Assignment 2

Moser Stefan
09-277-013

Superresolution

Primal Dual formulation

The energy term we need to minimize for this problem is

$$\min_{u \in U} \|\nabla u\| + \frac{\lambda}{2} \|Du - g\|_2^2 \tag{1}$$

with D being the downsampling operator that produces from the high resolution image u (of size $M \times N$) a low level resolution image (of size $\frac{M}{\alpha} \times \frac{N}{\alpha}$) by multiplying. For all images, we assume that they are given as vectors[1].

So far, everything is the same as in the previous exercise. But instead of solving Equation (1) directly, we will derive the primal-dual formulation

$$\min_{u \in U} \max_{y \in Y} < Ku, y > -F^*(y) + G(u) \tag{2}$$

and solve it instead. All needed conditions are met:

- $F = \|\cdot\|_2$ is a convex function
- $G = \frac{\lambda}{2} \|D \cdot -g\|_2^2$ is a convex function
- $K = \nabla \cdot$ is a linear operator

[1] following the flattening conventions of matlabs (:) operator

Legendre Fenchel Transform of TV term

Since F only appears as it's convex conjugate F^* in Equation (2), we need to compute it first.

$$
\begin{aligned}
F^*(y) &= (\|\cdot\|_2)^*(y) && (3)\\
&= \sup_x x^T y - \|x\|_2 && (4)\\
&= \sup_x x^T y - \max_{\|z\|_2 \leq 1} x^T z && (5)\\
&= \sup_x \min_{\|z\|_2 \leq 1} x^T (y - z) && (6)\\
&= \begin{cases} 0 & \text{if } \|y\|_2 \leq 1 \\ \infty & \text{otherwise} \end{cases} && (7)\\
&= \delta(y) && (8)
\end{aligned}
$$

In step (5) we used the Cauchy-Schwarz inequality

$$
\|x\|_2 = \max_{\|z\|_2 \leq 1} x^T z \tag{9}
$$

Further, the result in Equation (7) can be derived by seeing that the inner minimum term always will be 0 in the first case, since then z will adapt to be equal to y, resulting in the multiplication with a zero vector. If z can not be set equal to y, i. e. $\|y\|_2 > 1$, the superior is unbound.

Consequently, the convex conjugate of the two norm is the *unit ball indicator function*[2].

We arrive at the Primal-Dual formulation for super resolution by simply putting our results so far into Equation (2).

$$
\min_{u \in U} \max_{y \in Y} < \nabla u, y > -\delta(y) + \frac{\lambda}{2} \|Du - g\|_2^2 \tag{10}
$$

[2] 1 if inside, ∞ if outside

Primal-Dual steps

Equation (2) can be solved using Primal-Dual steps. Specifically, the following algorithm:

$$
\begin{aligned}
y^{n+1} &= \mathrm{prox}_{\sigma F^*}(y^n + \sigma K \bar{x}^n) && (11)\\
x^{n+1} &= \mathrm{prox}_{\tau G}(x^n + \tau K^* y^{n+1}) && (12)\\
\bar{x}^{n+1} &= x^{n+1} + \theta(x^{n+1} - x^n) && (13)
\end{aligned}
$$

with $\theta \in (0, 1]$ and $\tau\sigma \left\|K\right\|^2 < 1$.

The proximity operator used there is defined as

$$
\mathrm{prox}_{\lambda F}(z) = \arg\min_x \frac{1}{2} \left\|x - z\right\|_2^2 + \lambda F(x) \tag{14}
$$

The terms from Equation (12) and (11) need some further derivation before they can be used, which is done in the following.

Derivation of y^{n+1}

For the first term defined in Equation (11) we get the following derivation:

$$
\begin{aligned}
y^{n+1} &= prox_{\sigma F^*}(y^n + \sigma K \bar{x}^n) && (15)\\
&= y^n + \sigma K \bar{x}^n - \sigma \cdot \mathrm{prox}_{F/\sigma}((y^n + \sigma K \bar{x}^n)/\sigma) && (16)\\
&= y^n + \sigma \nabla \bar{x}^n - \sigma \cdot \mathrm{prox}_{\|\cdot\|_2/\sigma}((y^n + \sigma \nabla \bar{x}^n)/\sigma) && (17)
\end{aligned}
$$

using Moreau's Identity in step (16) which is defined as

$$
\mathrm{prox}_{\lambda F^*}(z) = z - \lambda \cdot \mathrm{prox}_{F/\lambda}(z/\lambda) \tag{18}
$$

Now, we need to compute the proximity operator of the two norm as it occurs in Equation (17).

$$
\begin{aligned}
\mathrm{prox}_{\|\cdot\|_2/\lambda}(z/\lambda) &= \arg\min_x \frac{1}{2} \left\|x - \frac{z}{\lambda}\right\|_2^2 + \frac{1}{\lambda} \left\|x\right\|_2 && (19)\\
&= \arg\min_x P(x) && (20)
\end{aligned}
$$

This can be solved by deriving and setting to zero:

$$\frac{\partial}{\partial x}P(x) = \frac{\partial}{\partial x}\left(\frac{1}{2}\left\|x - \frac{z}{\lambda}\right\|_2^2 + \frac{1}{\lambda}\|x\|_2\right) \tag{21}$$

$$= \frac{\partial}{\partial x}\left(\frac{1}{2}(x - \frac{z}{\lambda})(x - \frac{z}{\lambda}) + \frac{1}{\lambda}(x^2)^{\frac{1}{2}}\right) \tag{22}$$

$$= x - \frac{z}{\lambda} + \frac{\partial}{\partial x}\left(\frac{1}{\lambda}(x^2)^{\frac{1}{2}}\right) \tag{23}$$

$$= x - \frac{z}{\lambda} + \frac{1}{\lambda}2x\frac{1}{2}(x^2)^{-\frac{1}{2}} \tag{24}$$

$$= x - \frac{z}{\lambda} + \frac{1}{\lambda}\frac{x}{\|x\|_2} \stackrel{!}{=} 0 \tag{25}$$

Since we can not easily solve Equation (25) for x, we need to express $P(x)$ as defined in (20) in a more convenient form, e. g. express x as composition of its direction and magnitude. Eventually we end up with

$$\operatorname{prox}_{\|\cdot\|_2/\lambda}(z/\lambda) = z/\lambda \cdot \max\left(0, 1 - \frac{1}{\|z\|_2}\right) \tag{26}$$

Plugging this into Equation (17), we get

$$y^{n+1} = y^n + \sigma\nabla\bar{x}^n - \sigma \cdot ((y^n + \sigma\nabla\bar{x}^n)/\sigma \cdot \max\left(0, 1 - \frac{1}{\|y^n + \sigma\nabla\bar{x}^n)\|_2}\right) \tag{27}$$

$$= y^n + \sigma\nabla\bar{x}^n - (y^n + \sigma\nabla\bar{x}^n) \cdot \max\left(0, 1 - \frac{1}{\|y^n + \sigma\nabla\bar{x}^n)\|_2}\right) \tag{28}$$

$$= \begin{cases} y^n + \sigma\nabla\bar{x}^n & \text{if } \|y^n + \sigma\nabla\bar{x}^n)\|_2 \leq 1 \\ \frac{y^n + \sigma\nabla\bar{x}^n}{\|y^n + \sigma\nabla\bar{x}^n\|_2} & \text{otherwise} \end{cases} \tag{29}$$

$$= \frac{y^n + \sigma\nabla\bar{x}^n}{\max\left(1, \|y^n + \sigma\nabla\bar{x}^n\|_2\right)} \tag{30}$$

Derivation of x^{n+1}

The derivation of the second term (Equation (12)) is similar to the one in the last section.

$$x^{n+1} = \operatorname{prox}_{\tau G}(x^n + \tau K^* y^{n+1}) \tag{31}$$

$$= \operatorname{prox}_{\tau\frac{\lambda}{2}\|D\cdot - g\|_2^2}(x^n + \tau\nabla^* y^{n+1}) \tag{32}$$

$$= \arg\min_m \frac{1}{2}\left\|m - x^n + \tau\nabla^* y^{n+1}\right\|_2^2 + \tau\frac{\lambda}{2}\|Dm - g\|_2^2 \tag{33}$$

$$= \arg\min_m L(m) \tag{34}$$

The extrema of $L(m)$ can be found by setting the derivative with respect to m to zero. Since it is a convex function, we can be sure that the found value is the minimum.

$$
\begin{aligned}
\frac{\partial}{\partial m} L(m) &= \frac{\partial}{\partial m}\left(\frac{1}{2}\left\|m - x^n + \tau \nabla^* y^{n+1}\right\|_2^2 + \tau \frac{\lambda}{2}\left\|Dm - g\right\|_2^2\right) && (35)\\
&= m - x^n + \tau \nabla^* y^{n+1} + \frac{\partial}{\partial m} \tau \frac{\lambda}{2}\left\|Dm - g\right\|_2^2 && (36)\\
&= m - x^n + \tau \nabla^* y^{n+1} + \tau \lambda D(m - g) && (37)\\
&= m - x^n + \tau \nabla^* y^{n+1} + \tau \lambda Dm - \tau \lambda Dg && (38)\\
&= (I + \tau \lambda D)m - x^n + \tau \nabla^* y^{n+1} - \tau \lambda Dg \overset{!}{=} 0 && (39)\\
\iff m &= (I + \tau \lambda D)^{-1}(x^n - \tau \nabla^* y^{n+1} + \tau \lambda Dg) && (40)\\
&= (I + \tau \lambda D)^{-1}(x^n + \tau \nabla \cdot y^{n+1} + \tau \lambda Dg) && (41)\\
&= x^{n+1} && (42)
\end{aligned}
$$

Overview of Primal-Dual steps

After derivation, we have the following steps:

$$
\begin{aligned}
y^{n+1} &= \frac{y^n + \sigma \nabla \bar{x}^n}{\max\left(1, \|y^n + \sigma \nabla \bar{x}^n\|_2\right)} && (43)\\
x^{n+1} &= (I + \tau \lambda D)^{-1}(x^n + \tau \nabla \cdot y^{n+1} + \tau \lambda Dg) && (44)\\
\bar{x}^{n+1} &= x^{n+1} + \theta(x^{n+1} - x^n) && (45)
\end{aligned}
$$

and are ready for doing our implementation.

Implementation

I chose the following parameters for my implementation

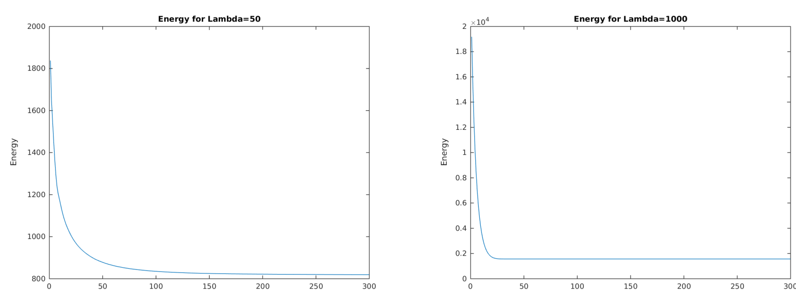
$$K_a = 8 \tag{46}$$
$$\tau = 0.001 \tag{47}$$
$$\sigma = \frac{1}{K_a \cdot \tau} \tag{48}$$
$$\theta = 1 \tag{49}$$

and as initialization

$$x^0 = 0 \tag{50}$$
$$y^0 = 0 \tag{51}$$
$$\bar{x}^0 = D^T g \tag{52}$$

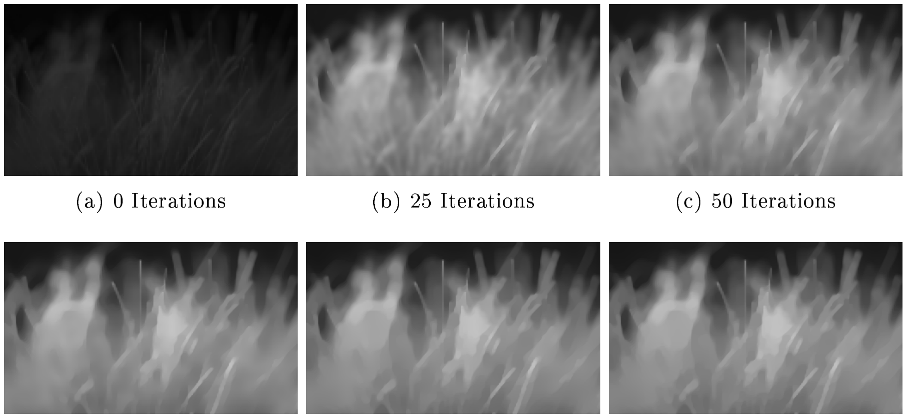
Iteration count

The number of iterations needed is heavily dependent on lambda. Smaller lambdas need much longer to converge than larger ones. The behavior of the energy function for both cases is visible in Figure 1. Some intermediary images for a small lambda can be seen in Figure 2, for a large lambda in Figure 3.



(a) $\lambda = 50$ (b) $\lambda = 1000$

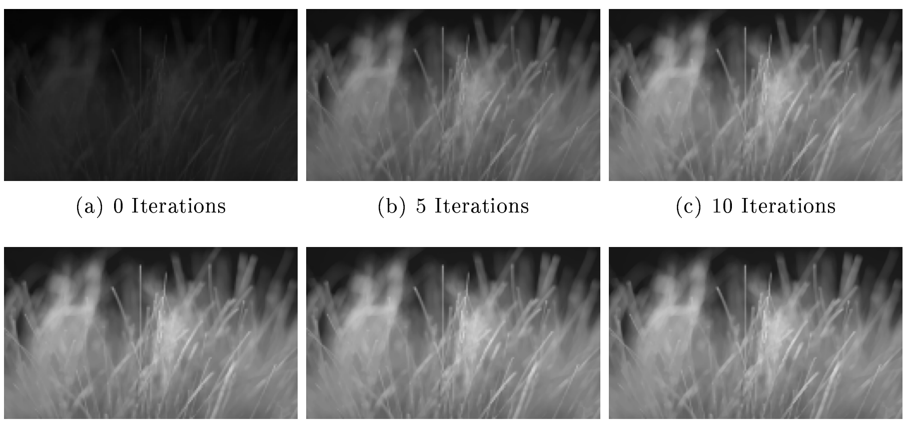
Fig. 1: Behavior of the energy function for different lambdas. For lambdas above a thousand, 50 iterations are enough. Smaller lambdas still improve slightly until about 300 iterations.



(a) 0 Iterations (b) 25 Iterations (c) 50 Iterations

(d) 100 Iterations (e) 200 Iterations (f) 300 Iterations

Fig. 2: The superresolution image after various iteration counts. 0 iterations shows the initial guess. After iteration 200 there are only minor changes. The lambda was set to 50 for these images.



(a) 0 Iterations (b) 5 Iterations (c) 10 Iterations

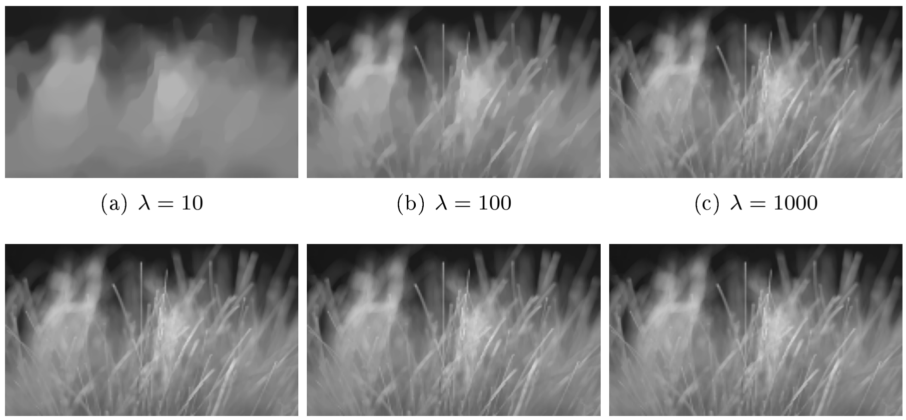
(d) 15 Iterations (e) 20 Iterations (f) 300 Iterations

Fig. 3: The superresolution image after various iteration counts. 0 iterations shows the initial guess. After iteration 30 there are only minor changes. The lambda was set to 1000 for these images.

Effect of Lambda

As in the previous assignment, the lambda defines the smoothness of the converged solution. Surprisingly, I could increase lambda to very large values without the quality deteriorating. With smaller lambdas ($\lambda < 100$), the algorithm did not converge anymore with default parameters, so I adapted my implementation to choose τ from Equation 47 according to lambda:

$$\tau_\lambda = \begin{cases} 0.1 & \text{if } \lambda < 100 \\ 0.001 & \text{otherwise} \end{cases} \tag{53}$$



(a) $\lambda = 10$ (b) $\lambda = 100$ (c) $\lambda = 1000$

(d) $\lambda = 2000$ (e) $\lambda = 5000$ (f) $\lambda = 10000$

Fig. 4: Various resulting images with different lambdas after 1000 iterations. The high amount of iterations was chosen to make sure the solution is converged.

Optimal Lambda

As already in the previous assignment, I could not find an upper bound to lambda where it starts to worsen again. But as visible in Figure 5, the SSD only decreases marginally after $\lambda = 1000$.

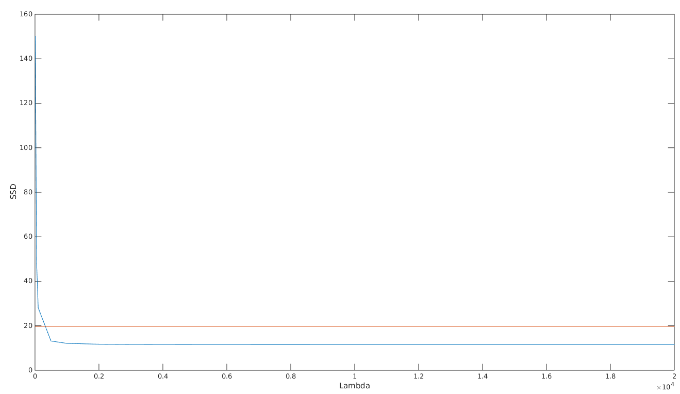
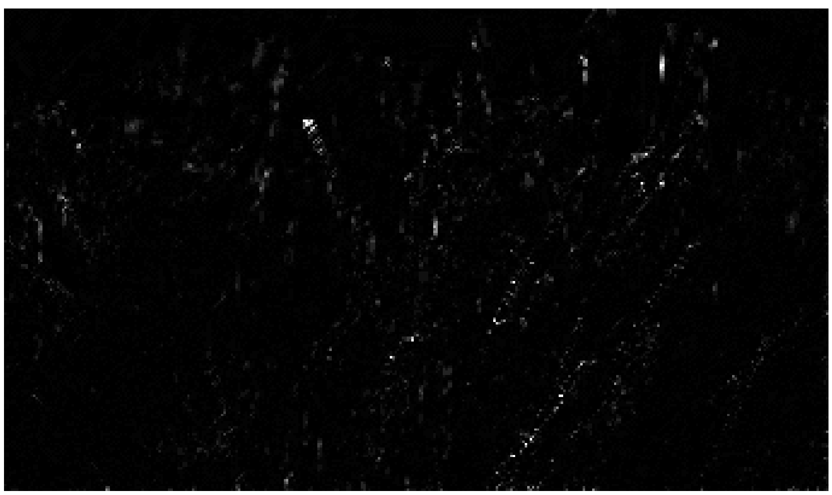


Fig. 5: SSD for various lambdas after 1000 iterations. For comparison, the SSD of the image obtained through nearest-neighbor scaling was added (green line).

Conclusions

While computational cost of one single iteration is larger is larger in the primal dual method, the number of iterations used until convergence is much lower. The convergence much trickier to achieve for the primal dual problem, since there are more parameters to tune there. For super resolution it took quite some time to find acceptable values. The converged results were almost the same with an SSD of 0.7646 at $\lambda = 1000$. The deviation does not seem to be caused by a boundary effect, as can be seen by the visualisation of the error in Figure 6.

(a) Ground truth



(b) SSD of results obtained trough gradient descent and primal dual method

Fig. 6: SSD scaled by 1000 between the image obtained through gradient descent and one obtained through the primal dual method. The most deviation is in places with edges.