Paul Hilt and Sebastian Müller – Cauchy Noise Removal by Nonconvex ADMM

Zusatzprojekt „Einführung in Mathematische Bildverarbeitung” im SS 2019

Sebastian Müller
sebastian_mueller@stud.uni-heidelberg.de
Paul Hilt
hiltpa@gmail.com

September 24, 2020

Abstract

In order to restore images from blurs and noise while also preserving their edges, one often applies total variation (TV) minimization. Cauchy noise is a kind of impulsive and non-Gaussian noise whose removal can be achieved by solving a nonconvex TV minimization problem, which is difficult due to nonconvexity and nonsmoothness. In paper [1] a specific alternating direction method of multiplier (ADMM) is developed to solve this problem and theoretically the convergence of the method to a stationary point is established. Experimental results demonstrate that the proposed method achieves higher PSNRs for 0.5 dB average.

1 Motivation and Outline

Images contain natural non-Gaussian noises, such as impulse noise, Poisson noise, multiplicative noise, and Cauchy noise and may have been blurred by the point spread function (PSF) during their acquisition. We focus on recovering images corrupted by Cauchy noise but **not** by blurring. We assume that the original image u is defined on a connected bounded domain $\Omega \subset \mathbb{R}^2$ with a compacted Lipschitz boundary. The observed image with Cauchy noise is given as

$$f = Ku + \eta \tag{1}$$

with u as the real, unperturbated image defined on a connected, bounded domain $\Omega \subset \mathbb{R}^2$ with a compacted Lipschitz boundary, $f \in L^2(\Omega)$ as the observation and $\eta \in L^2(\Omega)$ denoting the additive Cauchy noise. $K \in \mathbb{L}(L^1(\Omega), L^2(\Omega))$ representing a known linear blurring operator is set to the indicator function in the following because we don't consider blur. The objective is to reconstruct u from the observed image f which is corrupted by Cauchy noise.

2 Approach

2.1 Variational formulations

2.1.1 Convex formulation

A convex TV-based variational method for recovering is given by

$$\min_{u \in BV(\Omega)} \int_\Omega |\mathrm{D}u| + \frac{\lambda}{2} \left(\int_\Omega \log\left((Ku - f)^2 + \gamma^2\right) \mathrm{d}x + \alpha \left\| u - \tilde{u} \right\|_2^2 \right) \tag{2}$$

with scale parameter of Cauchy distribution $\gamma > 0$ and the space of bounded TVs $BV(\Omega) = \left\{u \in L^1(\Omega) | TV(u) < \infty\right\}$ while the TV is defined as

$$\mathrm{TV}(u) = \int_\Omega |\mathrm{D}u| = \sup_{\mathbf{v}} \left\{ \int_\Omega u \,\mathrm{div}\, \mathbf{v} \,\mathrm{d}x \; : \mathbf{v} \in \left(C_0^\infty(\Omega)\right)^2, \, \|\mathbf{v}\|_\infty \leq 1 \right\} \tag{3}$$

$BV(\Omega)$ equipped with the norm $\|u\|_{BV(\Omega)} = \|u\|_{L^1(\Omega)} + TV(u)$ is a Banach space. The regularization parameter $\gamma > 0$ controls trade-off between TV regularization and fitting to f and median filter $\tilde{u}$, $\alpha > 0$ is a penalty parameter. The objective is *strictly convex* and therefore uniquely solvable in case of $8\alpha\gamma^2 \geq 1$. Because of *strict convexity* the model 2 avoids issues of nonconvex optimization as the solution's dependence of numerical methods and initialization values. Due to the fact that pushing the solution u to the median filter $\tilde{u}$ does not provide satisfactory results, one unfortunately has to focus back to a *nonconvex* model.

2.1.2 Nonconvex formulation

In [1] an alternating direction method of multipliers (ADMM) is developed to solve the following *nonconvex* variational model for denoising (and deblurring in general). Starting from any initialization the method exhibits global convergence (to a starionary point) under certain conditions. We consider

$$\min_{u \in BV(\Omega)} \frac{\lambda}{2} \int_\Omega \log\left((Ku - f)^2 + \gamma^2\right) \mathrm{d}x + \int_\Omega |\mathrm{D}u| \tag{4}$$

with the regularization parameter $\gamma > 0$. While $\int_\Omega |\mathrm{D}u|$ is *convex*, due to log in the data-fitting term $\int_\Omega \left(\gamma^2 + (Ku - f)^2\right) \mathrm{d}x$ is *nonconvex*. Therefore as already mentioned above the solution of 4 strongly depends of numerical method and initialization. The existence and uniqueness of solution is proven in [1, 3.2].

2.2 ADMM Algorithm for Nonconvex and Nonsmooth Problem

In this section the ADMM algorithmus is applied to the nonconvex variational model 4 recovering images degraded by Cauchy noise (and blurring in general). Therefore we distretize the image as a set of pixels $\Omega \in \mathbb{R}^{n \times n}$ (mind that $n = \mathtt{d}$ in the code) and formulate the discrete minimization nonconvex model of 4 by

$$\min_{u \in \mathbb{R}^{n^2}} \|\boldsymbol{\nabla}_h u\|_1 + \frac{\lambda}{2} \langle \log\left(\gamma^2 \cdot \mathbb{1} + (K\mathbf{u} - \mathbf{f})^2\right), \mathbb{1} \rangle \tag{5}$$

with $\mathbb{1} = (1, 1, \ldots, 1)^T \in \mathbb{R}^{n^2}$ being a vector of ones, $\mathbf{f}, \mathbf{u} \in \mathbb{R}^{n^2}$ being now vectors obtained by stacking the coloumns of the corresponding $n \times n$ images and obviously being $K = \mathbb{I} \in \mathbb{R}^{n^2 \times n^2}$ the identity matrix. log is evaluated component wise here!

While the inner product is given by $\langle a, b \rangle = \sum_{i=1}^{n^2} a_i \cdot b_i$, the discretized TV regularization $\|\boldsymbol{\nabla}_h u\|_1$ defined as

$$\mathrm{TV}(u) = \|\boldsymbol{\nabla}_h u\|_1 = \sum_{i=1}^{n^2} \sqrt{(\boldsymbol{\nabla}_x u)_i^2 + (\boldsymbol{\nabla}_y u)_i^2} \tag{6}$$

with $\boldsymbol{\nabla}_x \in \mathbb{R}^{n^2 \times n^2}$ and $\boldsymbol{\nabla}_y \in \mathbb{R}^{n^2 \times n^2}$ being the discrete first order forward differences. The discrete gradient is defined as $\boldsymbol{\nabla}_h u = \left[(\boldsymbol{\nabla}_x u)^T, (\boldsymbol{\nabla}_y u)^T\right]^T \in \mathbb{R}^{2n^2}$.

The ADMM algorithm is then derived by introducing an auxiliary variable $\mathbf{v} \in \mathbb{R}^{n^2}$ and establishing a constrained minimization problem

$$\min_{u \in \mathbb{R}^{n^2}} \|\boldsymbol{\nabla}_h u\|_1 + \frac{\lambda}{2} \langle \log\left(\gamma^2 \cdot \mathbb{1} + (K\mathbf{u} - \mathbf{f})^2\right), \mathbb{1} \rangle \tag{7}$$

$$\text{s.t.} \;\; Ku = v \tag{8}$$

Then letting $\mathbf{w} \in \mathbb{R}^{n^2}$ be the Langrangian multiplier for the constraint $Ku = v$ we obtain the corresponding augmented Lagrangian

$$\mathcal{L}_\beta(\mathbf{u},\mathbf{v},\mathbf{w}) = \|\boldsymbol{\nabla}_h u\|_1 + \frac{\lambda}{2}\langle \log\left(\gamma^2\cdot\mathbb{1} + (K\mathbf{u}-\mathbf{f})^2\right), \mathbb{1}\rangle + \langle \mathbf{w}, K\mathbf{u}-\mathbf{v}\rangle + \frac{\beta}{2}\|K\mathbf{u}-\mathbf{v}\|_2^2 \tag{9}$$

with penalty parameter $\beta > 0$. This leads us directly to the algorithm for recovering the image corrupted by Cauchy noise (and in general also blurred). See [1, 4.1] for details). One obtains in total as the ADMM algorithm

1. INITIALIZE $\mathbf{u}^0$, $\mathbf{v}^0$, $\mathbf{w}^0$; set λ, β
2. For $k = 1, 2, \ldots$ ITERATE $\mathbf{u}^{k+1}$, $\mathbf{v}^{k+1}$, $\mathbf{w}^{k+1}$ by

$$\mathbf{u}^{k+1} \in \arg\min_{\mathbf{u}} \|\boldsymbol{\nabla}_h u\|_1 + \frac{\beta}{2}\left\|K\mathbf{u} - \mathbf{v}^k + \frac{\mathbf{w}^k}{\beta}\right\|_2^2 \tag{10}$$

$$\mathbf{v}^{k+1} = \arg\min_{\mathbf{v}} \frac{\lambda}{2}\langle \log\left(\gamma^2\cdot\mathbb{1} + (\mathbf{v}-\mathbf{f})^2\right), \mathbb{1}\rangle + \frac{\beta}{2}\left\|K\mathbf{u}^{k+1} - \mathbf{v} + \frac{\mathbf{w}^k}{\beta}\right\|_2^2 \tag{11}$$

$$\mathbf{w}^{k+1} = \mathbf{w}^k + \beta\left(K\mathbf{u}^{k+1} - \mathbf{v}^{k+1}\right) \tag{12}$$

3. STOP if $\mathbf{u}^{k+1}$ satisfies stopping criteria

Convergence of the ADMM method is proved in [1, Theorem 4.2].

Step 11 is solved by a Newton algorithm using the Gâteaux derivatives of the vector valued objective $\mathbf{F}(\mathbf{v}) = \arg\min_{\mathbf{v}} \frac{\lambda}{2}\langle \log\left(\gamma^2\cdot\mathbb{1} + (\mathbf{v}-\mathbf{f})^2\right), \mathbb{1}\rangle + \frac{\beta}{2}\left\|K\mathbf{u}^{k+1} - \mathbf{v} + \frac{\mathbf{w}^k}{\beta}\right\|_2^2$

$$\mathrm{D}\mathbf{F}(\mathbf{v}^k) = \lambda\frac{\mathbf{v}^k - \mathbf{f}}{\gamma^2\cdot\mathbb{1} + (\mathbf{v}^k-\mathbf{f})^2} - \beta\left(\mathbf{u} - \mathbf{v}^k + \frac{\mathbf{w}^k}{\beta}\right) \tag{13}$$

$$\mathrm{D}^2\mathbf{F}(\mathbf{v}^k) = \lambda\frac{\left(\gamma^2\cdot\mathbb{1} + (\mathbf{v}^k-\mathbf{f})^2\right) - 2(\mathbf{v}^k-\mathbf{f})^2}{\left(\gamma^2\cdot\mathbb{1} + (\mathbf{v}^k-\mathbf{f})^2\right)^2} + \beta\cdot\mathbb{1} = \lambda\frac{\gamma^2\cdot\mathbb{1} - (\mathbf{v}^k-\mathbf{f})^2}{\left(\gamma^2\cdot\mathbb{1} + (\mathbf{v}^k-\mathbf{f})^2\right)^2} + \beta\cdot\mathbb{1} \tag{14}$$

Our first approach to solve step 10 was using [2] (see subsection 2.3.1), but due to lack of convergence we used [3] instead as described in the following section 2.3.

2.3 Fast Gradient Projection (FGP)

The following method solving step 10 of the ADMM is described in [2] and implemented in `main.m`. We use the Fast Gradient Projection method (FGP) to solve a dual problem implemented in `denoising_isotrop_tv`. Therefore we modified the given `denoising_anisotrop_tv`. In our case we restrict to the dimensions $m = n = \mathtt{d}$. With the isotropic TV given as

$$\mathrm{TV}_I(\mathbf{x}) = \sum_{i=1}^{m-1}\sum_{j=1}^{n-1}\sqrt{(x_{i,j}-x_{i+1,j})^2 + (x_{i,j}-x_{i,j+1})^2} + \sum_{i=1}^{m-1}|x_{i,n} - x_{i+1,n}| + \sum_{j=1}^{n-1}|x_{m,j} - x_{m,j+1}| \tag{15}$$

where one assumes the standard reflexive boundary conditions $x_{m+1,j} - x_{m,j} = 0\ \forall j$ and $x_{i,n+1} - x_{i,n} = 0\ \forall i$. We consider constrained minimization problems of the form

$$\min_{\mathbf{x}\in C}\|\mathbf{x}-\mathbf{b}\| + 2\lambda\,\mathrm{TV}(\mathbf{x}) \tag{16}$$

with a closed convex set $C \subset \mathbb{R}^{m\times n}$ with $C = B_{l,u} = \{\mathbf{x} : l \le x_{ij} \le u,\ \forall i, j\}$ where $l = 0,\ u = 1$.

To construct a dual of the constrained problem, we introduce the dual variables $(\mathbf{p}, \mathbf{q})$ as elements of $\mathcal{P}$ which is a set of matrix-pairs $(\mathbf{p}, \mathbf{q})$ with $\mathbf{p} \in \mathbb{R}^{(m-1)\times n}$ and $\mathbf{q} \in \mathbb{R}^{m\times(n-1)}$ satisfying

$$p_{i,j}^2 + q_{i,j}^2 \le 1 \quad i = 1, \ldots, m-1;\ j = 1, \ldots, n-1 \tag{17}$$

$$|p_{i,j}| \le 1 \quad i = 1, \ldots, m-1 \tag{18}$$

$$|q_{i,j}| \le 1 \quad j = 1, \ldots, n-1 \tag{19}$$

while the entries are given by

$$p_{i,j} = x_{i,j} - x_{i+1,j}, \quad i = 1,\dots,m-1;\ j = 1,\dots,n \tag{20}$$

$$q_{i,j} = x_{i,j} - x_{i,j+1}, \quad i = 1,\dots,m;\ j = 1,\dots,n-1 \tag{21}$$

Furthermore the linear operator $\mathcal{L} : \mathbb{R}^{(m-1)\times n} \times \mathbb{R}^{m\times(n-1)} \mapsto \mathbb{R}^{m\times n}$ is defined by

$$\mathcal{L} : (\mathbf{p},\mathbf{q})_{i,j} = p_{i,j} + q_{i,j} - p_{i-1,j} - q_{i,j-1} \ \forall i = 1,\dots,m;\ j = 1,\dots,n \tag{22}$$

Obviously $\mathcal{L}$ is equivalent to the *adjoint* discrete gradient operator `G'` $= (\boldsymbol{\nabla}_h)^* = -\operatorname{div}_h$, the discrete divergence operator implemented in `gradient_discrete_4`.

Consequently, $\mathcal{L}^T : \mathbb{R}^{m\times n} \mapsto \mathbb{R}^{(m-1)\times n} \times \mathbb{R}^{m\times(n-1)}$ represents the discrete gradient operator `G` $= \boldsymbol{\nabla}_h$ applying the first order forward differences in both dimensions to the stacked coloumns vector `x`. The mapping is given by

$$\mathcal{L}^T(\mathbf{x}) = (\mathbf{p},\mathbf{q}) \tag{23}$$

In the code $\mathbf{p}$ corresponds to `px`, $\mathbf{q}$ coresponds to `py` and $(\mathbf{p},\mathbf{q})$ corresponds to `p1`. Note that `px`, `py` and `p1` are each vectors of stacked coloumns of the image.

Next we consider the orthogonal projection operator P_C on the set $C = B_{l,u}$

$$P_C(\mathbf{x})_{ij} = P_{B_{l,u}}(\mathbf{x})_{ij} = \begin{cases} l & x_{ij} < l \\ x_{ij} & l \le x_{ij} \le u \\ u & x_{ij} > u \end{cases} \tag{24}$$

In the following, we will shortly recover the Gradient Projection Method (GP) from [3, III-B]. Therefore we consider the nonsmooth convex optimization model $\min_{\mathbf{x}} f(\mathbf{x}) + g(\mathbf{x})$ with g being a proper closed convex function and f a continously Lipschitz differentiable function. The standard smooth convex constrained optimization problem $\min_{\mathbf{x}\in C} f(x)$ is obtained by choosing $g(\mathbf{x}) \equiv \delta_C$ with $C = B_{l,u} =\subset \mathbb{R}^{m\times n}$ being some closed convex set as above. The proximal mapping of $g = \delta_C$ is then given by $prox_t(g) = prox_t(\delta_C) = (I + t\partial\delta_C)^{-1} = P_C = P_{B_{l,u}}$. The optimality condition for $\mathbf{x}^*$ solving the convex minimization problem is given by $0 \in t\boldsymbol{\nabla} f(\mathbf{x}) + t\partial g(\mathbf{x}^*)$ that is equivalent to $\mathbf{x}^* = (I + t\partial g)^{-1}(I - t\boldsymbol{\nabla} f)(\mathbf{x}^*)$. This leads directly to the GP: $\mathbf{x}_k = P_C(\mathbf{x}_{k-1} - t_k\boldsymbol{\nabla} f(\mathbf{x}_{k-1}))$.

Coming back to the solving strategy for 16, we further need to introduce another projection operator $P_{\mathcal{P}}$ on $\mathcal{P}$ given by

$$P_{\mathcal{P}}(\mathbf{p},\mathbf{q}) = (\mathbf{r},\mathbf{s}) \tag{25}$$

where $\mathbf{r} \in \mathbb{R}^{(m-1)\times n}$ and $\mathbf{s} \in \mathbb{R}^{m\times(n-1)}$ are given by

$$r_{ij} = \begin{cases} \frac{p_{ij}}{\max\{1,\sqrt{p_{ij}^2+q_{ij}^2}\}} & i = 1,\dots,m-1;\ j = 1,\dots,n-1 \\ \frac{p_{in}}{\max\{1,|p_{in}\}|} & i = 1,\dots,m-1 \end{cases} \tag{26}$$

$$s_{ij} = \begin{cases} \frac{q_{ij}}{\max\{1,\sqrt{p_{ij}^2+q_{ij}^2}\}} & i = 1,\dots,m-1;\ j = 1,\dots,n-1 \\ \frac{q_{mj}}{\max\{1,|q_{mj}\}|} & j = 1,\dots,n-1 \end{cases} \tag{27}$$

$(\mathbf{r},\mathbf{s})$ corresponds to `r` in the code.

While [3, Proposition 4.1] establishes the dual problem that is convex and contains a continously differentiable objective, [3, Lemma 4.1] derives the gradient of the objective. We gain the **Fast Gradient Projection** algorithm **FGP**(b,λ,N) given by

- INPUT: $\mathbf{b}$ (observed image), λ) (regularization parameter), N (iterations)
- OUTPUT: optimal solution $\mathbf{x}^*$ of 16
- STEP 0: $(\mathbf{r}_1,\mathbf{s}_1) = (\mathbf{p}_0,\mathbf{q}_0) = (\mathbf{0}_{(m-1)\times n}, \mathbf{0}_{m\times(n-1)})$, $t_1 = 1$

- STEP k ($k = 1, \dots, N$) compute

$$(\mathbf{p}_k, \mathbf{q}_k) = P_{\mathcal{P}}\left[(\mathbf{r}_k, \mathbf{s}_k) + \frac{1}{8\lambda}\mathcal{L}^T(P_C[\mathbf{b} - \lambda\mathcal{L}(\mathbf{r}_k, \mathbf{s}_k)])\right] \tag{28}$$

$$t_{k+1} = \frac{1 + \sqrt{1 + 4t_k^2}}{2} \tag{29}$$

$$(\mathbf{r}_{k+1}, \mathbf{s}_{k+1}) = (\mathbf{p}_k, \mathbf{q}_k) + \frac{t_k - 1}{t_{k+1}}(\mathbf{p}_k - \mathbf{p}_{k-1}, \mathbf{q}_k - \mathbf{q}_{k-1}) \tag{30}$$

- SET

$$\mathbf{x}^* = P_C\left[\mathbf{b} - \lambda\mathcal{L}(\mathbf{p}_N, \mathbf{q}_N)\right] \tag{31}$$

The FGP method works pretty fine for our applications. The rate of convergence of the FGP measures $O(1/k^2)$.

2.3.1 Failed approach (Chambolle)

In this section we will shortly review our failed approach to solve step 10 of the ADMM with the Chambolle algorithm [2] instead of FGP. This is implemented in `mainChambolle.m`, the algorithm does **not** show convergency.

Given the minimization problem where we regard $g = v^k + \frac{w^k}{\beta}$ from the ADMM

$$\min_{u \in X} \frac{\|u - g\|^2}{2\lambda} + J(u) \tag{32}$$

one gets that $w = (g - u)/\lambda$ is the minimizer of

$$\frac{\|w - (g/\lambda)\|^2}{2} + \frac{1}{\lambda}J^*(w) \tag{33}$$

where $J* = \chi_K$ of a convex set K. One deduces $w = \pi_K(g/\lambda)$, hence the minimizing u is given as $u = g - \pi_{\lambda K}(g)$. Computing $\pi_{\gamma K}(g)$ amounts to solving the problem

$$\min\left\{\|\lambda \operatorname{div} p - g\|^2 : \; p \in Y, \; |p_{ij}|^2 - 1 \leq 0 \; \forall i, j = 1, \dots, N\right\} \tag{34}$$

Using Karush-Kuhn-Tucker conditions and a Lagrange multiplier this leads to the fixed point iteration

$$p_{i,j}^{n+1} = \frac{p_{i,j}^n + \tau(\boldsymbol{\nabla}(\operatorname{div} p^n - g/\lambda))_{i,j}}{1 + \tau|(\operatorname{div} p^n - g/\lambda))_{i,j}|} \tag{35}$$

[2, Theorem 3.1] shows that $\lambda \operatorname{div} p^n$ converges to $\pi_{\lambda K}(g)$ for $n \to \infty$.

Since σ is less difficult to determine than λ, we proposed another algorithm that tackles directly the solution of the objective

$$\min\left\{J(u) : \|u - g\|^2 = N^2\sigma^2\right\} \tag{36}$$

The task is to find $\lambda > 0$ s.t. $\|\pi_{\lambda K} g\|^2 = N^2\sigma^2$. Set $f(s) = \|\pi_{sK} g\|$.

Then the following algorithm was supposed to solve 36. The goal is to find $\overline{\lambda} > 0$ s.t. $f(\overline{\lambda}) = \|\pi_{\overline{\lambda}K} g\| = N\sigma$.

- assume $N\sigma \in [0, \|g - < g >\|]$
- set $\lambda_0 > 0$
- set $v_0 = \pi_{\lambda_0 K}(g) (= \lambda_0 \operatorname{div} p)$
- set $f_0 = f(\lambda_0) = \|v_0\|$
- step $\lambda_{n+1} = \frac{N\sigma}{f_n}\lambda_n$
- step $v_{n+1} = \pi_{\lambda_{n+1}K}(g) = \lambda_{n+1} \operatorname{div} p$

- step $f_{n+1} = \|v_{n+1}\|$

[2, Theorem 4.2] proofs that for $n \to \infty$, $f_n \to N\sigma$ the expression $g - v_n$ converges to the unique solution of 36.

Our assumption was that the update of $\lambda = 1/\beta$ (β from ADMM) in each step of the `tv_MinimizationChambolle` might cause the problem of non-existing convergence in the implementation, but skipping the substep did not change anything.

3 Experiments and results

In accordance to [1] we used the peak signal to noise ratio (PSNR) and the structured similarity index (SSIM) as measures for the performance of the algorithm. The PSNR shows better performance as it becomes larger whereas the SSIM shows better performance as it approaches 1. The following two images depict the history of SSIM and PSNR for the third color channel over all iterations, together with the resulting image for some parameter settings. The parameters in question are γ, β and λ as defined in [1]. γ corresponds to the scale parameter of the Cauchy distribution used to degrade the image. Throughout all tests we set $\gamma = 0.1$ which results in noisy images as depicted in figure 2 and figure 3.

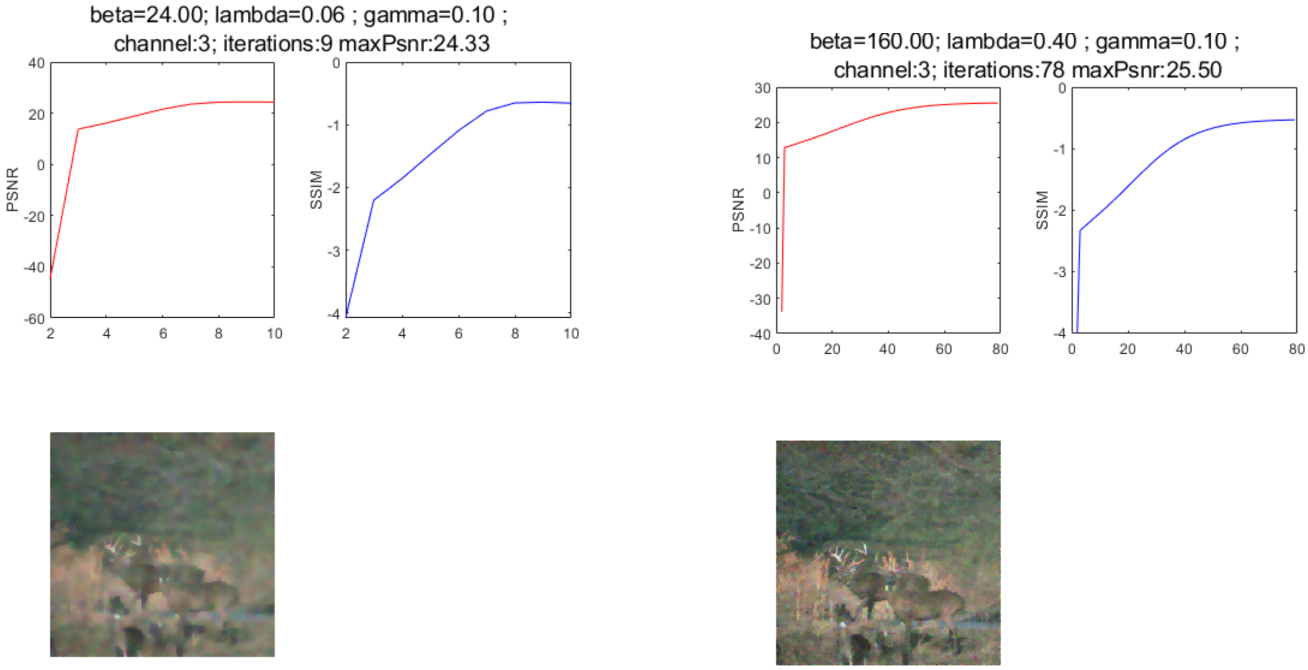

Figure 1: Result of two setups with different parameter settings

λ is the weight for the data term in our objective function and β is the weight for the relaxed constrained of the auxiliary variable v. Therefore, increasing λ and/or β induces less weight on the TV regularization which aims for sharper images but might not filter out noise efficiently.
Figure 1 compares the result of two parameter setups. In the first setting λ and β are rather small whereas in the second setting they take larger values. There is no great difference in the achieved PSNR however the setting with larger parameter values is indeed more blurry than the other setup which also does not seem to still contain much

noise.λ and β also affect the number of total iterations of the algorithm in the way, that larger values infer more iterations.

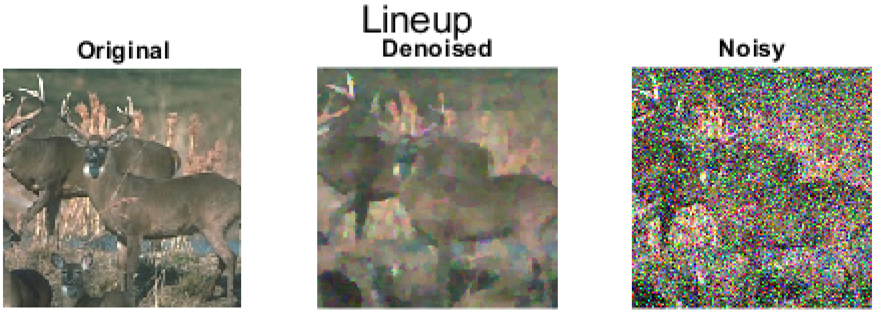

Figure 2: Comparison of result in right figure in 1 vs true and degraded image (zoomed in)

Figure 3 shows the results for the primal dual algorithm for the anisotropic TV as described in [3], solving the problem

$$\arg\min_{\mathbf{u}} \lambda \left\| \boldsymbol{\nabla}_h u \right\|_1 + \frac{\left\| u - f \right\|^2}{2} \tag{37}$$

Which is the classical objective for TV regularization, paired with a data term. We let the algorithm iterate for 200 iterations. The denoised image still contains a lot of noise while it is also quite blurry. This is mirrored also in the PSNR value which is considerably lower than the values achieved by the proposed ADMM.

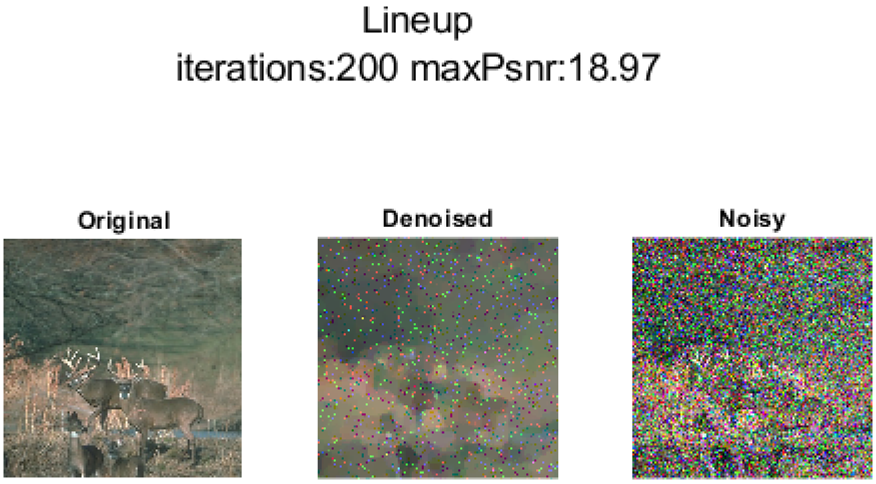

Figure 3: In this setting we applied the primal dual algorithm in [3] for anisotropic TV denoising to the objective function $\lambda \left\| \boldsymbol{\nabla}_h u \right\|_1 + \frac{\left\| u - f \right\|^2}{2}$

Table 3 shows some PSNR results, together with the number of iterations needed to achieve them. It strikes that better results always need more iterations to reach their maximal PSNR value.

β	4	40	50	20	100	140	160
λ	0.01	0.1	0.5	0.01	0.2	0.2	0.4
PSNR	23.42	24.51	24.35	23.75	25.08	25.1	**25.4**
iterations	10	14	46	8	32	42	78

Table 1: Results for various parameter settings

4 Conclusion

The proposed method proves to perform quite well on the stated problem. We did not actually address an image model as

$$f = Ku + \eta \tag{38}$$

but rather as

$$f = u + \eta \tag{39}$$

since in [1] the method in [2] was used to solve the sub problem 10, but in [2] it is explicitely stated, that the involvement of K is a different problem to which, so far, they did not find a solution.

References

[1] Mei, Jin-Jin ; Dong, Yiqiu ; Huang, Ting-Zhu ; Yin, Wotao: Cauchy noise removal by nonconvex ADMM with convergence guarantees. In: *J. Sci. Comput.* 74 (2018), Nr. 2, 743–766. http://dx.doi.org/10.1007/s10915-017-0460-5. – DOI 10.1007/s10915–017–0460–5. – ISSN 0885–7474 1, 2, 3, 6, 8

[2] Chambolle, Antonin: An algorithm for total variation minimization and applications. Version: 2004. http://dx.doi.org/10.1023/B:JMIV.0000011320.81911.38. 2004. – DOI 10.1023/B:JMIV.0000011320.81911.38. – ISSN 0924–9907, 89–97. – Special issue on mathematics and image analysis 3, 5, 6, 8

[3] Beck, Amir ; Teboulle, Marc: Fast gradient-based algorithms for constrained total variation image denoising and deblurring problems. In: *IEEE Trans. Image Process.* 18 (2009), Nr. 11, 2419–2434. http://dx.doi.org/10.1109/TIP.2009.2028250. – DOI 10.1109/TIP.2009.2028250. – ISSN 1057–7149 3, 4, 7