Numerical Algorithms for Visual Computing III 2011
Example Solutions for Assignment 2

Problem 1 (Tortellini and Ambrosia)

1. We begin our analysis by considering the Lagrangian of the Functional

$$E_{TA}(u,v) \quad = \quad \int_a^b \beta\,(u-f)^2 + v^2\,(u')^2 + \alpha\left(\gamma\,(v')^2 + \frac{(1-v)^2}{4\gamma}\right)\,dx$$

   which is given as

$$F(x,u,v,u_x,v_x) \quad = \quad \beta\,(u-f)^2 + v^2\,(u')^2 + \alpha\left(\gamma\,(v')^2 + \frac{(1-v)^2}{4\gamma}\right)$$

   The Euler-Lagrange equation for this energy functional is given as

$$\begin{aligned} 0 &\stackrel{!}{=} F_u - \frac{\mathrm{d}}{\mathrm{d}x}F_{u'} \\ 0 &\stackrel{!}{=} F_v - \frac{\mathrm{d}}{\mathrm{d}x}F_{v'} \end{aligned}$$

   with the ingredients

$$\begin{aligned} F_u &= 2\beta(u-f) \\ F_v &= 2v(u')^2 - \frac{2\alpha}{4\gamma}(1-v) \\ F_{u'} &= 2u'v^2 \\ F_{v'} &= 2\alpha\gamma v' \end{aligned}$$

   resulting in

$$\begin{aligned} 0 &\stackrel{!}{=} \beta(u-f) - \frac{\mathrm{d}}{\mathrm{d}x}u'v^2 \\ 0 &\stackrel{!}{=} v(u')^2 - \frac{\alpha}{4\gamma}(1-v) - \frac{\mathrm{d}}{\mathrm{d}x}\alpha\gamma v' \end{aligned}$$

2. With $w := (u,v)^\top$ we can compute

$$\nabla_w E_{TA}(w) \quad = \quad \begin{pmatrix} F_u - \frac{\mathrm{d}}{\mathrm{d}x}F_{u'} \\ F_v - \frac{\mathrm{d}}{\mathrm{d}x}F_{v'} \end{pmatrix} = \begin{pmatrix} \beta(u-f) - \frac{\mathrm{d}}{\mathrm{d}x}u'v^2 \\ v(u')^2 - \frac{\alpha}{4\gamma}(1-v) - \frac{\mathrm{d}}{\mathrm{d}x}\alpha\gamma v' \end{pmatrix}$$

3. We need to find out whether the function is convex or not. Consider the second part of the above computed Euler-Lagrange equation, neglecting $\frac{\mathrm{d}}{\mathrm{d}x} 2\alpha\gamma v' = 2\alpha\gamma v''$. With this, we can compute a function $v(|u_x|^2)$ that is depending solely on the first derivative of u, hence, simplifying the Euler-Lagrange equation. This idea is taken from the analysis by Chan and Vese of the Ambrosio-Tortorelli algorithm. Therefore, the modified second equation becomes:

$$\begin{aligned}
& v(u')^2 - \tfrac{\alpha}{4\gamma}(1-v) &&= 0 \\
\Leftrightarrow\quad & v(u')^2 - \tfrac{\alpha}{4\gamma} + \tfrac{\alpha v}{4\gamma} &&= 0 \\
\Leftrightarrow\quad & v((u')^2 + \tfrac{\alpha}{4\gamma}) &&= \frac{\alpha}{4\gamma} \\
\Leftrightarrow\quad & v &&= \frac{\alpha}{\alpha + 4\gamma(u')^2}
\end{aligned}$$

We can then plug this result into the initial energy functional (neglecting the v' term):

$$\begin{aligned}
E_{TA}(u,v) &= \int_a^b \beta\,(u-f)^2 + v^2\,(u')^2 + \alpha\frac{(1-v)^2}{4\gamma}\,dx \\
&= \int_a^b \beta\,(u-f)^2 + \left(\frac{\alpha}{\alpha+4\gamma(u')^2}\right)^2 (u')^2 + \frac{\alpha}{4\gamma}\left(\frac{4\gamma(u')^2}{\alpha+4\gamma(u')^2}\right)^2 dx \\
&= \int_a^b \beta\,(u-f)^2 + \frac{\alpha(u')^2}{\alpha+4\gamma(u')^2}\,dx
\end{aligned}$$

If we consider the arising regulariser we can see that this function is non-convex, i.e. this function does not have a unique minimiser.

4. At that point, we have several issues to discuss. The first question is the question how to choose appropriate boundary conditions and the second is how we discretise in the first place. In order to achieve a second order derivative as given in the problem description, we should discretise such that the second derivative arises. This is done for example by considering forward derivatives for

$$u'\big|_{x=x_j} \approx \frac{u_{j+1}-u_j}{\Delta x}$$

and backward derivatives for

$$\left.\frac{\mathrm{d}}{\mathrm{d}x}\right|_{x=x_j} \approx \frac{[\cdots]|_{x=x_j} - [\cdots]|_{x=x_{j-1}}}{\Delta x}$$

We now discretise for $j = 1, \ldots, N$ the first Euler-Lagrange equation (also u_0 and u_{N+1} are boundary values that we will discuss later):

$$\begin{aligned}
\left[\beta(u-f) - \frac{\mathrm{d}}{\mathrm{d}x}u'v^2\right]\Bigg|_{x=x_j} &= \beta(u_j - f_j) - \frac{[u'v^2]|_{x=x_j} - [u'v^2]|_{x=x_{j-1}}}{\Delta x} \\
&= \beta(u_j - f_j) - \frac{\left(\frac{u_{i+1}-u_i}{\Delta x}\right)v_j^2 - \left(\frac{u_i - u_{i-1}}{\Delta x}\right)v_{j-1}^2}{\Delta x} \\
&= \beta(u_j - f_j) - \left(\frac{u_{i+1}-u_i}{\Delta x^2}\right)v_j^2 + \left(\frac{u_i - u_{i-1}}{\Delta x^2}\right)v_{j-1}^2
\end{aligned}$$

If we consider now Neumann boundary conditions for this equation, we have to consider ghost pixels $u_{N+1} := u_N$ and $u_0 := u_1$, as well as $v_{N+1} := v_N$ and $v_0 := v_1$. This leads to

$$\begin{aligned}
\text{for } u_1: &\qquad \beta(u_1 - f_1) - \left(\frac{u_2 - u_1}{\Delta x^2}\right)v_1^2 \\
\text{for } u_N: &\qquad \beta(u_N - f_N) + \left(\frac{u_N - u_{N-1}}{\Delta x^2}\right)v_{N-1}^2
\end{aligned}$$

Let us now come to the second equation, which we will discretise with the same setup:

$$\begin{aligned}
&\left[v(u')^2 - \frac{\alpha}{4\gamma}(1-v) - \frac{\mathrm{d}}{\mathrm{d}x}\alpha\gamma v'\right]\Bigg|_{x=x_j} \\
=\ & v_j\left(\frac{u_{j+1}-u_j}{\Delta x}\right)^2 - \frac{\alpha}{4\gamma}(1-v_j) - \alpha\gamma\frac{[v']|_{x=x_j} - [v']|_{x=x_{j-1}}}{\Delta x} \\
=\ & v_j\left(\frac{u_{j+1}-u_j}{\Delta x}\right)^2 - \frac{\alpha}{4\gamma}(1-v_j) - \alpha\gamma\frac{\frac{v_{j+1}-v_j}{\Delta x} - \frac{v_j - v_{j-1}}{\Delta x}}{\Delta x} \\
=\ & v_j\left(\frac{u_{j+1}-u_j}{\Delta x}\right)^2 - \frac{\alpha}{4\gamma}(1-v_j) - \alpha\gamma\frac{v_{j+1} - 2v_j + v_{j-1}}{\Delta x^2}
\end{aligned}$$

Again, with the Neumann boundary conditions, we also have

$$\begin{aligned}
\text{for } v_1: &\qquad v_j\left(\frac{u_2 - u_1}{\Delta x}\right)^2 - \frac{\alpha}{4\gamma}(1-v_1) - \alpha\gamma\frac{v_2 - v_1}{\Delta x^2} \\
\text{for } v_N: &\qquad -\frac{\alpha}{4\gamma}(1-v_N) - \alpha\gamma\frac{v_{N-1} - v_N}{\Delta x^2}
\end{aligned}$$

5. An iterative scheme is then given by the explicit iteration scheme

$$
\begin{aligned}
u_j^{k+1} &= u_j^k + \Delta t \left(\beta(u_j^k - f_j) - \left(\frac{u_{i+1}^k - u_i^k}{\Delta x^2} \right) (v_j^k)^2 + \left(\frac{u_i^k - u_{i-1}^k}{\Delta x^2} \right) (v_{j-1}^k)^2 \right) \\
v_j^{k+1} &= v_j^k + \Delta t \left(v_j \left(\frac{u_{j+1}^k - u_j^k}{\Delta x} \right)^2 - \frac{\alpha}{4\gamma}(1 - v_j^k) - \alpha\gamma \frac{v_{j+1}^k - 2v_j^k + v_{j-1}^k}{\Delta x^2} \right).
\end{aligned}
$$

Notice that the iterative scheme for the pixels at the boundaries needs to be taken care of in the implementation.

6. Let us discretise the energy functional:

$$
E_{\Delta TA}(u, v) = \Delta x \sum_{j=1}^{N} \left(\beta \left(u_j - f_j\right)^2 + v_j^2 \left(u'\right)^2 \Big|_{x=x_j} + \alpha \left(\gamma \left(v'\right)^2 \Big|_{x=x_j} + \frac{\left(1 - v_j\right)^2}{4\gamma} \right) \right)
$$

Now we need to discretise the derivatives. Our choice will be to use forward derivatives:

$$
\approx \Delta x \sum_{j=1}^{N} \left(\beta \left(u_j - f_j\right)^2 + v_j^2 \left(\frac{u_{j+1} - u_j}{\Delta x} \right)^2 + \alpha \left(\gamma \left(\frac{v_{j+1} - v_j}{\Delta x} \right)^2 + \frac{\left(1 - v_j\right)^2}{4\gamma} \right) \right)
$$

For computing a unique minimiser, we consider the derivatives $\frac{\partial E_{\Delta TA}}{\partial u_j}$ and $\frac{\partial E_{\Delta TA}}{\partial v_j}$ for all j. As before, we consider Neumann boundary conditions, which result in special conditions:

$$
\begin{aligned}
\frac{\partial E_{\Delta TA}}{\partial u_1} &= 2\Delta x \beta(u_1 - f_1) - 2\Delta x \frac{2v_1^2}{\Delta x^2}(u_2 - u_1) \\
\frac{\partial E_{\Delta TA}}{\partial u_j} &= 2\Delta x \beta(u_j - f_j) - 2\Delta x \frac{2v_j^2}{\Delta x^2}(u_{j+1} - u_j) + 2\Delta x \frac{2v_{j-1}^2}{\Delta x^2}(u_j - u_{j-1}) \\
&\qquad \text{for } j = 2, \ldots, N-1 \\
\frac{\partial E_{\Delta TA}}{\partial u_N} &= 2\Delta x \beta(u_N - f_N) + 2\Delta x \frac{2v_{N-1}^2}{\Delta x^2}(u_N - u_{N-1})
\end{aligned}
$$

Also we have to derive w.r.t. the v-components:

$$
\begin{aligned}
\frac{\partial E_{\Delta TA}}{\partial v_1} &= 2\Delta x v_1 \left(\frac{u_2 - u_1}{\Delta x}\right)^2 - \Delta x \frac{2\alpha\gamma}{\Delta x^2}(v_2 - v_1) - \Delta x \frac{2\alpha}{4\gamma}(1 - v_1) \\
\frac{\partial E_{\Delta TA}}{\partial v_j} &= 2\Delta x v_j \left(\frac{u_{j+1} - u_j}{\Delta x}\right)^2 - \Delta x \frac{2\alpha\gamma}{\Delta x^2}(v_{j+1} - v_j) \\
&\quad + \Delta x \frac{2\alpha\gamma}{\Delta x^2}(v_j - v_{j-1}) - \Delta x \frac{2\alpha}{4\gamma}(1 - v_j) \\
&= 2\Delta x v_j \left(\frac{u_{j+1} - u_j}{\Delta x}\right)^2 - \Delta x \frac{2\alpha}{4\gamma}(1 - v_j) - \Delta x \frac{2\alpha\gamma}{\Delta x^2}(v_{j+1} - v_j + v_{j-1}) \\
\frac{\partial E_{\Delta TA}}{\partial v_N} &= \Delta x \frac{2\alpha\gamma}{\Delta x^2}(v_N - v_{N-1}) - \Delta x \frac{2\alpha}{4\gamma}(1 - v_N)
\end{aligned}
$$

Notice that all computed derivates should be equal to zero for a minimiser, i.e. the term Δx cancels out everywhere.

7. An iterative scheme is then given by

$$
\begin{aligned}
u_j^{k+1} &= u_j^k + \Delta t \left(\beta(u_j^k - f_j) - \left(\frac{u_{i+1}^k - u_i^k}{\Delta x^2}\right)(v_j^k)^2 + \left(\frac{u_i^k - u_{i-1}^k}{\Delta x^2}\right)(v_{j-1}^k)^2\right) \\
v_j^{k+1} &= v_j^k + \Delta t \left(v_j \left(\frac{u_{j+1}^k - u_j^k}{\Delta x}\right)^2 - \frac{\alpha}{4\gamma}(1 - v_j^k) - \alpha\gamma \frac{v_{j+1}^k - 2v_j^k + v_{j-1}^k}{\Delta x^2}\right).
\end{aligned}
$$

Notice that this is the same iterative scheme as above, however this might change if other discretisations for the derivatives have been chosen.

Problem 2 (The Problem of Nail and Grandsonman™)

1. We want to minimise the following functional:

$$
\begin{aligned}
E_2(u, v) &= \int_\Omega ((f_x u + f_y v + f_t)^2 \\
&\quad + \alpha(\nabla u^\top D(\nabla f)\nabla u + \nabla v^\top D(\nabla f)\nabla v))\, \mathrm{d}x\, \mathrm{d}y
\end{aligned}
$$

This is an Optic Flow model with an anisotropic image-driven regulariser as proposed by Nagel in 1983. The Lagrangian of this functional is given by

$$
\begin{aligned}
G(x, y, u, v, \nabla u, \nabla v) &= ((f_x u + f_y v + f_t)^2 \\
&\quad + \alpha(\nabla u^\top D(\nabla f)\nabla u + \nabla v^\top D(\nabla f)\nabla v)
\end{aligned}
$$

A minimiser of the sought functional satisfies necessarily the Euler-Lagrange equations

$$\begin{aligned} G_u - \partial_x G_{u_x} - \partial_y G_{u_y} &= 0 \\ G_v - \partial_x G_{v_x} - \partial_y G_{v_y} &= 0 \end{aligned}$$

Let us now compute the missing components of the E-L equations:

$$\begin{aligned} G_u &= 2(f_x u + f_y v + f_t) f_x \\ G_v &= 2(f_x u + f_y v + f_t) f_y \\ G_{u_x} &= \alpha(e_1^\top D(\nabla f)\nabla u + \nabla u^\top D(\nabla f) e_1) = 2\alpha(e_1^\top D(\nabla f)\nabla u) \\ G_{u_y} &= \alpha(e_2^\top D(\nabla f)\nabla u + \nabla u^\top D(\nabla f) e_2) = 2\alpha(e_2^\top D(\nabla f)\nabla u) \\ G_{v_x} &= \alpha(e_1^\top D(\nabla f)\nabla v + \nabla v^\top D(\nabla f) e_1) = 2\alpha(e_1^\top D(\nabla f)\nabla v) \\ G_{v_y} &= \alpha(e_2^\top D(\nabla f)\nabla v + \nabla v^\top D(\nabla f) e_2) = 2\alpha(e_2^\top D(\nabla f)\nabla v) \end{aligned}$$

It should be noted that the terms $e_i^\top D(\nabla f)\nabla u_j$ and $\nabla u_j^\top D(\nabla f) e_i$ are equal, as the result of this multiplication gives a scalar value eventually. Combining all entries, gives us the sought Euler-Lagrange equations

$$\begin{aligned} 2(f_x u + f_y v + f_t) f_x - \partial_x 2\alpha(e_1^\top D(\nabla f)\nabla u) - \partial_y 2\alpha(e_2^\top D(\nabla f)\nabla u) &= 0 \\ 2(f_x u + f_y v + f_t) f_y - \partial_x 2\alpha(e_1^\top D(\nabla f)\nabla v) - \partial_y 2\alpha(e_2^\top D(\nabla f)\nabla v) &= 0 \end{aligned}$$

By employing the following rule for the divergence operator

$$\text{div } (A) = \sum_i \partial_{x_i}(e_i^\top A),$$

we can simplify the Euler-Lagrange equation to

$$\begin{aligned} \text{div } (D(\nabla f)\nabla u) - \frac{1}{\alpha}(f_x u + f_y v + f_t) f_x &= 0 \\ \text{div } (D(\nabla f)\nabla v) - \frac{1}{\alpha}(f_x u + f_y v + f_t) f_y &= 0 \end{aligned}$$

2. You can see, if you set $D(\nabla f) = I$, that the regulariser becomes that of the Horn and Schunck method, i.e. this is a special case for this kind of regulariser. This can be seen from the regulariser itself by applying $\nabla u^\top D(\nabla f) u = u^\top I u = u^\top u = |\nabla u|^2 = u_x^2 + u_y^2$. Also, the Euler-Lagrange equation boils down to the same equation, as $\text{div } (D(\nabla f)\nabla u) = \text{div } (I\nabla u) = \text{div } (\nabla u) = \Delta u$