Deblurring Poissonian images by split Bregman techniques

S. Setzer, G. Steidl, T. Teuber *

University of Mannheim, Department of Mathematics and Computer Science, A5, 68131 Mannheim, Germany

A R T I C L E   I N F O

Article history:
Received 16 July 2009
Accepted 15 October 2009
Available online 24 October 2009

Keywords:
Deblurring
Poisson noise
I-divergence
Kullback-Leibler divergence
TV
Alternating split Bregman algorithm
Douglas-Rachford splitting

A B S T R A C T

The restoration of blurred images corrupted by Poisson noise is an important task in various applications such as astronomical imaging, electronic microscopy, single particle emission computed tomography (SPECT) and positron emission tomography (PET). In this paper, we focus on solving this task by minimizing an energy functional consisting of the I-divergence as similarity term and the TV regularization term. Our minimizing algorithm uses alternating split Bregman techniques (alternating direction method of multipliers) which can be reinterpreted as Douglas-Rachford splitting applied to the dual problem. In contrast to recently developed iterative algorithms, our algorithm contains no inner iterations and produces nonnegative images. The high efficiency of our algorithm in comparison to other recently developed algorithms to minimize the same functional is demonstrated by artificial and real-world numerical examples.

1. Introduction

Blurred images corrupted by Poisson noise appear in various applications such as astronomical imaging [1], electronic microscopy [2–4], SPECT [5] and PET [6]. The MAP estimation via the negative log likelihood function resulting from Poissonian observations \( f : \Omega \to \mathbb{R} \) of a convolved image, \( \Omega \subset \mathbb{R}^2 \) bounded and measurable, leads to the following variational problem

\[
\argmin_{\nu \geq 0} \left\{ \int_D K \nu - f \log(K \nu) \, dx + \lambda R(u) \right\}, \quad \lambda > 0,
\]

where \( K \) is a convolution kernel, which is supposed to be known. Since in most applications \( f \) is an image captured by some imaging system which detects, e.g., photons or electrons, the nonnegativity constraint \( u \geq 0 \) guarantees that no negative intensities occur in the restored image. Now consider \( \nu = K \). The term

\[
I(f, \nu) := \int_D \log \frac{f}{\nu} - f + \nu \, dx = \int_D \nu - f \log \nu + f \log f - f \, dx
\]

is known as Csiszár’s I-divergence [7] or as generalized Kullback-Leibler divergence. If it is finite, it coincides with the Bregman distance [8] of the Boltzmann-Shannon entropy. Therefore, it shares the useful properties of a Bregman distance, in particular \( I(f, \nu) \geq 0 \). In this paper, we focus on the TV regularization term \( R(u) := \sup_{f, \nu} \langle f, \nu \rangle \text{div} \nu \) : \( \psi \in C_0^1(\Omega, \mathbb{R}^2), |\psi(x)| \leq 1 \ \forall x \in \Omega \), which reads for functions with (weak) derivatives in \( L_1 \) as follows:

\[
R(u) := \int_\Omega |\nabla u| \, dx.
\]

Let \( BV \) denote the space of functions of bounded variation. It is well-known that for images \( g \) corrupted by additive white Gaussian noise, the minimizer of the Rudin-Osher-Fatemi (ROF) functional [9] is defined as

\[
\argmin_{u \in BV} \left\{ \int_D \frac{1}{2} (g - u)^2 \, dx + \lambda \int_\Omega |\nabla u| \, dx \right\}, \quad \lambda > 0
\]

(2)

can be used as a denoised version of \( g \) with preserved edges. Recently, various algorithms were developed based on convex analysis to compute the minimizer of a discrete version of (2), e.g.,

- gradient descent reprojction algorithms [10], see also [11],
- Nesterov’s algorithm [12],
- the (monotone) FISTA algorithm [13],
- alternating split Bregman algorithms [14].

The efficiency of the first three (first-order) algorithms for the denoising problem increases in the given order. For a comparison of first-order algorithms see also [15–17].

In the following, we will work exclusively in a discrete setting. To this end, we are stacking pixels of digital images columnwise into vectors in \( \mathbb{R}^N \). If not stated otherwise the multiplication of vectors, their square roots, logarithms etc. are meant componentwise. We are interested in the discrete version of problem (1) given by

* Corresponding author.
E-mail addresses: setzer@kiwi.math.uni-mannheim.de (S. Setzer), steidl@math.uni-mannheim.de (G. Steidl), tteuber@kiwi.math.uni-mannheim.de (T. Teuber).
This is usually fulfilled for blur matrices. The functional in (3) is nonnegative, this does not imply that there exists a unique minimizer of (3). In [19] Sawatzky et al. proposed an EM-TV algorithm. This can be done by one of the above-mentioned algorithms. An algorithm combining alternatingly an EM step and a TV step as follows:

**Algorithm (EM-TV)**

Initialization: $u^{(0)} = f$.

For $k = 1, \ldots$, repeat until a stopping criterion is reached

$$u^{(k+1)} = u^{(k)} K^* \left( \frac{f}{Ku^{(k)}} \right) \quad \text{(EM step),}$$

$$u^{(k+1)} = \arg\min_{u \in \Omega} \left\{ \frac{1}{2} \sum_{i=1}^N \left( \frac{u^{(k+1)} - v_i}{u_i^{(k)}} \right)^2 + \lambda \phi(Dv) \right\} \quad \text{(TV step).}$$

Indeed, the TV step requires us to minimize an ROF (like) functional

$$\arg\min_w \left\{ \frac{1}{2} \| w - h \|^2_2 + \lambda \phi(Daw) \right\}, \quad w := \frac{v}{\sqrt{u^{(k)}}, \quad h := \frac{u^{(k+1)}}{\sqrt{u^{(k)}}},$$

where $A := \text{diag}\left(\sqrt{u^{(k)}}\right)$.

This can be done by one of the above-mentioned algorithms. An analytical examination of the EM-TV algorithm was announced in [21].

Our approach is mainly based on the Poisson image deconvolution by augmented Lagrangian (PIDL) algorithm of Figueiredo and Bioucas-Dias in [22]. Both algorithms can be deduced by alternating split Bregman techniques as described in Section 3. One major difference consists in the different operator splittings. As a consequence, our algorithm avoids inner iteration loops and the associated problem of choosing an appropriate stopping rule. By contrast, the PIDL algorithm needs an inner iteration loop for computing the minimizer of a discrete ROF functional similar to the EM-TV algorithm. Moreover, it only ensures that $Ku^{(k)}$ and finally $u^{(k)}$ is nonnegative. However, even if $K$ has only nonnegative entries, this does not imply that $u$ is nonnegative. In our algorithm the iterates $u^{(k)}$ fulfill indeed a nonnegativity constraint. Numerical examples show that our new algorithm is more efficient than the EM-TV and the PIDL algorithms. For another algorithm for removing Poisson noise which uses operator splittings we refer to [23].

This paper is structured as follows. In Section 2, we review the alternating split Bregman algorithm from the point of view we need for our settings. Then, we apply the results to obtain the known PIDAL algorithm and our novel algorithm in Section 3. In Section 4, we illustrate the good performance of our algorithm by numerical experiments. In particular, we compare our algorithm with the EM-TV algorithm and the PIMAL algorithm.

### 2. Alternating split Bregman algorithm

Let $\mathbb{R}$ denote the extended real numbers $\mathbb{R} \cup \{+\infty\}$. The alternating split Bregman algorithm was introduced in [14] to efficiently solve problems of the form

$$\arg\min_{u \in \mathbb{R}^N, \phi(Cu) \in \mathbb{R}} \left\{ \psi(u) + \phi(Cu) \right\},$$

where $C \in \mathbb{R}^{N \times N}$ and $\phi : \mathbb{R}^N \to \mathbb{R}$ and $\psi : \mathbb{R}^M \to \mathbb{R}$ are proper, convex and l.s.c. functionals. In the following, we assume that (6) has a solution and

$$0 \in \text{int}(\text{dom} \psi - \text{dom} \phi), \quad 0 \in \text{int}(\text{dom} \psi + C \text{dom} \phi).$$

To derive the alternating split Bregman algorithm, we consider the following constrained version of (6), cf. [24].

$$\min_{u \in \mathbb{R}^N, \phi(Cu) \in \mathbb{R}} \{ \psi(u) + \phi(Wu) \} \quad \text{s.t.} \quad Cu = w.$$  

(7)

This constrained problem can be solved via the following split Bregman method which is the same as the augmented Lagrangian method applied to (7):

$$u^{(k+1)} = \arg\min_{u \in \mathbb{R}^N, \phi(Cu) \in \mathbb{R}} \left\{ \psi(u) + \phi(Cu) + \frac{1}{2}\lambda \| b^{(k)} + Cu - w \|_2^2 \right\},$$

$$b^{(k+1)} = b^{(k)} + Cu^{(k+1)} - w^{(k+1)}.$$  

Minimizing alternatingly with respect to $v$ and $w$ yields the alternating split Bregman method, introduced in [14]:

$$u^{(k+1)} = \arg\min_{u \in \mathbb{R}^N} \left\{ \psi(u) + \frac{1}{2}\lambda \| b^{(k)} + Cu - w^{(k)} \|^2_2 \right\},$$

$$w^{(k+1)} = \arg\min_{w \in \mathbb{R}^M} \left\{ \phi(Wu) + \frac{1}{2}\lambda \| b^{(k)} + Cu^{(k+1)} - w \|^2_2 \right\},$$

$$b^{(k+1)} = b^{(k)} + Cu^{(k+1)} - w^{(k+1)}.$$  

(9)

As Esser pointed out in [25], this algorithm can be traced back to the alternating direction method of multipliers proposed in [26,27,33]. Interestingly, there exists a connection of (8)-(10) to the Douglas-Rachford algorithm applied to the dual problem of (6)

$$\tilde{p} = \arg\min_{p \in \mathbb{R}^M} \{ \phi'(C_p - p) + \phi(p) \},$$

where $\phi'$ denotes the conjugate function of $\phi$. To derive the Douglas-Rachford algorithm for this problem, we rewrite (11) using Fermat’s rule and that $0 \in \text{int}(\text{dom} \psi' + C \text{dom} \phi)$, which gives

$$0 \in A(p) + B(\tilde{p}).$$  

(12)

where $A := \partial(\phi' \circ (-C))$ and $B := \partial \phi'$. Since $\phi$ and $\psi$ are proper, convex and l.s.c., the set-valued operators $A$ and $B$ are maximal monotone and their resolvents defined by $J_{\lambda} := (I + A)^{-1}$ are single-valued again, cf. [28]. Then, the Douglas-Rachford splitting algorithm reads as follows:

**Algorithm (Douglas-Rachford)**

Initialization: $r^{(0)}, p^{(0)}$.

For $k = 1, \ldots$, repeat until a stopping criterion is reached

$$r^{(k+1)} = J_{\lambda}\left(2p^{(k)} - r^{(k)}\right) + r^{(k)} - p^{(k)},$$

$$p^{(k+1)} = J_{\lambda}\left(r^{(k+1)}\right).$$  

It is well-known, see [29-32], that in finite-dimensional spaces the sequences $(r^{(k)})_{k \in \mathbb{N}}$ and $(p^{(k)})_{k \in \mathbb{N}}$ converge for any initialization,
where the latter converges to a solution of (12). The relation between the alternating split Bregman algorithm and the Douglas-Rachford algorithm is given as follows, see [31,33,34].

**Proposition 2.1.** For \( \eta := 1/\gamma \) and the starting values \( t^{(0)} := \eta (b^{(0)} + w^{(0)}) \) and \( p^{(0)} := \eta b^{(0)} \), the Douglas-Rachford splitting algorithm coincides with the alternating Split Bregman algorithm (8)-(10) in the sense that 

\[
t^{(k)} = \eta (b^{(k)} + w^{(k)}), \quad p^{(k)} = \eta b^{(k)}, \quad k \geq 0.
\]

Using this relation and the known convergence properties of the Douglas-Rachford splitting algorithm we obtain with a little more effort concerning the convergence of \( u^{(k)} \) the following convergence result for the alternating split Bregman algorithm, see [31,33,34].

**Proposition 2.2.** For all starting values and any step sizes \( \gamma > 0 \), the sequences \( (b^{(k)})_{k \in \mathbb{N}} \) and \( (w^{(k)})_{k \in \mathbb{N}} \) generated by the alternating split Bregman method (8)-(10) converge. Denote by \( b \) and \( w \) the limits of these sequences. Then, \( b + w \) is a solution of the dual problem (11). Furthermore, the sequence \( (u^{(k)})_{k \in \mathbb{N}} \) calculated by (8) converges to a solution of the primal problem (6) if one of the following conditions are fulfilled:

(i) The primal problem has a unique solution.

(ii) The theorem \( \min_{u \in \mathbb{R}^N} \left\{ \Psi(u) + \frac{1}{2\gamma} \| b + Cu - w \|_2^2 \right\} \) has a unique solution.

For our applications we consider instead of (6) the more general primal problem

\[
\min_{u \in \mathbb{R}^N} \sum_{i=1}^m f_i(C_i u),
\]

where \( C_i \in \mathbb{R}^{m \times N} \) and \( f_i : \mathbb{R}^N \to \mathbb{R} \). Now, we can apply the splitting idea even further and write problem (13) as

\[
\min_{w_i \in \mathbb{R}^N, w \in \mathbb{R}^N} \left\{ 0, u \right\} + \sum_{i=1}^m f_i(w_i) \quad \text{s.t.} \quad w_i = C_i u.
\]

Setting \( \Psi := (0, \cdot) \) and \( \Phi(w) := \sum_{i=1}^m f_i(w_i) \) in (8)-(10) we obtain the following alternating split Bregman algorithm

\[
\begin{align*}
&u^{(k+1)} = \min_{u \in \mathbb{R}^N} \left\{ (0, u) + \frac{1}{2\gamma} \left\| \begin{pmatrix} C_1 u \\ \vdots \\ C_m u \end{pmatrix} - \begin{pmatrix} b^{(k)} \\ \vdots \\ w^{(k)} \end{pmatrix} \right\|^2 \right\} \\
&= \begin{pmatrix} C_1 u^{(k+1)} \\ \vdots \\ C_m u^{(k+1)} \end{pmatrix} = \min_{w_i \in \mathbb{R}^N} \sum_{i=1}^m f_i(w_i) + \frac{1}{2\gamma} \left\| \begin{pmatrix} b^{(k)} \\ \vdots \\ w^{(k)} \end{pmatrix} \right\|^2 \\
&= \begin{pmatrix} C_1 u^{(k+1)} \\ \vdots \\ C_m u^{(k+1)} \end{pmatrix} = \min_{w_i \in \mathbb{R}^N} \sum_{i=1}^m f_i(w_i) + \frac{1}{2\gamma} \left\| \begin{pmatrix} b^{(k)} \\ \vdots \\ w^{(k)} \end{pmatrix} \right\|^2 \\
&= \begin{pmatrix} C_1 u^{(k+1)} \\ \vdots \\ C_m u^{(k+1)} \end{pmatrix} = \min_{w_i \in \mathbb{R}^N} \sum_{i=1}^m f_i(w_i) + \frac{1}{2\gamma} \left\| \begin{pmatrix} b^{(k)} \\ \vdots \\ w^{(k)} \end{pmatrix} \right\|^2 \\
&b^{(k+1)} = b^{(k)} + C_1 u^{(k+1)} - w^{(k+1)}.
\end{align*}
\]

The variables \( w_i \) in (16) are not coupled so that we can minimize separately with respect to the \( w_i, i = 1, \ldots, m \). As already described in [25], this is very useful if more than one functional in the primal problem is a concatenation of a proper, convex and l.s.c. function with a linear operator. Let us have a brief look at the corresponding Douglas-Rachford splitting algorithm. Since

\[
\Psi'(x) = (0, \cdot)'(x) = \begin{cases} 0 & \text{if } x = 0, \\ +\infty & \text{if } x \neq 0, \end{cases}
\]

and \( \Phi'(x) = \sum_{i=1}^m f_i'(x_i) \) the operators \( A \) and \( B \) in (12) for the corresponding Douglas-Rachford splitting algorithm are \( A(p) := \partial((0, \cdot)' \circ (-C_1 \cdots C_m))(p) \) and \( B(p) := (\partial g_1(p_1), \ldots, \partial g_m(p_m)) \). More specifically, since

\[
\partial((0, \cdot)'(x)) = \begin{cases} \partial g^N & \text{if } x = 0, \\ 0 & \text{if } x \neq 0, \end{cases}
\]

we get for \( A \)

\[
A(p) = - \begin{pmatrix} C_1 \\ \vdots \\ C_m \end{pmatrix} \partial((0, \cdot)'(-C_1 \cdots C_m)(p))
\]

\[
= \begin{cases} \left( \begin{array}{c} C_1 \\ \vdots \\ C_m \end{array} \right) q_1 q \in \mathbb{R}^N & \text{if } \sum_{i=1}^m C_i p_i = 0, \\ 0 & \text{if } \sum_{i=1}^m C_i p_i \neq 0. \end{cases}
\]

By the definition of \( A \), the resolvent \( J_{\lambda A} \) in the Douglas-Rachford splitting algorithm is the orthogonal projection onto the null space of \( (C_1 \cdots C_m) \), i.e., for any \( p \in \mathbb{R}^{m \times (m+1)} \) we have

\[
J_{\lambda A}(p) = P_{\mathbb{R}^N}(C_1 \cdots C_m) = p - \left( \begin{array}{c} C_1 \\ \vdots \\ C_m \end{array} \right) \argmin_{q \in \mathbb{R}^N} \left\{ \left\| \begin{array}{c} C_1 \\ \vdots \\ C_m \end{array} \right\| y - p \right\|^2.
\]

Observe that formula (18) has already the structure of (15).

By Proposition 2.2, the sequences \( (b^{(k)})_{k \in \mathbb{N}} \) and \( (w^{(k)})_{k \in \mathbb{N}} \) generated by the alternating split Bregman method (16), (17) converge, in particular, \( b^{(k)} \) converges to a solution of the dual problem. The sequence \( (u^{(k)})_{k \in \mathbb{N}} \) converges to the solution of the primal problem (11) if either this primal problem has a unique solution or \( \sum_{i=1}^m C_i C_i^T \) is invertible.

### 3. Deblurring methods for Poissonian images

In [22], problem (3) was considered without the constraint \( u \geq 0 \). With the notation from the previous section the authors apply the alternating split Bregman algorithm (15)-(17) for \( m = 2 \) and

\[
\argmin_{u \in \mathbb{R}^N} \left\{ (I, Ku - f \log(Ku)) + \lambda \tilde{\phi}(Du) \right\}
\]

i.e., \((C_1, C_2) := (K, I)\). In particular, we have the constraint \( w_2 = u \). Separating the minimization with respect to the variables \( w_i \) and \( u = w_2 \) in step (16) this leads to the following PIDAL algorithm:

**Algorithm (PIDAL)**

**Initialization:** \( b^{(0)} = b_0^{(0)} = 0, w_0^{(0)} = k_{\gamma}, w_2^{(0)} = f \).

For \( k = 1, \ldots, r \) repeat until a stopping criterion is reached

\[
\begin{align*}
&u^{(k+1)} = \argmin_{u \in \mathbb{R}^N} \left\{ \left\| b^{(k)} + Ku - w^{(k)} \right\|_2^2 + \left\| b^{(k)} + u - w_2^{(k)} \right\|_2^2 \right\}, \\
&w_i^{(k+1)} = \argmin_{w_i \in \mathbb{R}^N} \left\{ \left(1, w_1 - f \log(w_1) \right)^{\gamma} \right\}_{1}^{K} + \frac{1}{2\gamma} \left\| b^{(k)} + Ku - w_i^{(k+1)} \right\|_2^2 \\
&\quad + \frac{1}{2\gamma} \left\| b^{(k+1)} + w_i^{(k+1)} - w_i^{(k)} \right\|_2^2 \\
&b^{(k+1)} = b^{(k)} + Ku^{(k+1)} - w^{(k+1)}.
\end{align*}
\]
\[ w^{(k+1)} = \arg \min_{w \in \mathbb{R}^n} \left\{ \lambda \phi(Dw_2) + \frac{1}{2 \gamma} \| b^{(k)}_2 + u^{(k-1)} - w_2 \|^2 \right\}, \]  
(19)

\[ b^{(k)}_1 = b^{(k)}_1 + Ku^{(k-1)} - w_1^{(k-1)}, \]
\[ b^{(k)}_2 = b^{(k)}_2 + u^{(k-1)} - w_2^{(k-1)}. \]

Output: \( u^{\text{approx}} := w_2^{(k+1)}. \)

To be consistent with [22], we set our final result to be \( w_2^{(k+1)} \) instead of \( u^{(k)} \) which is justified by the constraint \( u = w_2 \). The first two steps of the PIDAL algorithm can be solved explicitly:

\[ u^{(k+1)} = (I + K'K)^{-1} \left( \tilde{K} (w_1^{(k)} - b^{(k)}_1) + \tilde{w}_2^{(k)} - b^{(k)}_2 \right), \]

\[ w_1^{(k+1)} = \left\{ \begin{array}{ll}
\frac{1}{2} (b^{(k)}_1 + Ku^{(k-1)} - \gamma + \sqrt{(b^{(k)}_1 + Ku^{(k-1)} - \gamma)^2 + 4\gamma}) & \text{if} \ b^{(k)}_1 + u^{(k-1)} \geq 0, \\
0 & \text{otherwise}.
\end{array} \right. \]

Note that \( I + K'K \) is positive definite and therefore invertible. Moreover, supposing Neumann boundary conditions for our images, the corresponding linear system of equations can be solved efficiently by the discrete cosine transform (DCT-II) requiring only \( O(N \log N) \) arithmetic operations. In the third step (19), we have to solve a ROF denoising problem via one of the iterative algorithms mentioned in the introduction. The PIDAL algorithm ensures that the sequence \( \{w_n^{(k)}\} \in \mathbb{N} \) is nonnegative so that it converges to a nonnegative image \( w_1 = K' \). Unfortunately, this does not imply that it itself is nonnegative even if \( K \) has only nonnegative entries summing up columnwise to 1.

The idea of our new method is to use the special structure of the regularization functional \( \phi * D \) in (3) and to exploit the splitting idea even further. Moreover, we impose a nonnegativity constraint on \( u \). Let \( \omega \in \mathbb{R}^n \) denote the indicator function of \( \mathbb{R}^n \). We apply the alternating split Bregman algorithm (15)-(17) to

\[ \arg \min_{w \in \mathbb{R}^n} \left\{ \frac{1}{2} \| Ku - f \log(Ku) \|^2 + \lambda \phi(Du) + u \geq 0 \right\} \]

i.e., \( m = 3 \) and \( (C_1, C_2, C_3) = (K, D, I) \). In particular, we have the constraint \( w_3 = u \). The corresponding alternating split Bregman algorithm, which we call PIDSplit+, is given below. For speed comparisons we also apply the algorithm without the positivity constraint, i.e., with \( m = 2 \) and \( (C_1, C_2) = (K, D) \) and call this variant the PIDSplit algorithm.

**Algorithm (PIDSplit+)**

**Initialization:** \( b_1^{(0)} = b_2^{(0)} = b_3^{(0)} = 0, w_1^{(0)} = Kf, w_2^{(0)} = Dw, w_3^{(0)} = f. \)

For \( k = 1, \ldots \), repeat until a stopping criterion is reached

\[ u^{(k+1)} = \arg \min_{u \in \mathbb{R}^n} \left\{ \| b^{(k)}_1 + Ku - w_1^{(k)} \|^2 + \| b^{(k)}_2 + Du - w_2^{(k)} \|^2 + \| b^{(k)}_3 + u - w_3^{(k)} \|^2 \right\}, \]

\[ w_1^{(k+1)} = \arg \min_{w_1 \in \mathbb{R}^n} \left\{ \frac{1}{2} \left( w_1 \cdot f - \log(w_1) \right) + \frac{1}{2 \gamma} \| b^{(k)}_1 + Ku^{(k-1)} - w_1 \|^2 \right\}, \]

\[ w_2^{(k+1)} = \arg \min_{w_2 \in \mathbb{R}^n} \left\{ \frac{1}{2} \left( w_2 \cdot f - \log(w_2) \right) + \frac{1}{2 \gamma} \| b^{(k)}_2 + Du^{(k-1)} - w_2 \|^2 \right\}, \]

\[ w_3^{(k+1)} = \arg \min_{w_3 \in \mathbb{R}^n} \left\{ \frac{1}{2} \left( w_3 \cdot f - \log(w_3) \right) + \frac{1}{2 \gamma} \| b^{(k)}_3 + u^{(k-1)} - w_3 \|^2 \right\}, \]

\[ b^{(k)}_1 = b^{(k)}_1 + Ku^{(k-1)} - w_1^{(k-1)}, \]
\[ b^{(k)}_2 = b^{(k)}_2 + Du^{(k-1)} - w_2^{(k-1)}, \]
\[ b^{(k)}_3 = b^{(k)}_3 + u^{(k-1)} - w_3^{(k-1)}. \]

Output: \( u^{\text{approx}} := w_3^{(k)}. \)

Note that \( w_3^{(k)} \) is nonnegative for all \( k \), which is one reason to prefer \( w_3^{(k)} \) over \( u^{(k)} \). Thus, in contrast to the PIDAL algorithm we guarantee that \( u^{\text{approx}} \geq 0 \) and we no longer need to solve an ROF denoising problem as inner iteration loop. Instead, \( w_1^{(k-1)}, w_2^{(k-1)}, w_3^{(k)} \) can be computed by direct formulas and to get \( u^{(k)} \) we solve a linear system of equations:

\[ u^{(k)} = (I + K'K + D'D)^{-1} \left( K^T (w_3^{(k)} - b_3^{(k)}) + D^T (w_2^{(k)} - b_2^{(k)}) + (w_1^{(k)} - b_1^{(k)}) \right), \]

\[ w_1^{(k-1)} = \frac{1}{2} \left( b^{(k)}_1 + Ku^{(k-1)} - \gamma + \sqrt{(b^{(k)}_1 + Ku^{(k-1)} - \gamma)^2 + 4\gamma} \right), \]

\[ w_2^{(k-1)} = \text{shrink}_{\gamma}(b_2^{(k)} + Du^{(k-1)}), \]

\[ w_3^{(k)} = \left\{ \begin{array}{ll}
b^{(k)}_3 + u^{(k-1)} & \text{if} \ b^{(k)}_3 + u^{(k-1)} \geq 0, \\
0 & \text{otherwise.}
\end{array} \right. \]

The \( \text{shrink}_{\gamma} : \mathbb{R}^{2N} \to \mathbb{R}^{2N} \) operator denotes the coupled soft shrinkage which is given componentwise for \( p := (p_1, p_2) : p_i := (p_{i,j})_{j=1}^{N}, i = 1, 2, \)

\[ \text{shrink}_{\gamma}(p) := \left\{ \begin{array}{ll}
p_i - \gamma p_{i,j}/\sqrt{p_{i,j}^2 + p_{j,j}^2} & \text{if} \ p_{i,j}^2 + p_{j,j}^2 \geq \gamma, \\
0 & \text{otherwise.}
\end{array} \right. \]

see, e.g., [34]. Supposing Neumann boundary conditions for our images, the matrix \( I + K'K + D'D \) can be diagonalized by the DCT-II. Thus, also in this case the solution of the corresponding linear system of equations requires only \( O(N \log N) \) arithmetic operations.

**4. Numerical results**

Finally, we present numerical examples and compare our new PIDSplit+ algorithm with the EM-TV and PIDAL algorithms in terms of computational speed. To this end, we implemented all algorithms in MATLAB and performed the computations on a dual core desktop with two 2.4 GHz processors and 2.85 GB physical memory. For adding Poisson noise to an image \( U \) given in double precision, we first scaled the image by \( U_{\max}/\text{max}(1/10^3) \), where \( U_{\max} \) is the maximal value of \( U \) and \( U_{\max} \) is a specified maximal intensity. Then, we applied the MATLAB routine \text{imnoise}(1,'poisson') \) and afterwards scaled back again. The values of the images lie all in the interval \([0, 255]\). To plot the results, we used the MATLAB routine \text{imagesc}, which incorporates an affine gray value scaling.

For solving the ROF-like subproblems (5) and (19) of the EM-TV and PIDAL algorithms in an efficient way, we tried different methods. To gain an additional speed up we applied the following initialization for the inner loops which was suggested by Brune [36]:

**Inner loop initialization:** Use the final values of an inner iteration loop as initial values for the next loop.

For solving (5), we decided to use the gradient descent reprojection algorithm [10], since it outperformed the FISTA algorithm [13] due to the fact that using the inner loop initialization described above the number of iterations needed per loop decreased faster for the gradient descent reprojection algorithm than for FISTA. Similarly, for solving the TV subproblem (19) we compared the gradient descent reprojection algorithm [10] to the alternating split Bregman algorithm in [14]. The second algorithm required to solve in each step a linear system of equations, which we solved again by using the discrete cosine transform (DCT-II). For the same reason as above, the gradient descent reprojection algorithm performed better and we use it for the experiments below. The corresponding step size was set to 0.249. Moreover, as suggested in
we used $\gamma = 50/\lambda$ for the PIDAL and also for the PIDSplit/PIDSplit+ algorithms.

For our first example in Fig. 1, we took a part of the 'cameraman' image, blurred it by a Gaussian kernel and additionally contaminated it by Poisson noise ($I_{\text{max}} = 3000$). The deblurred image is depicted in Fig. 1 (right). Since for this example the restoration result of (3) without the additional constraint $u \geq 0$ is already positive, all algorithms provide the same result. The computational times of the different algorithms are listed in Table 1. For direct comparison with the PIDAL algorithm, we also included the computation time of the PIDSplit algorithm. As stopping criterion for the inner iteration loops of the PIDAL and EM-TV algorithms, we checked if the maximal change per pixel in the image from one iteration to the next is less than an error bound $\varepsilon$. The results in Table 1 show that the PIDSplit/PIDSplit+ algorithms outperform the PIDAL and EM-TV algorithms significantly. Moreover, all algorithms need a similar number of outer iterations in this example. Only the EM-TV algorithm needed around two times as many.

A restoration result and a comparison of the speed of the PIDSplit+ and EM-TV algorithms for the whole 'cameraman' image is given in Fig. 2 and Table 2. Note that for these results only a maximal pixel difference of 0.3 to a converged reference image was required.

To apply our PIDSplit+ algorithm also to real confocal data, our last example in Fig. 3 shows at the top left an image of a blurred human cell nucleus of size $401 \times 311$ provided by Prof. T. Cremer, LMU Munich. We thank S. Remmele for estimating the blur kernel to be a symmetric Gaussian of standard deviation 3.8. For this example, much more iterations are necessary to obtain equally small errors as for the 'cameraman' image. However, we get already after a few iterations a visually pleasant result as illustrated in Fig. 3. Here, it is also interesting to mention that if we solved model (3) for, e.g., $\lambda = 0.002$ without the additional nonnegativity constraint, the result would contain negative value down to $-249$ although the blurred image was nonnegative.

### 5. Summary and conclusions

We have proposed a simple and very efficient algorithm for the restoration of blurred images corrupted with Poissonian noise by minimizing the TV penalized I-divergence. We use alternating split Bregman techniques (alternating direction method of multipliers) to decouple this problem in such a way that inner loops do not oc-
cur. Thus, in contrast to other algorithms, we avoid the choice of additional step length parameters and appropriate stopping rules. Further, our algorithm produces nonnegative restored images and the convergence is guaranteed by known convergence results of the alternating split Bregman algorithm.

Acknowledgment

T.T. gratefully acknowledges partial support by DFG Grant Ste 571/9-1.

References


