\right]\ \mathrm{d}t}_{\overline{u}_{ij}(t+\delta t)-\overline{u}_{ij}(t)} = \frac{1}{|\sigma_i|}\int\limits_{t}^{t+\delta t}\left\{\sum_{k=1}^{4}\int\limits_{l_{ij,k}}(D\nabla u)\vec{n}\ \mathrm{d}s\right\}\ \mathrm{d}t. \tag{3}$$

In order to give a shorter notation for the time discretisation, we set in the following $\bar{u}_{ij}(t+\delta t) =: u_{ij}^{n+1}$ and $\bar{u}_{ij}(t) =: u_{ij}^{n}$. Let us rewrite this now as an explicit scheme, i.e.

$$u_{ij}^{n+1} = u_{ij}^{n} + \frac{1}{|\sigma_i|}\int\limits_{t}^{t+\delta t}\left\{\sum_{k=1}^{4}\int\limits_{l_{ij,k}}(D\nabla u)\vec{n}\ \mathrm{d}s\right\}\ \mathrm{d}t \tag{4}$$

Now we have to do the spatial discretisation, as in (12.11) we can rewrite this as

$$\approx u_{ij}^{n} + \frac{\delta t}{|\sigma_i|}\left\{\sum_{k=1}^{4}\int\limits_{l_{ij,k}}(D\nabla u)\big|_t\vec{n}\ \mathrm{d}s\right\} \tag{5}$$

$$= u_{i}^{n} + \frac{\delta t}{\Delta x\Delta y}\left\{\sum_{k=1}^{4}\int\limits_{l_{ij,k}}(D\nabla u)\big|_t\vec{n}\ \mathrm{d}s\right\} \tag{6}$$

Now, we have to approximate the boundary integrals as in (12.13), however, we need to be careful here. In the lecture we have used $h = \Delta x = \Delta y$. In most applications, this suffices, however in order to be precise, we have to consider the following. Each direction needs proper approximation, so if we would want to approximate with respect to the direction $l_{ij,1}$, i.e. the pixels right neighbour, then we have to consider this setup:

$$\left.\underbrace{\boxed{u_i \;\Big|\; u_{i+1}}}_{\text{spatial derivative}}\right\}\Delta y$$

This means, that we consider for the spatial derivative the standard approach, however, as the pixel width in y-direction, we have to consider that the spatial differentiation occurs along the entire border between both pixels u_i and u_{i+1}. As the size of this border is Δy, we have to include this in the approximation, i.e.

$$\int\limits_{l_{ijk}} (D\nabla u)\cdot\vec{n}\;\mathrm{d}s \quad\approx\quad \begin{cases} \Delta y \cdot [(D\nabla u)\cdot\vec{n}]|_{m_{ij,k},t} & \text{if } k = 1,3 \\ \Delta x \cdot [(D\nabla u)\cdot\vec{n}]|_{m_{ij,k},t} & \text{if } k = 2,4 \end{cases}$$

From this we can plug this in into (6):

$$\begin{aligned} = u_{ij}^n + \frac{\delta t}{\Delta x\Delta y}\Big(& \quad \Delta y[D\nabla u\cdot\vec{n}]|_{m_{ij,1},t} + \Delta x[D\nabla u\cdot\vec{n}]|_{m_{ij,2},t} \\ + & \quad \Delta y[D\nabla u\cdot\vec{n}]|_{m_{ij,3},t} + \Delta x[D\nabla u\cdot\vec{n}]|_{m_{ij,4},t}\Big) \end{aligned} \tag{7}$$

At this point, we will get a little more specific for our problem with $D(u) = D = I$ (so you can use up until this part also for the first half of exercise 2). This means that a discretisation of the diffusion tensor, evaluated for each point $m_{i,j,k}, t$ is

$$D\Big|_{m_{i,j,1},t} \quad =: \quad D^n_{i+\frac{1}{2},j} := \frac{D^n_{ij} + D^n_{i+1,j}}{2} = \frac{1+1}{2} = 1 \tag{8}$$

so we can use this in equation (7) together with the scalar products with the respective normal vectors $\vec{n} = (\pm 1, 0)^\top$ and $\vec{n} = (0, \pm 1)^\top$

$$\begin{aligned} = u_{ij}^n + \frac{\delta t}{\Delta x\Delta y}\Big(& \quad \Delta y u_x|_{i+\frac{1}{2},j,t} + \Delta x u_y|_{i,j+\frac{1}{2},t} \\ - & \quad \Delta y u_x|_{i-\frac{1}{2},j,t} - \Delta x u_y|_{i,j-\frac{1}{2},t}\Big) \end{aligned} \tag{9}$$

Now we can plug in the standard discretisation (as e.g. (12.16)) for the first derivatives

$$
\begin{aligned}
= u_{ij}^n + \frac{\delta t}{\Delta x \Delta y}\big(& \Delta y \frac{u_{i+1,j}^n - u_{ij}^n}{\Delta x} + \Delta x \frac{u_{i,j+1}^n - u_{ij}^n}{\Delta y} \\
- & \Delta y \frac{u_{i,j}^n - u_{i-1j}^n}{\Delta x} - \Delta x \frac{u_{i,j}^n - u_{i,j-1}^n}{\Delta y}\big)
\end{aligned} \tag{10}
$$

Now we just have to regroup this equation, and we get the desired

$$
\begin{aligned}
u_{ij}^{n+1} = u_{ij}^n & + \frac{\delta t}{\Delta x^2}(u_{i+1,j}^n - 2u_{ij}^n + u_{i-1,j}^n) \\
& + \frac{\delta t}{\Delta y^2}(u_{i,j+1}^n - 2u_{ij}^n + u_{i,j-1}^n).
\end{aligned} \tag{11}
$$

Problem 2 (Creepy nonlinear Parabolic Phenomenae)

1. So, as mentioned earlier, we skip the first steps of this, as we have already done them in exercise No.1 and continue at that point, where we introduced the Diffusion tensor. So we continue at equation (7)

$$
\begin{aligned}
u_{ij}^{n+1} = u_{ij}^n + \frac{\delta t}{\Delta x \Delta y}\big(& \Delta y [D\nabla u \cdot \vec{n}]\big|_{m_{ij,1},t} + \Delta x [D\nabla u \cdot \vec{n}]\big|_{m_{ij,2},t} \\
+ & \Delta y [D\nabla u \cdot \vec{n}]\big|_{m_{ij,3},t} + \Delta x [D\nabla u \cdot \vec{n}]\big|_{m_{ij,4},t}\big)
\end{aligned}
$$

   As being given in the instructions, we set as the diffusion tensor (with the function $g(s^2) = \frac{1}{1+\frac{s^2}{\lambda^2}}$)

$$
\begin{aligned}
D = g(|\nabla u|^2) I & = \begin{pmatrix} g(|\nabla u|^2) & 0 \\ 0 & g(|\nabla u|^2) \end{pmatrix} \\
& = \begin{pmatrix} \dfrac{1}{1+\dfrac{u_x^2+u_y^2}{\lambda^2}} & 0 \\ 0 & \dfrac{1}{1+\dfrac{u_x^2+u_y^2}{\lambda^2}} \end{pmatrix}
\end{aligned}
$$

   Now, we have to employ a discretisation of the derivatives. For this, we will use the central differences, i.e.

$$
g(|\nabla u|^2)\big|_t \approx \frac{1}{1+\dfrac{\left(\dfrac{u_{i+1,j}^n - u_{i-1,j}^n}{2\Delta x}\right)^2 + \left(\dfrac{u_{i,j+1}^n - u_{i,j-1}^n}{2\Delta y}\right)^2}{\lambda^2}}
$$

In the following, we abbreviate this to $D^n_{ij} := g(|\nabla u|^2)|_{i,j,t}$. The discretisation of the tensor are then given as in (12.15)

$$
\begin{aligned}
D\big|_{m_{i,j,1},t} &=: D^n_{i+\frac{1}{2},j} &= \frac{D^n_{ij} + D^n_{i+1,j}}{2} \\
D\big|_{m_{i,j,2},t} &=: D^n_{i,j+\frac{1}{2}} &= \frac{D^n_{ij} + D^n_{i,j+1}}{2} \\
D\big|_{m_{i,j,3},t} &=: D^n_{i-\frac{1}{2},j} &= \frac{D^n_{ij} + D^n_{i-1,j}}{2} \\
D\big|_{m_{i,j,4},t} &=: D^n_{i,j-\frac{1}{2}} &= \frac{D^n_{ij} + D^n_{i,j-1}}{2}
\end{aligned}
$$

We can plug this now in into our initial equation (together with the scalar products with the respective norms) and get

$$
\begin{aligned}
u^{n+1}_{ij} = u^n_{ij} + \frac{\delta t}{\Delta x \Delta y}\Big(& \Delta y \frac{D^n_{ij} + D^n_{i+1,j}}{2}\frac{u^n_{i+1,j} - u^n_{i,j}}{\Delta x} + \Delta x \frac{D^n_{ij} + D^n_{i,j+1}}{2}\frac{u^n_{i,j+1} - u^n_{i,j}}{\Delta y} \\
- & \Delta y \frac{D^n_{ij} + D^n_{i-1,j}}{2}\frac{u_{ij} - u_{i-1,j}}{\Delta x} - \Delta x \frac{D^n_{ij} + D^n_{i,j-1}}{2}\frac{u^n_{i,j} - u^n_{i,j-1}}{\Delta y}\Big)
\end{aligned}
$$

This leads to our sought iterative method

$$
\begin{aligned}
u^{n+1}_{ij} = u^n_{ij} \quad &+ \quad \frac{\delta t}{\Delta x^2}\left(\frac{D^n_{ij} + D^n_{i+1,j}}{2}(u^n_{i+1,j} - u^n_{i,j}) - \frac{D^n_{ij} + D^n_{i-1,j}}{2}(u_{ij} - u_{i-1,j})\right) \\
&+ \quad \frac{\delta t}{\Delta y^2}\left(\frac{D^n_{ij} + D^n_{i,j+1}}{2}(u^n_{i,j-1} - u^n_{i,j}) - \frac{D^n_{ij} + D^n_{i,j-1}}{2}(u^n_{i,j} - u^n_{i,j-1})\right),
\end{aligned}
$$

which concludes our calculations.

2. Let us consider a general nonlinear diffusion equation

$$
u_t \quad = \quad \nabla \cdot (D(u)\nabla u), \tag{12}
$$

where $D \equiv D(u)$ is a nonlinear diffusion tensor, with only diagonal entries. Let us employ the Theorem of Gauß at (12) over a pixel σ_i and divide by $|\sigma_i|$ (as in (11.3):

$$
\frac{1}{|\sigma_i|}\int_{\sigma_i} \frac{\mathrm{d}}{\mathrm{d}t} u(\vec{x}, t)\, \mathrm{d}\vec{x} \quad = \quad \frac{1}{|\sigma_i|}\int_{\partial\sigma_i} \nabla \cdot [D(u)\nabla u]\, \mathrm{d}\vec{s}
$$

Assuming smoothness of u, we can pull the temporal derivative in front of the left hand side integral, i.e.

$$
\underbrace{\frac{\mathrm{d}}{\mathrm{d}t}\left[\frac{1}{|\sigma_i|}\int_{\sigma_i} u(\vec{x}, t)\, \mathrm{d}\vec{x}\right]}_{\substack{\text{Average grey value} \\ \text{over pixel } \sigma_i}} \quad = \quad \frac{1}{|\sigma_i|}\int_{\partial\sigma_i} \nabla \cdot [D(u)\nabla u]\, \mathrm{d}\vec{s}
$$

$$\Leftrightarrow \frac{\mathrm{d}}{\mathrm{d}t}\overline{u}_i(t) = \frac{1}{|\sigma_i|}\int\limits_{\partial\sigma_i} [D(u)\cdot\nabla u]\cdot\vec{n}\;\mathrm{d}s. \qquad (13)$$

For $\nabla u \equiv \vec{0}$ along $\partial\sigma_i$ according to von Neumann boundary conditions we obtain

$$\frac{\mathrm{d}}{\mathrm{d}t}\overline{u}_i(t) = \frac{1}{|\sigma_i|}\int\limits_{\partial\sigma_i} [D(u)\cdot\vec{0}]\cdot\vec{n}\;\mathrm{d}s = 0, \qquad (14)$$

i.e., analogously to the procedure in §11, we see that the average grey value is conserved for von Neumann boundary conditions. In order to have a proper numerical implementation, this von Neumann boundary needs some further considerations. Consider a 1-D signal $u = (u_1, \ldots, u_N)^\top$. Now consider, we would have to introduce ghost boundary pixels u_0 and u_{N+1} in order to employ von Neumann conditions for the derivatives on the border pixels. For this, you may just mirror pixels u_1 and u_N, so that you have $u_0 = u_1$ and $u_{N+1} = u_N$. This suffices for the ensurance of the average grey value.
