Convex Multi-Class Image Labeling by Simplex-Constrained Total Variation

Lellmann, Jan and Kappes, Jörg and Yuan, Jing and Becker, Florian and Schnörr, Christoph

Convex Multi-Class Image Labeling by Simplex-Constrained Total Variation

Technical Report, October 2008

Jan Lellmann, Jörg Kappes, Jing Yuan, Florian Becker, and Christoph Schnörr

Image and Pattern Analysis Group (IPA)
Heidelberg Collaboratory for Image Processing (HCI)
Dept. of Mathematics and Computer Science, University of Heidelberg
{lellmann,kappes,yuanjing,becker,schnoerr}@math.uni-heidelberg.de,
http://ipa.iwr.uni-heidelberg.de/

Abstract. Multi-class labeling is one of the core problems in image analysis. We show how this combinatorial problem can be approximately solved using tools from convex optimization. We suggest a novel functional based on a multidimensional total variation formulation, allowing for a broad range of data terms. Optimization is carried out in the operator splitting framework using Douglas-Rachford Splitting. In this connection, we compare two methods to solve the Rudin-Osher-Fatemi type subproblems and demonstrate the performance of our approach on single- and multichannel images.

1 Introduction

1.1 Overview, Motivation

In this paper, we study the variational approach

$$\inf_{u \in C} f(u) \ , \quad f(u) = -\int_{\Omega} \langle u(x), s(x) \rangle dx + \lambda \operatorname{TV}(u) \ , \quad \lambda > 0 \ , \tag{1}$$

for determining a labeling $u : \Omega \to \mathbb{R}^L$, that is a contextual classification of each pixel $x \in \Omega$ into one out of L classes, based on an arbitrary vector-valued similarity function $s(x) \in \mathbb{R}^L$ as input data that has been computed from image data beforehand.

The objective function (1) comprises the common form of a data term plus a regularization term. The data term is given by the L^2 inner product of the assignment variables u and the similarity function s, and the regularizer is a total variation (TV) formulation for vector-valued data,

$$\operatorname{TV}(u) = \int_{\Omega} \sqrt{\|\nabla u_1\|^2 + \cdots + \|\nabla u_L\|^2} dx \ . \tag{2}$$

Preprint of the IWR, University of Heidelberg, October 2008.

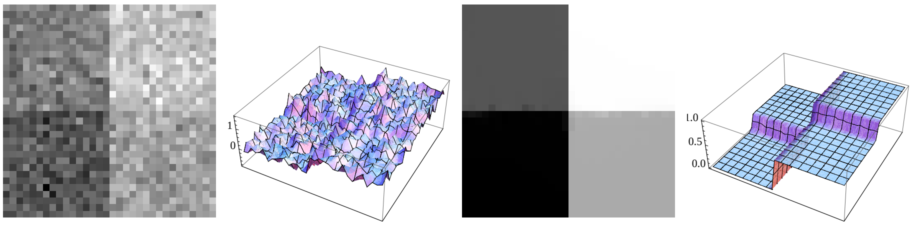

Fig. 1. Left: Noisy input image. **Right:** The labeled image based on the non-binary assignment u as global minimizer of the convex approach (1). The discrete problem is accurately solved by a continuous approach.

Furthermore, the constraint $u \in C$ restricts the vector field $u(x)$ at each location $x \in \Omega$ to lie in the standard probability simplex, that is $u(x) \in \mathbb{R}^L_+$ and $\sum_{i=1}^L \big(u(x)\big)_i = 1$ for all $x \in \Omega$.

Our work is motivated by the following observation. Suppose that at each pixel $x \in \Omega$, there is an *unambiguous* assignment (labeling) of the data $s(x)$ to some class $l \in \{1, \dots, L\}$ represented by the corresponding l-th unit vector, $u(x) = e^l$. Then, an interface with area A between two image regions labeled with l and l', respectively, adds $A\sqrt{2}$ to the regularization term iff $l \neq l'$, because then all but two gradients under the square root vanish. As a result, under these assumptions and up to the immaterial constant $\sqrt{2}$, the TV term corresponds to the well-known Potts model that assigns constant penalties to local changes of the labeling.

A significant difference between the Potts model and our approach (1), however, is that the former amounts to solve a *discrete combinatorial* problem, whereas the latter is a *continuous convex* optimization problem. Experiments show that our approach (1) approximates *discrete* decisions fairly well – see Figures 1 and 2 – by computing a global optimum to a single convex optimization problem. By contrast, the state-of-the-art discrete approach [1] approximates the combinatorial solution by solving a non-uniquely defined *sequence* of globally optimal binary problems via graph cuts. This fact, along with the potential of continuous convex optimization for parallel implementations and their more robust dependency on (hyper-) parameters, motivated to investigate the approach (1) as a promising model for a general "labeling submodule" within computer vision systems. To this end,

- We have a closer look at the data and regularization terms (section 2).
- We apply an operator splitting approach to (1) in order to decompose the computation of a globally optimal labeling into two independent computational steps: TV denoising for vector-valued data, and projection of the labeling vectors $u(x)$ on the canonical simplex. Iterating a suitable combination of these steps ensures convergence to a global optimum (section 3).
- We evaluate two different algorithms for the TV denoising subroutine (section 4) and compare the performance of our convex method to a range of established graph cut-based approaches (section 5).

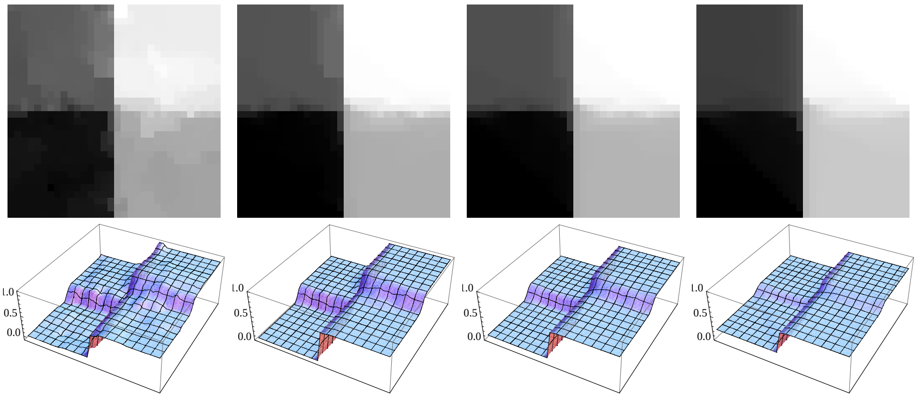

Fig. 2. Output of the standard TV approach [2] for scalar-valued images applied to the noisy input image depicted in Figure 1, for different values of the regularization parameter λ. Irrespective of this value, the performance is worse than with the approach (1) (cf. Fig. 1, right), because the latter approximates the Potts model that does *not* depend on the size (contrast) of discontinuities. Consequently, the former approach cannot remove noise without degrading weak discontinuities, as apparent above for the horizontal discontinuities.

1.2 Related Work

Many publications on TV-based segmentation are focused on the fully discrete setting, which – under anisotropic discretization – can be solved using graph cuts. Numerous algorithms have been proposed in this field, see e.g. [3] for a reference. Graph cut-based algorithms require submodularity of the energy function in order to find a global optimum [4]. While this criterion is often met for binary segmentation, multi-class labeling requires extensions which usually rely on solving a sequence of (binary) graph cuts to find a local minimum [1]. Also, anisotropic discretizations exhibit a bias for edges in some directions, depending on the neighborhood definition.

In the continuous setting, binary TV-regularized segmentation corresponds to finding a characteristic function which minimizes the objective function, also called *continuous cut* [5].

Interestingly, Nikolova et al. [6] showed that this problem can be relaxed and solved on a convex set, while still allowing to reconstruct a *true* binary solution. Our work is motivated by their approach, but is aimed at the multi-class case.

In [7], an approach comparable to ours was presented, based on Ishikawa's analysis in [8]. However, their regularization term prefers transitions between "nearby" labels as measured by the *label index.* While this can be a desirable property, e.g. in stereo reconstruction, our formulation is more suited to the case when there is no natural label ordering, as in the Potts model.

The Potts model was studied as a special case of the metric labeling problem in [9]. The corresponding energy functional relates to (1) in the discrete case and for an anisotropic discretization of the TV term. Approximate solutions were computed by an LP relaxation with explicit constraints. In contrast, our approach considers the general TV term and "sparse numerics" through a problem decomposition into efficiently solvable subproblems, without the need to introduce any additional variables.

1.3 Notation

We consider the *discretized* version of our approach (1). Let $\Omega = \{1, \ldots, n_1\} \times \cdots \times \{1, \ldots, n_d\} \subseteq \mathbb{R}^d$, $d \in \mathbb{N}$, denote a regular image grid of $n := |\Omega|$ pixels. The (multidimensional) image space $X := \mathbb{R}^{n \times L}$ is equipped with the inner product

$$\langle u, s \rangle_\Omega = \sum_{x \in \Omega} \big\langle u(x), s(x) \big\rangle ,$$

where $\langle \cdot, \cdot \rangle$ is the usual Euclidean inner product. We naturally identify $v = (v^1, \ldots, v^L) \in R^{n \times L}$ with $((v^1)^\top \cdots (v^L)^\top)^\top \in \mathbb{R}^{nL}$. Using the notation $e = (1, 1, \ldots, 1)^\top$, the standard simplex on $\mathbb{R}^L$ and its extension C on $\mathbb{R}^{n \times L}$ are given by

$$\Delta_L = \big\{ v \in \mathbb{R}^L \,\big|\, v \geq 0 \,,\ \langle e, v \rangle = 1 \big\} \,, \quad C := \prod_{x \in \Omega} \Delta_L \,. \tag{3}$$

Vectors are indexed by superscripts, and scalars by subscripts, e.g. $v^1, v^2, v^3, \ldots$ denotes a collection of vectors, while v_k is the k-th component of a vector v.

As a discrete, multidimensional analogon for the continuous total variation formulation we will use the following definition: Let

$$\text{grad} := \begin{pmatrix} \text{grad}_1 \\ \vdots \\ \text{grad}_d \end{pmatrix} , \quad \text{grad}_i := I_{n_1} \otimes \ldots \otimes I_{n_{i-1}} \otimes F_{n_i} \otimes I_{n_{i+1}} \otimes \ldots \otimes I_{n_d} ,$$

be the d-dimensional forward difference gradient operator for Neumann boundary condition, where

$$F_m \in \mathbb{R}^{m \times m},\ (F_m)_{ij} = \begin{cases} -1, & j = i,\ i < m \,, \\ 1, & j = i + 1,\ i < m \,, \\ 0, & \text{otherwise} \,. \end{cases}$$

Accordingly, $\text{div} := -\text{grad}^\top$ is the backward difference divergence operator for Dirichlet boundary condition. These operators extend to $\mathbb{R}^{n \times L}$ via

$$\text{Grad} := (I_L \otimes \text{grad}) \,, \quad \text{Div} := (I_L \otimes \text{div}) \,. \tag{4}$$

We will also need the convex sets

$$B_\lambda := \Big\{ (p^1, \ldots, p^L) \in \mathbb{R}^{d \times L} \,\Big|\, \Big(\sum_{i=1}^L \| p^i \|_2^2 \Big)^{1/2} \leqslant \lambda \Big\} , \tag{5}$$

$$D_\lambda := \prod_{x \in \Omega} B_\lambda \subseteq \mathbb{R}^{n \times d \times L} , \tag{6}$$

$$E_\lambda := \{ u = (u^1, \ldots, u^L) \in \mathbb{R}^{n \times L} | u = \text{Div} p \,, p \in D_\lambda \} \,. \tag{7}$$

The discrete total variation on vector-valued data is then defined as

$$\mathrm{TV}(u) := \sigma_{E_1}(u) \ , \tag{8}$$

where $\sigma_M(u) := \sup_{p \in M} \langle u, p \rangle$ is the support function from convex analysis. We further define $\delta_C(x)$ to be 0 iff $x \in C$, and $+\infty$ otherwise.

2 Variational Approach

Based on the introduced notation, our novel approach (1) reads

$$\inf_{u \in C} f(u) \ , \quad f(u) = \underbrace{-\langle u, s \rangle_\Omega}_{\text{data term}} + \underbrace{\lambda \, \mathrm{TV}(u)}_{\text{regularization term}} \ , \quad \lambda > 0 \ , \tag{9}$$

As the objective function f and the constraint set C (see (3)) are convex, the overall problem is convex as well. We will now define and motivate each term.

2.1 Data Term

The data term in (9) is fairly general. Any vector-valued similarity function s can be used, whose components $\big(s(x)\big)_i$ indicate the affinity of some data point at x with class i. We consider an example.

Suppose we have image features $g(x)$, $x \in \Omega$, and are given prototypical feature vectors $G = (G^1, \ldots, G^L)$ as well as a distance measure d on the features. We might think of g as a grayscale image, of G as some prototypical gray values, and of d as a quadratic distance measure, possibly derived from a statistical noise model.

The hard assignment of the pixel $x \in \Omega$ to a label (or class) $l(x) \in \{1, \ldots, L\}$ should then be penalized by the distance $d(g(x), G^{l(x)})$ of the corresponding feature to the prototype of the assigned class. Denoting the negative distance by s in order to comply with our affinity notation, and summing up over the image domain, we see that

$$\sum_{x \in \Omega} d(g(x), G^{l(x)}) = - \sum_{x \in \Omega} \big\langle s(x), u(x) \big\rangle \quad \text{for } u(x) = e^{l(x)} \ . \tag{10}$$

Thus, instead of looking for $l \in \{1, \ldots, L\}^n$, we may equivalently look for $u \in \{e^1, \ldots, e^L\}^n$. However, the right hand side formulation has the advantage that it extends naturally to the *soft* assignment $u \in C$: we may now solve the easier problem of optimizing for u on the *convex* set C.

In our experiments, we chose $d(x, y) = ||x - y||_1$ for grayscale as well as color images, as the ℓ_1-norm is still convex but known to be more robust against noise and outliers. However, s is not restricted to representing distances. More generally, all data terms of the form

$$\sum_{x \in \Omega} h_x(g, l(x)) \tag{11}$$

are covered by our approach if one sets $(s(x))_l := -h_x(g, l)$. The h_x correspond to the unary potentials in the discrete MRF formulation. This formulation has the appealing property that h_x can be arbitrarily nonlinear and nonconvex, and involve nonlocal operations on g. The complexity of the similarity measure is completely hidden within the precomputed vector s.

2.2 Regularization Term

Using the total variation definition (8), we see that the regularizer of (9) is defined as

$$\mathrm{TV}(u) = \sup_{p \in D_1} \langle u, \mathrm{Div} p \rangle \ . \tag{12}$$

In view of the definitions (6), TV can be directly expressed as

$$\mathrm{TV}(u) = \sup_{p \in D_1} \langle \mathrm{Grad}\, u, p \rangle = \sum_{x \in \Omega} \|G_x u\|_2 \tag{13}$$

where G_x is an $(Ld) \times n$ matrix composed of rows of (Grad) s.t. $G_x u$ gives the gradients of all u_i in x stacked one above the other.

This definition for vector-valued u parallels the definition of the "isotropic" total variation measure in the scalar-valued case [10, 2, 11]. It is also known as *MTV* [12–14], and was recently studied in [15] in its continuous formulation. Contrary to the anisotropic discretization, where one would substitute the sum of 1-norms in (5), it is less biased towards edges parallel to the axes.

See also [16] for an overview of TV-based research and applications.

2.3 Optimality

After solving the relaxed problem, it remains to show that a binary solution can be recovered. For the continuous, *binary* case, Nikolova et al. [6] showed that an exact solution can be obtained by thresholding at almost any threshold.

However, their results do not immediately transfer to the discrete *multi*-class case. In particular, the crucial "layer cake" formula holds for ℓ_1-, but not ℓ_2 discretizations of the TV.

Contrary to the binary case, it is not clear which rounding scheme should be employed for vector-valued u. For our experiments, we chose the final class label for each pixel x as the index l of the maximal $u_l(x)$ of the global optimum u^* of (9). This defines a suboptimal discrete solution u_t^*. Bounding the error $u_t^* - u_d^*$ with respect to the unknown discrete optimum u_d^* in terms of u^* will be subject of our future work.

3 Optimization

Two basic problems arise concerning the optimization of (9):

1. Nondifferentiability of the objective function due to the TV term, and
2. handling of the simplex constraint $u \in C$.

We cope with the latter point using the tight *Douglas-Rachford* splitting method as presented in the following section.

3.1 Operator Splitting

We will state some preliminaries from the theory of maximal monotone operators, cf. [17, ch. 12]. Given a Hilbert space $\mathcal{H}$, an *operator* (or *set-valued mapping*) on $T : \mathcal{H} \rightrightarrows \mathcal{H}$ is a simply a subset $T \subseteq \mathcal{H} \times \mathcal{H}$, which assigns to each point $x \in \mathcal{H}$ a subset $T(x) := \{y \in \mathcal{H} | (x, y) \in T\}$. Operators can be *inverted*, $T^{-1} := \{(y, x) | (x, y) \in T\}$, and *added*, $(T + U) := \{(x, y + y') | y \in T(x), y' \in U(x)\}$. The *domain* is defined as $\operatorname{dom} T := T^{-1}(\mathcal{H})$.

T is said to be *monotone* iff, for all $x, x', y, y' \in T$ s.t. $y \in T(x)$, $y' \in T(x')$,

$$\langle x' - x, y' - y \rangle \geqslant 0.$$

T is *maximal monotone* iff T is monotone and there is no other operator U that is a superset of T (precisely, $T \subseteq U \Rightarrow T = U$). In what follows, we will fix $\mathcal{H} = \mathbb{R}^n$.

Subgradient mappings ∂f of proper, convex, lower semi-continuous (*lsc*) functions $f : \mathbb{R}^n \to \mathbb{R} \cup \{+\infty\}$ constitute maximal monotone operators [17, Thm. 12.17]. Accordingly, for any closed, nonempty, convex set $C \subseteq \mathbb{R}^n$, the *normal cone operator* N_C defined by $N_C(x) := \{x' \in \mathcal{H} | \forall y \in C : \langle x' - x, y \rangle \leqslant 0\}$ is maximal monotone, as $N_C = \partial \delta_C$.

Minimization of a proper, convex, lsc function $f : \mathbb{R}^n \to \mathbb{R} \cup \{+\infty\}$ amounts to finding a *zero* of its subgradient mapping $T := \partial f$, i.e. finding any $x \in (\partial f)^{-1}(0)$. There are two basic building blocks for constructing fixpoint methods to find such a zero:

- The *forward step*,
$$x^{k+1} \in (I - \tau T)(x^k), \quad \tau > 0\,, \tag{14}$$
- and the *backward step*,
$$x^{k+1} \in (I + \tau T)^{-1}(x^k) = J_{\tau T}(x^k), \quad \tau > 0\,, \tag{15}$$
where $J_{\tau T} := (I + \tau T)^{-1}$ is called the *resolvent* of T.

Applied directly to the subgradient mapping, the *forward step* is just a simple subgradient descent, and as such may suffer from nonuniqueness and nonconvergence. However, maximal monotonicity of T ensures that any fixpoint of the forward step is a zero of T, and thus solves the minimization problem.

The beauty of the *backward step* lies in the fact that resolvents of monotone operators are firmly nonexpansive: for all $x, x', y, y' \in \mathbb{R}^n$ with $y \in T(x)$ and $y' \in T(x')$,

$$\|y' - y\|^2 \leqslant \|x' - x\|^2 - \|(x' - y') - (x - y)\|^2,$$

holds. Maximality of the operator additionally ensures surjectivity of the resolvent. Both properties together make $J_{\tau T}$ single-valued and thus x^{k+1} uniquely (and always) defined. Additionally, (x^k) converges for any step size $\tau > 0$.

However, inverting $J_{\tau T}$ is generally as difficult as the original problem. In the *operator splitting* approach, T is decomposed into the sum of two "easier" maximal monotone operators, $T = A + B$, for which forward and backward steps are computationally feasible. Then a sequence is constructed which uses only forward and backward steps on A and B, but allows to find a zero of T.

Here, we consider the (tight) *Douglas-Rachford-Splitting* iteration [18, 19],

$$z^{k+1} \in \underbrace{(J_{\tau A}(2J_{\tau B} - I) + (I - J_{\tau B}))}_{=:G_{\tau,A,B}}(z^k). \tag{16}$$

Under the very general constraint that $A, B : \mathbb{R}^n \rightrightarrows \mathbb{R}^n$ are maximal monotone and $A + B$ has at least one zero, the sequence (z^k) will converge to a fixpoint z of $G_{\tau,A,B}$, with the additional property that $x := J_{\tau B}(z)$ is a zero of T ([20, Thm. 3.15], [20, Prop. 3.20], [20, Prop. 3.19]; for a well-written analysis, see [20] or [21]).

For a proper, convex, lsc function $f = f_1 + f_2$ with $\operatorname{ri}(\operatorname{dom} f_1) \cap \operatorname{ri}(\operatorname{dom} f_2) \neq \emptyset$, it can be shown [17, Cor. 10.9] that $\partial f = \partial f_1 + \partial f_2$. Thus f can be minimized using operator splitting for the subgradients. As

$$x \in J_{\tau \partial f_i}(y) \Longleftrightarrow x = \operatorname{argmin} \frac{1}{2\tau}\|x - y\|_2^2 + f_i(x), \quad i \in \{1, 2\},$$

the computation of the resolvents reduces to proximal point optimization problems involving only the f_i. The Douglas-Rachford iteration takes the form

1. Choose arbitrary $u^0 \in \mathbb{R}^n$ and fix $\tau > 0$.
2. Solve $u^k = \operatorname{argmin}_u \left\{\frac{1}{2\tau}\|u - z^k\|_2^2 + f_1(u)\right\}$.
3. Solve $w^k = \operatorname{argmin}_w \left\{\frac{1}{2\tau}\|w - (2u^k - z^k)\|_2^2 + f_2(u)\right\}$.
4. Set $z^{k+1} \leftarrow z^k + w^k - u^k$, $k \leftarrow k+1$ and go to 2. until convergence in (z^k, u^k).

Under the assumptions that f_1 and f_2 are proper, convex, lsc; $\operatorname{ri}(\operatorname{dom} f_1) \cap \operatorname{ri}(\operatorname{dom} f_2) \neq \emptyset$; and that $\min_u f(u)$ has a solution, the Douglas-Rachford iteration converges and $u^k \to u \in \operatorname{argmin} f(u)$ [20, Prop. 3.40].

3.2 Application

For our specific problem, we split

$$\inf_{u \in C} f(u) = \inf_u (f_1(u) + f_2(u)) \ , \quad f_1(u) = -\langle u, s\rangle_\Omega + \lambda \operatorname{TV}(u) \ , \quad f_2(u) = \delta_C(u) \ . \tag{17}$$

and get the following Douglas-Rachford scheme:

Algorithm 1 Outer loop (Douglas-Rachford)

```
choose some $u^0$ and a fixed step size $\tau > 0$
repeat
    solve
        $u^k \leftarrow \mathrm{argmin}_u \Big\{ \frac{1}{2\tau} \|u - z^k\|^2 - \langle u, s \rangle + \sigma_{E_\lambda}(u) \Big\}$    (18)
    solve
        $w^k \leftarrow \mathrm{argmin}_w \Big\{ \frac{1}{2\tau} \|w - (2u^k - z^k)\|^2 + \delta_C(w) \Big\}$    (19)
    $z^{k+1} \leftarrow z^k + w^k - u^k$
until $\|u^k - u^{k-1}\|_\infty \leqslant \delta_{\mathrm{outer}}$.
```

From the remarks in section 3.1, we get convergence of the scheme for the discrete case: $\delta_C(w)$ and σ_{E_λ} are both proper, convex, lsc with $\mathrm{dom}\sigma_{E_\lambda} = \mathbb{R}^n$ and $\mathrm{rint}(C) \neq \emptyset$. Also, f is bounded from below on the compact set C and thus attains its minimum.

In practice, one has to deal with solutions of the subproblems with limited accuracy. While there are extensions of the convergence result that take these inexact solutions into account [20, Prop. 4.50], they require the subproblems to be solved with increasing accuracy. While not strictly theoretically justified, we found that in practice the method generally converged even though these requirements were not met.

We see that the second subproblem (19) is just a projection on the constraint set:

$$w^k = \Pi_C(2u^k - z^k) \, . \tag{20}$$

As C is the direct product of unit simplices, this can be solved by one projection on the low-dimensional unit simplex Δ_L per $x \in \Omega$. These projections can be computed in a finite number of steps [22].

The first subproblem (18) can be rewritten as

$$u^k = \mathrm{argmin}_u \frac{1}{2} \|u - (z^k + \tau s)\|^2 + (\tau\lambda) TV(u), \tag{21}$$

which is just the classical Rudin-Osher-Fatemi (ROF, TV-L^2) problem with the regularization parameter set to $\tau\lambda$, and extended to vector-valued u. There is a vast body of literature on the solution of the ROF problem. Among others, authors have suggested PDE, fixpoint or interior point methods for primal [2, 23], dual [24–26] or mixed [27] formulations.

Here we evaluate two approaches: First, we will formulate a particularly simple gradient projection method in the operator splitting framework. This scheme was introduced in [25] and extended to the multidimensional case in [15]. The second, faster approach is an application of the fast half-quadratic method presented by Yang et al. [14].

4 Inner Loop Optimization

4.1 Forward-backward approach for the inner problem

We start by rewriting the optimality condition of (18),

$$\begin{aligned} &\frac{1}{\tau}(z^k - u) + s \in \partial \sigma_{E_\lambda}(u) \\ &\qquad \Leftrightarrow \quad u \in N_{E_\lambda}\left(\left(z^k - u\right)/\tau + s\right) \\ &\qquad \Leftrightarrow \quad u = \tau\left(\left(z^k/\tau + s\right) - \Pi_{E_\lambda}\left(z^k/\tau + s\right)\right). \end{aligned}$$

To compute the projection Π_{E_λ}, we use the dual representation,

$$\Pi_{E_\lambda}(x) = \operatorname*{argmin}_{q \in E_\lambda} \frac{1}{2}\|q - x\|_\Omega^2 = \operatorname{Div}\left\{\operatorname*{argmin}_{p} \frac{1}{2}\|\operatorname{Div}p - x\|_\Omega^2 + \delta_{D_\lambda}(p)\right\}. \quad (22)$$

Using a simple forward-backward splitting for the inner problem results in the (gradient projection) update rule

$$p^{j+1} = \Pi_{D_\lambda}\left(p - \nu \operatorname{Div}^\top\left(\operatorname{Div}p - x\right)\right).$$

The projection Π_{D_λ} can be computed explicitly and is separable in x, while the inner part can be computed for all models independently. This opens up the method to parallelization.

Convergence is guaranteed for $\nu < 2/\|\operatorname{Div}^\top \operatorname{Div}\|$ (see e.g. [20, Thm. 3.12]). Extending the argument in [24, Thm. 3.1], we find that $\|\operatorname{div}\| \leqslant \sqrt{4d}$. Accordingly, we may set $\nu < \frac{1}{2d}$. In our experiments, we set $\nu = \frac{0.95}{2d}$ to avoid numerical problems close to the theoretical maximum. Wrapping up, we have

Algorithm 2 Inner loop, forward-backward approach

```
$x \leftarrow \frac{z^k}{\tau} + s$, choose arbitrary $p^0 \in \mathbb{R}^{n \times d \times L}$
repeat
    $p^{j+1} = \Pi_{D_\lambda}(p^j - \nu \operatorname{Div}^\top(\operatorname{Div}p - x))$
until $\|p^{j+1} - p^j\|_\infty \leqslant \delta_{\text{inner}}$
$u^k \leftarrow \tau(x - \operatorname{Div}p^{j+1})$.
```

4.2 Half-quadratic approach for the inner problem

While the forward-backward method is simple and easy to implement, its convergence speed is in practice not satisfactory. As an alternative, we tested a method by Yang et al. [14], which was proposed for general multichannel image restoration. In the following, we give a short overview specialized to the ROF case. Half-quadratic regularization has been introduced by [28]. For an overview of related techniques we refer to [29].

Starting from (21), the problem is to find

$$u^k = \operatorname{argmin}_u g(u)\,, \quad g(u) := \frac{\mu}{2}\|u - f\|^2 + TV(u)\,, \tag{23}$$

where $\mu := \frac{1}{\tau\lambda}$ and $f := z^k + \tau s$. To avoid the nondifferentiability of the discrete TV, the authors employ a smoothing approach for the discrete norms. For given $\beta > 0$, let

$$\begin{aligned} \mathrm{TV}_\beta(u) &:= \sum_{x\in\Omega} \phi_\beta(G_x u), \\ \phi_\beta(t) &:= \begin{cases} \frac{\beta}{2}\|t\|^2 + \frac{1}{2\beta}, & \text{if } \|t\| \leqslant \frac{1}{\beta}, \\ \|t\|, & \text{otherwise}\,. \end{cases} \end{aligned}$$

ϕ_β is the Huber function. For large β we do not lose much when solving (23): Following [30, 3.30], we have

$$0 \leqslant \mathrm{TV}_\beta(u) - \mathrm{TV}(u) \leqslant \frac{n}{2\beta}.$$

Define $g_\beta(u)$ as $g(u)$ with TV replaced by TV_β. For solutions u^*_β and u^* of the smoothed respective original problem, we get

$$g(u^*_\beta) \leqslant g_\beta(u^*_\beta) \leqslant g_\beta(u^*) \leqslant g(u^*) + \frac{n}{2\beta}.$$

Thus the solution u^*_β of the smoothed problem is $\frac{n}{2\beta}$ - suboptimal for the original problem. ε-suboptimality requires

$$\beta \geqslant \frac{n}{2\varepsilon}. \tag{24}$$

Using a half-quadratic approach, Yang et al. derive the splitting/penalty formulation

$$(u, y) = \operatorname*{argmin}_{y_x\in\mathbb{R}^{Ld}, x\in\Omega, u\in\mathbb{R}^{nL}} \sum_{x\in\Omega} \left(\|y_x\| + \frac{\beta}{2}\|y_x - G_x u\|^2\right) + \frac{\mu}{2}\|u - f\|^2_\Omega\,. \tag{25}$$

This can be solved using alternating minimization w.r.t. u and the auxiliary variables y_x. The latter is highly parallelizable, as it boils down to n separate explicit operations:

$$y_x^{j+1} = \max\left\{\|G_x u\| - \frac{1}{\beta}, 0\right\} \frac{G_x u}{\|G_x u\|}. \tag{26}$$

On the other hand, minimizing (25) for u amounts to solving the normal equations for the quadratic program:

$$\left(\mathrm{Grad}^\top \mathrm{Grad} + \frac{\mu}{\beta} I_{(nL)}\right) u^{j+1} = \mathrm{Grad}^\top y^{j+1} + \frac{\mu}{\beta} f, \tag{27}$$

where y^{j+1} is a proper rearrangement of the y_x.

For periodic boundary conditions, Yang et al. solved (27) rapidly using FFT. In our case, we have Neumann boundary conditions, so the Discrete Cosine Transform (DCT-2) is appropriate [31]. Specifically, define $c^i \in \mathbb{R}^{n_i}$ as

$$\begin{aligned}
(c^i)_k &:= 1 - \cos\left(\frac{3}{2}\frac{k\pi}{n_i}\right) / \cos\left(\frac{1}{2}\frac{k\pi}{n_i}\right), \quad i \in \{1, \ldots, d\}, \\
D_{\text{grad}} &:= \left(\sum_{i=1}^{d} I_{n_1} \otimes \ldots \otimes I_{n_{i-1}} \otimes \operatorname{diag}(c^i) \otimes I_{n_{i+1}} \otimes \ldots \otimes I_{n_d}\right), \\
D_{\text{Grad}} &:= I_L \otimes D_{\text{grad}}.
\end{aligned}$$

Note that D_{grad} and D_{Grad} are both diagonal. Then

$$\begin{aligned}
\text{grad}^\top \text{grad} &= (\text{DCT}^{-1}) D_{\text{grad}} (\text{DCT}), \\
\text{Grad}^\top \text{Grad} &= (I_L \otimes \text{DCT}^{-1}) D_{\text{Grad}} (I_L \otimes \text{DCT}),
\end{aligned}$$

where DCT is the DCT transformation matrix for $(n_1, \ldots, n_d)$-dimensional data. To solve the linear equation system (27), we compute

$$\begin{aligned}
u^{j+1} &= \left(\text{Grad}^\top \text{Grad} + \frac{\mu}{\beta} I_{(nL)}\right)^{-1} \left(\text{Grad}^\top y^{j+1} + \frac{\mu}{\beta} f\right) \qquad (28) \\
&= \left(I_L \otimes \left(\text{DCT}^{-1} \left(D_{\text{grad}} + \frac{\mu}{\beta} I_n\right)^{-1} \text{DCT}\right)\right) \left(\left(I_L \otimes \text{grad}^\top\right) y^{j+1} + \frac{\mu}{\beta} f\right)
\end{aligned}$$

This amounts to $2L$ independent (parallelizable) individual DCTs which can be efficiently computed in $O(n \log n)$ each.

By the alternating application of (26) and (28), we can solve (25) for fixed β large enough to guarantee the required suboptimality. In practice, convergence can be sped up by starting with a smaller β and solving a sequence of problems for increasing β, where each problem is warm-started with the solution for the previous problem. The complete algorithm for $\beta > 0$ and arbitrary $u^0 \in \mathbb{R}^{nL}$ reads:

Algorithm 3 Inner loop, half-quadratic approach

```
while stopping criterium not satisfied do
    compute $y^{j+1}$ from (26)
    compute $u^{j+1}$ from $y^{j+1}$ and (28),
    possibly increase $\beta$
end while
```

The stopping criteria can be based on the residual [14]. For our experiments, we set a fixed iteration count, as increasing β at each step turned out to lead to fastest convergence, and residua for different β are not comparable.

$\tau\lambda$	0.1	1	2	5	10	20	50
t_{HQ}	1.14	1.23	1.20	1.31	0.98	0.95	1.08
t_{FB}	1.03	1.02	1.06	1.03	1.22	1.25	1.19
r_{HQ}	3901.9	27660.7	36778.5	40038.8	42262.8	44377.1	44752.5
r_{FB}	3901.9	27660.4	36760.6	40104.3	42924.3	46988.6	57504.9
rel. diff.	1.17e-16	1.24e-5	4.85e-4	-1.64e-3	-0.0156	-0.0588	-0.285

Table 1. Run times t (in seconds), objective function values r and relative differences $(r_{\mathrm{HQ}} - r_{\mathrm{FB}})/r_{\mathrm{HQ}}$ for the experiment in Fig. 3. For larger $\tau\lambda$, the half-quadratic method gives more accurate results in the same time.

5 Experiments, Performance Evaluation

5.1 Inner Problem

We compared Yang's half-quadratic approach to the conventional forward-backward method. The difficulty with the former lies in the choice of the update strategy for β. We chose a generalization of the exponential strategy as outlined in the original paper: set $\beta = \beta_{\min}$ and update by multiplying with $c := (\beta_{\max}/\beta_{\min})^{1/K}$ for some K until $\beta = \beta_{\max}$.

We made the following observations:

- In order to rapidly minimize the objective function, it is best to use a continuation strategy, i.e. to increase β at each step, rather than spending time on solving (25) exactly for each β.
- Increasing K generally improves the quality of the result.
- For fixed $\beta_{\max}$ and K, there seems to be a unique optimal $\beta_{\min}$ that minimizes the final objective function value.

With the fixed continuation strategy and fixed $\beta_{\max}$, we found the optimal $\beta_{\min}$ to usually lie in the range of $10^{-5}\beta_{\max}$ to $10^{-3}\beta_{\max}$. Unfortunately, there seems to be a strong dependency on the choice of λ as well as the scale and complexity of s. We set $\beta_{\min} = 0.2 \cdot 10^{-4}\beta_{\max}$, which worked well for our data. $\beta_{\max}$ was set at $n/0.2$ according to (24) with $\varepsilon = 0.1$.

To evaluate the performance of the two methods, we chose fixed iteration counts so that both had approximately the same runtime, and compared the results in terms of the objective function value (Fig. 3, Table 1). The algorithms were implemented and optimized in MATLAB. For small $\tau\lambda$, the forward-backward method gives slightly better results, while for larger $\tau\lambda$ (> 3 for the image shown), the half-quadratic method stays ahead. For $\tau\lambda = 20$, less than 10 iterations are required to reach the quality of the forward-backward method with 300 iterations, giving a speedup of about $4 - 5$. However, finding the optimal parameter set is more involved than for the forward-backward method.

5.2 Overall Problem

We evaluated the performance of our algorithm against five different methods in their publicly available implementations from the Middlebury MRF bench-

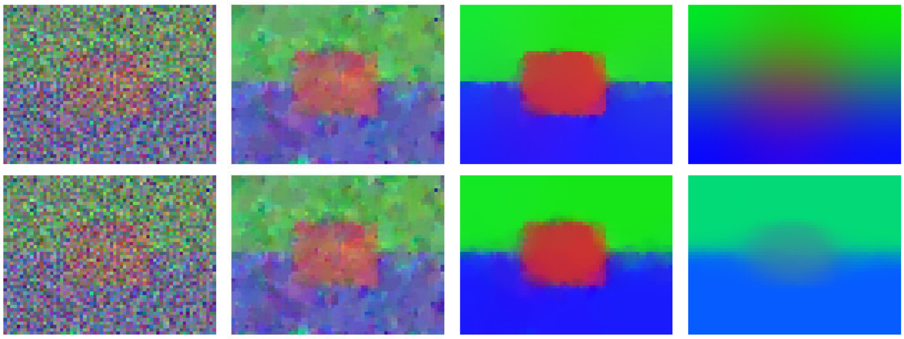

Fig. 3. Results of the speed comparison between half-quadratic method and forward-backward method for the inner problem, applied to data from the *first iteration* of the outer problem (cf. Table 1). **Top row:** Half-quadratic method. **Bottom row:** Forward-backward method. **Left to right:** Original input, $\tau\lambda = 2, 5, 20$. Iteration counts were fixed at 80 resp. 300 to equalize the runtime for both approaches. For larger regularization parameter, the half-quadratic method outperforms the forward-backward approach as smoothness increases.

mark [32]: Belief Propagation (BP), Sequential Belief Propagation (BPS), Graph Cuts with alpha-expansion (GCE), Graph Cuts with alpha-beta swap (GCS) and Sequential Tree Reweighted Belief Propagation (TRBPS). Each of the grayscale 32×32 images with pixel values in $[0, 1]$ was overlaid with normally distributed noise, and then segmented into four gray levels with fixed intensities with the distance measure from section 2.1. The experiments were repeated 20 times for each λ, with fixed step size $\tau = 1$. In view of the last section and in order not to mix up speed with accuracy issues, we used the forward-backward approach for the inner loop. Termination criteria were set at $\delta_{\text{inner}} = 1e-3, \delta_{\text{outer}} = 2e-2$.

For small λ, our method shows results comparable to the other approaches with respect to the number of bad labels. We point out again that this solution to the non-binary labeling problem is achieved by solving the *convex* optimization problem (9) followed by local rounding as explained in section (2.3).

In contrast to our method, the MRF benchmark algorithms optimize the *anisotropic* energy. To compensate, their λ was scaled by a common factor of $\approx \sqrt{2}$ that was found empirically. Nevertheless, their discretization gives them a small advantage on images with axis parallel edges (experiments 1 and 2). It also explains why in experiments 3 and 4, our method could perform well w.r.t. the number of bad labels, while the energy was quite high.

Fig. 6 demonstrates the performance of our algorithm for color segmentation. Generally only few outer iterations (20 in our case) are necessary for accurate optimization.

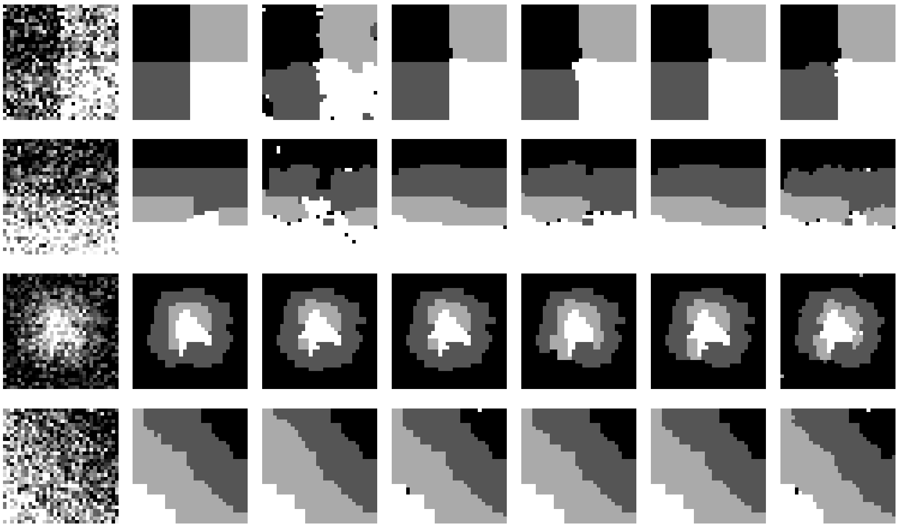

Fig. 4. Exemplary grayscale segmentation results for the benchmarked methods for four labels. **Left to right:** Noisy input data, final results for BP, BPS, GCE, GCS, TRWS, and the proposed method (TV). λ was manually chosen for each method. Axis-parallel edges are better recovered by the anisotropic methods, while our isotropic discretization has an advantage on diagonal edges.

6 Conclusion and Further Work

In this paper, we presented a convex variational approach to solve the combinatorial multi-labeling problem for energies involving a general data term, total-variation-like regularizers, and simplex constraints. To enforce the required simplex constraint, we based our approach on the globally convergent Douglas-Rachford operator splitting scheme. We evaluated two methods in order to efficiently solve the ROF-type subproblems, and showed that Yang's half-quadratic approach allows faster convergence at the price of more involved parameter tuning.

Experiments showed that the quality of the generated labelings is comparable to state of the art discrete optimization methods, and can be achieved by just solving a convex optimization problem.

Due to the generality of the data term, our method allows for a wide range of features or distance measures. To fully evaluate these possibilities in connection with variations of the TV measure is a subject of our future research.

Acknowledgement Jing Yuan gratefully acknowledges support by the German National Science Foundation (DFG) under grant SCHN 457/9-1.

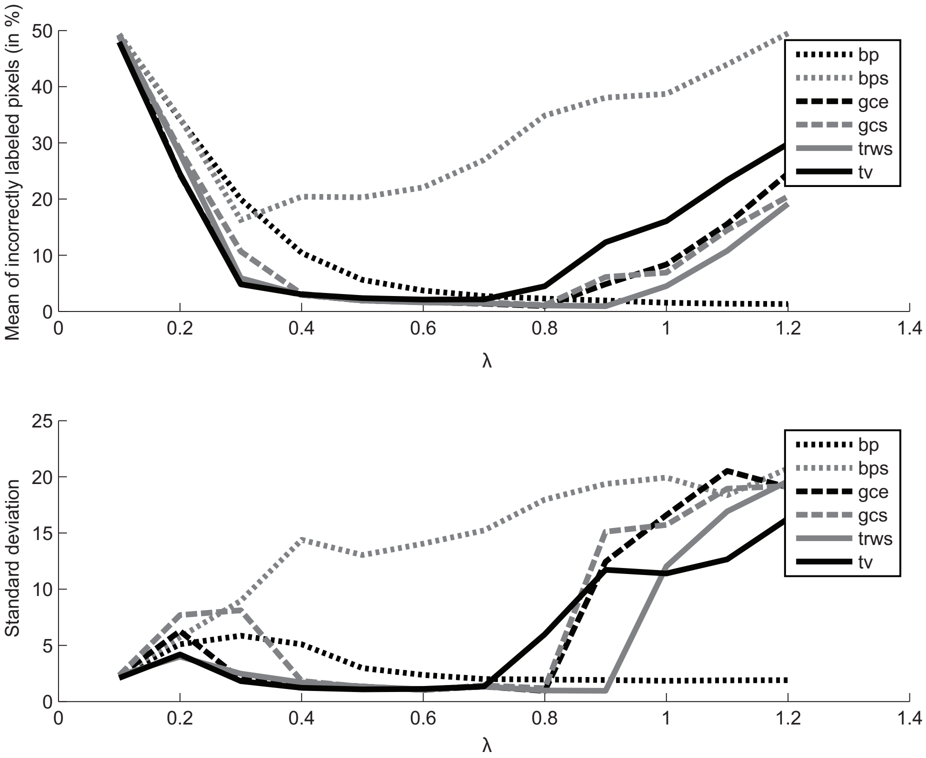

Fig. 5. Error rates compared to ground truth for the first experiment in Fig. 4 for varying λ. For each λ, all experiments were repeated 20 times with random noise (zero-mean Gaussian with $\sigma = 0.45, 0.35, 0.25$ resp. 0.35 for experiments 1–4 and image intensities in $[0, 1]$), and the percentage of incorrectly assigned labels compared to ground truth was recorded. Sequential Belief Propagation (BPS) generally performed worst, while our method (TV) was on par with the others, in particular for lower λ. The figure also reveals that belief propagation (BP) gets stuck in a good, but often inferior local optimum. As a consequence, the method does not respond to larger values of the regularization parameter λ, i.e. stronger regularizaton requested by the user.

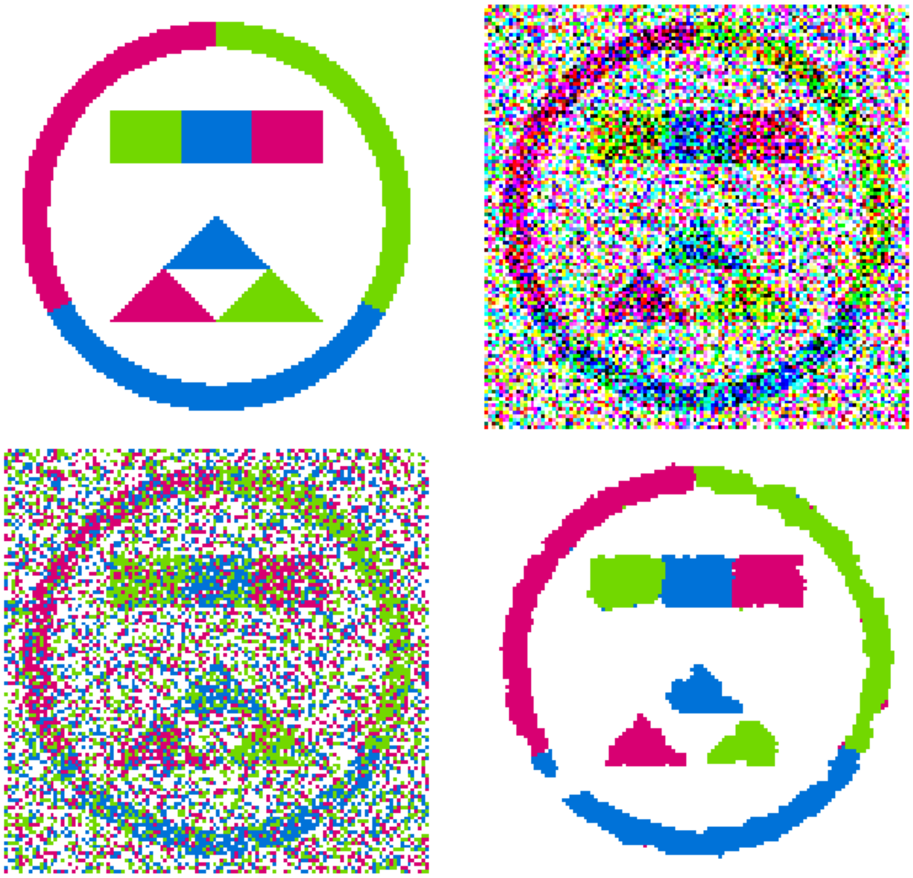

Fig. 6. Performance of our method for four-class segmentation based on ℓ_1 color distance. **Top row:** Ground truth, inspired by [27, 33] (left), overlaid with Gaussian noise, $\sigma = 1$ (right). **Bottom row:** Local nearest-neighbor labeling (left), our approach with $\lambda = 0.7$ after 20 outer iterations (right). The energy of the result is about 1% lower than the energy of the ground truth, suggesting that at this noise level, further improvements are limited by the model.

References

1. Boykov, Y., Veksler, O., Zabih, R.: Fast approximate energy minimization via graph cuts. PAMI **23**(11) (2001) 1222–1239
2. Rudin, L., Osher, S., Fatemi, E.: Nonlinear total variation based noise removal algorithms. Physica D **60** (1992) 259–268
3. Boykov, Y., Kolmogorov, V.: An experimental comparison of min-cut/max-flow algorithms for energy minimization in vision. PAMI **26**(9) (September 2004) 1124–1137
4. Kolmogorov, V., Zabih, R.: What energy functions can be minimized via graph cuts? PAMI **26**(2) (February 2004) 147–159
5. Strang, G.: Maximal flow through a domain. Mathematical Programming **26** (1983) 123–143
6. Chan, T.F., Esedoḡlu, S., Nikolova, M.: Algorithms for finding global minimizers of image segmentation and denoising models. J. Appl. Math. **66**(5) (2006) 1632–1648
7. Pock, T., Schönemann, T., Graber, G., Bischof, H., Cremers, D.: A convex formulation of continuous multi-label problems. In: Proceedings of the ECCV. Volume 3. (October 2008) 792–805
8. Ishikawa, H.: Exact optimization for markov random fields with convex priors. PAMI **25**(10) (October 2003) 1333–1336
9. Kleinberg, J., Tardos, E.: Approximation algorithms for classification problems with pairwise relationships: Metric labeling and markov random fields. In: FOCS. (1999) 14–23
10. Ziemer, W.: Weakly Differentiable Functions. Springer (1989)
11. Meyer, Y.: Oscillating Patterns in Image Processing and Nonlinear Evolution Equations. Volume 22 of Univ. Lect. Series. AMS (2001)
12. Sapiro, G., Ringach, D.L.: Anisotropic diffusion of multi-valued images with applications to color filterin. In: Trans. Image Process. Volume 5. (1996) 1582–1586
13. Chan, T.F., Shen, J.: Image processing and analysis. SIAM (2005)
14. Yang, J., Yin, W., Zhang, Y., Wang, Y.: A fast algorithm for edge-preserving variational multichannel image restoration. Technical Report TR08-09, Rice University (July 2008)
15. Duval, V., Aujol, J.F., Vese, L.: A projected gradient algorithm for color image decomposition. CMLA Preprint (2008-21) (June 2008)
16. Chan, T., Esedoglu, S., Park, F., Yip, A.: Total variation image restoration: Overview and recent developments. In: The Handbook of Mathematical Models in Computer Vision. Springer (2005)
17. Rockafellar, R., Wets, R.J.B.: Variational Analysis. 2nd edn. Springer (2004)
18. Douglas, J., Rachford, H.H.: On the numerical solution of heat conduction problems in two and three space variables. Transactions of the AMS **82**(2) (July 1956) 421–439
19. Lions, P.L., Mercier, B.: Splitting algorithms for the sum of two nonlinear operators. SIAM Journal on Numerical Analysis **16**(6) (December 1979) 964–979
20. Eckstein, J.: Splitting Methods for Monotone Operators with Application to Parallel Optimization. PhD thesis, MIT (June 1989)
21. Eckstein, J., Bertsekas, D.P.: On the douglas-rachford splitting method and the proximal point algorithm for maximal monotone operators. Mathematical Programming **55** (1992) 293–318
22. Michelot, C.: A finite algorithm for finding the projection of a point onto the canonical simplex of $\mathbb{R}^n$. Journal of Optimization Theory and Applications **50**(1) (July 1986) 195–200

23. Dobson, D.C., Curtis, Vogel, R.: Iterative methods for total variation denoising. J. Sci. Comput **17** (1996) 227–238
24. Chambolle, A.: An algorithm for total variation minimization and applications. JMIV **20** (2004) 89–97
25. Chambolle, A.: Total variation minimization and a class of binary MRF models. In: EMMCVPR. Volume 3757. (2005) 136–152
26. Aujol, J.F.: Some algorithms for total variation based image restoration. CMLA Preprint (2008-05) (March 2008)
27. Chan, T.F., Golub, G.H., Mulet, P.: A nonlinear primal-dual method for total variation-based image restoration. J. Sci. Comput **20** (1999) 1964–1977
28. Geman, D., Yang, C.: Nonlinear image recovery with halfquadratic regularization. IEEE Trans. Image Proc. **4**(7) (1995) 932–946
29. Cohen, L.: Auxiliary variables and two-step iterative algorithms in computer vision problems. JMIV **6**(1) (1996) 59–83
30. Weiss, P., Aubert, G., Blanc-Fraud, L.: Efficient schemes for total variation minimization under constraints in image processing. Technical Report 6260, INRIA (July 2007)
31. Strang, G.: The discrete cosine transform. SIAM Review **41**(1) (1999) 135–147
32. Szeliski, R., Zabih, R., Scharstein, D., Veksler, O., Kolmogorov, V., Agarwala, A., Tappen, M., Rother, C.: A comparative study of energy minimization methods for markov random fields. In: ECCV. Volume 2. (May 2006) 19–26
33. Hintermüller, M., Stadler, G.: An infeasible primal-dual algorithm for total bounded variation-based inf-convolution-type image restoration. J. Sci. Comput. **28**(1) (2006) 1–23