Computing 74, 353–376 (2005)
Digital Object Identifier (DOI) 10.1007/s00607-004-0097-8

Computing
Printed in Austria

Denoising of Smooth Images Using L^1-Fitting

T. Kärkkäinen, Jyväskylä, **K. Kunisch,** Graz, and **K. Majava,** Jyväskylä

Received November 25, 2003; revised June 30, 2004
Published online: November 15, 2004

Abstract

In this paper, denoising of smooth (H_0^1-regular) images is considered. The purpose of the paper is basically twofold. First, to compare the denoising methods based on L^1- and L^2-fitting. Second, to analyze and realize an active-set method for solving the non-smooth optimization problem arising from the former approach. More precisely, we formulate the algorithm, proof its convergence, and give an efficient numerical realization. Several numerical experiments are presented, where the convergence of the proposed active-set algorithm is studied and the denoising properties of the methods based on L^1- and L^2-fitting are compared. Also a heuristic method for determining the regularization parameter is presented and tested.

AMS Subject Classifications: 68U10, 90C30.

Keywords: Denoising, fitting techniques, non-smooth fitting, active-set methods.

1. Introduction

The basic model in image denoising is that a noisy image, denoted by z, results from a degradation of the form

$$z = z^* + \eta, \tag{1}$$

where z^* is the true image and $\eta \in \mathcal{N}(0, \sigma)$ represents normally distributed random noise [3,8]. In accordance with (1), the usual approach to restore z^* from z is based on minimizing a regularizing term with noise constraints [26]:

$$\min_u R(u), \tag{2}$$

$$\int_\Omega z - u dx = \int_\Omega \eta dx = 0, \quad \int_\Omega |u - z|^2 dx = \sigma^2. \tag{3}$$

The formulation is based on assumption that the noise η is Gaussian (normally distributed) with zero mean. The second constraint corresponds to the assumption that standard deviation of the noise is σ. In real applications, however, the standard deviation of noise is usually not known. In that case, the denoising problem can be considered in the unconstrained form as

$$\min_u \int_\Omega |u-z|^2 dx + \beta R(u), \tag{4}$$

where β denotes a regularization parameter, the value of which actually corresponds to the reciprocal of the Lagrange multiplier for the second constraint in (3) [2]. In the case that σ is not known, formulation (4) allows us to use different techniques for determining the value of the regularization parameter. In this paper, a heuristic method for this purpose is presented and tested.

The approach in (4) relies on strong (intrinsic) assumptions on the distribution of η: there are no heavy tails and the distribution is symmetric. If either of these assumptions fails, then the use of L^2-fitting is not an optimal choice. On the other hand, for the so-called robust formulation based on L^1-fitting $\int_\Omega |u-z|dx$, it has been shown that the corresponding statistics can tolerate up to 50 percent false observations and other inconsistencies [22]. Hence, this technique relaxes the underlying requirements for the L^2-case. Especially, the L^1-formulation suits significantly better for denoising data containing so-called outliers, i.e., observations containing large measurement errors [4]. For this purpose, the L^2-formulation needs some extra preprocessing stage utilizing robust procedures to locate the outliers [25].

The discussion and a numerical comparison of the robustness between the discrete l_1 and l_2 norms are given in [4], [24]. Moreover, in [11], the connection between robust statistics and different fitting techniques was discussed. In [21], the L^1-fitting was applied for processing outliers. It was shown that if a small value of the regularization parameter is chosen, then we are able to detect the outliers, while the remaining data stay essentially untouched. On the other hand, if the aim is to smooth outliers, then due to a larger value of the regularization parameter the data will necessarily be smoothed also beyond the positions of the outliers.

In this paper, we consider image denoising problems with smooth regularization. These formulations are intended for removing noise from smooth images, i.e., images not containing discontinuities (jumps, edges). For images containing sharp edges, one needs to apply a nonsmooth regularization, e.g., BV seminorm [2], [26]. In [17], L^1-fitting is combined with BV regularization.

The original image can, for instance, be obtained as a solution of some second order PDE. Especially, preprocessing for the equation error identification method as described in [6], [10] is one concrete application of the methodology developed here. Further examples of data containing outliers is given by electroencephalographic (EEG) recordings of the electrical activity of the brain, which may contain instrumental or biological artifacts (e.g., [20]). Another interesting example involves slow-combustion (smouldering) fronts propagating in burning sheets of paper [19]. The process can be modelled using the Kardar-Parisi-Zhang (KPZ) equation $\partial_t h = c + \nu\Delta h + \frac{\lambda}{2}(\nabla h)^2 + \eta$, where $h \equiv h(x,t)$ is the height of the interface, c its zero-slope velocity, and η denotes the effective noise [9]. In order to characterize the burning material using the model parameters c, ν, and λ, the propagation of the emerging one-dimensional front was recorded with a CCD-camera system. Outliers in the resulting image were caused by, e.g., ash particles detached from the front.

The basic difficulty concerning the image denoising problem with L^1-fitting is its nondifferentiability in the classical sense. This means that common gradient-based solution methods, such as the conjugate gradient method, can not be applied for solving this problem. For this purpose, we analyze and realize an active-set method. This work continues our investigations on active-set methods based on the augmented Lagrangian regularization for nonsmooth optimization problems that have previously been conducted in [7], [12], [14], [15].

The contents of the paper are as follows: In Sect. 2, we introduce the actual optimization problems considered, the corresponding optimality conditions, and the used notations. In Sect. 3, a precise description of the active-set algorithm for solving the L^1-problem is given and its convergence is studied. Finally, in Sect. 4, numerical experiments with the proposed methods are presented. In the numerical experiments, the convergence of the proposed active-set algorithm is studied and the denoising properties of the methods based on L^1- and L^2-fitting are compared. Also a heuristic method for determining the regularization parameter is presented and tested. Finally, an example of denoising lens-paper burn data is shortly considered.

2. Notations and Basic Formulations

2.1. Continuous Problems

We consider the image denoising problem with L^1-fitting and smooth regularization

$$\min_{u \in H_0^1(\Omega)} \int_\Omega \left[\frac{\beta}{2} |\nabla u|^2 + |u - z| \right] dx \tag{5}$$

for the given function $z \in L^1(\Omega)$ and domain $\Omega \subset \mathbb{R}^d$. In this paper, we only consider $d = 2$, although the proposed techniques are not restricted to this particular case. Problem (5) has a unique solution $u^* \in H_0^1(\Omega)$ for every $\beta > 0$. Note that the cost functional in (5) is nondifferentiable in the classical sense. In order to overcome this difficulty, we replace the original cost functional with its smoother counterpart, the construction of which is based on the use of Lagrange smoothing techniques for convex, nonsmooth optimization problems [1], [5].

We see that problem (5) is equivalent to

$$\min_{u \in H_0^1(\Omega)} \max_{\lambda \in C} \int_\Omega \left[\frac{\beta}{2} |\nabla u|^2 + \lambda(u - z) \right] dx = \min_{u \in H_0^1(\Omega)} \max_{\lambda \in C} \; l(u, \lambda), \tag{6}$$

where $l : H_0^1(\Omega) \times L^2(\Omega) \to \mathbb{R}$ is the Lagrange-functional and the set C is defined as

$$C := \left\{\lambda \in L^2(\Omega) : |\lambda(x)| \leq 1 \text{ a.e. } x \in \Omega\right\}.$$

The equivalence of the two formulations results from the following identity

$$\max_{\lambda \in C} \int_\Omega \lambda(u - z)dx = \int_\Omega |u - z|dx.$$

Hence, λ represents the Lagrange smoothing of the subdifferential $\operatorname{sign}(u - z)$ [5].

The regularized optimality conditions for the solution u^* of (6) read as

$$\begin{cases} -\beta\Delta u^* + \lambda^* = 0, \quad \text{in } \Omega, \\ \lambda^*(x) = \frac{\lambda^*(x)+c(u^*-z)(x)}{\max\{1,|\lambda^*(x)+c(u^*-z)(x)|\}}, \quad \text{a.e. } x \in \Omega, \text{ for each } c > 0. \end{cases} \tag{7}$$

The second condition realizing the regularized complementarity condition is equivalent to the actual definition of the subdifferential

$$\begin{aligned} \lambda^*(x) &= \frac{(u^* - z)(x)}{|(u^* - z)(x)|}, \quad && \text{in } I^*(x) = \{x \in \Omega | (u^* - z)(x) \neq 0\}, \\ |\lambda^*(x)| &\leq 1, && \text{in } J^*(x) = \{x \in \Omega | (u^* - z)(x) = 0\}. \end{aligned} \tag{8}$$

Here, we notice that the non-smooth components of the original cost functional belong to the second subdomain $J^*(x)$ of Ω.

The corresponding image denoising problem with quadratic L^2-fitting reads as

$$\min_{u \in H_0^1(\Omega)} \int_\Omega \left[\frac{\beta}{2}|\nabla u|^2 + |u - z|^2\right] dx \tag{9}$$

with the linear optimality condition

$$-\beta\Delta\tilde{u}^* + 2(\tilde{u}^* - z) = 0 \tag{10}$$

for the unique solution $\tilde{u}^*$. By comparing (10) to (7) and (8), one notices that for the L^2-formulation the distance between $\tilde{u}^*$ and z plays an important role in the optimality condition (10), whereas only the (regularized) sign-function for $u^* - z$ appears in (7). This illustrates the relative insensitivity of the L^1-formulation towards false observations, compared to the L^2-formulation.

2.2. Discretization

In what follows, we denote by Mat^n the general class of square matrices in $\mathbb{R}^{n,n}$ and let Diag^n be the subclass of Mat^n consisting of diagonal matrices. Moreover, we let $N = \{1, 2, \ldots, n\}$ denote the ordered set of indices. The usual Euclidean inner product is denoted by $\langle \cdot, \cdot \rangle$ and the corresponding norm by $\| \cdot \| = \sqrt{\langle \cdot, \cdot \rangle}$. For a given vector $v \in \mathbb{R}^n$ and matrix $M \in \mathrm{Mat}^n$, we use the notation $|v|_{1,M}$ for $\sum_{i=1}^{N} |Mv|_i$.

In practise, with real digital images, it is reasonable to assume that we observe the function z in some points (pixels), which define a rectangular domain $\Omega = (x_0, x_1) \times (y_0, y_1) \subset \mathbb{R}^2$. Hence, we assume further that the domain Ω is divided into a rectangularly triangulated grid, where the nodal points coincide with the original observation points. Let $\{\varphi_i\}$ denote the piecewise linear finite element basis on this grid. Furthermore, let $K \in \mathrm{Mat}^n$ be the usual finite element stiffness matrix $K_{ij} = \int_\Omega \nabla\varphi_i \cdot \nabla\varphi_j dx$ and $M \in \mathrm{Diag}^n$ the diagonal, so–called lumped counterpart of the normal mass matrix $\int_\Omega \varphi_i \varphi_j dx$, which is obtained by using a trapezoidal numerical integration rule on the given mesh.

The discrete counterpart of the continuous formulation is obtained by replacing u with $\sum_{i=1}^{n} u_i \varphi_i$ and λ with $\sum_{i=1}^{n} \lambda_i \varphi_i$. Therefore, from now on, we denote by $u \in \mathbb{R}^n$ and $\lambda \in \mathbb{R}^n$ the vectors for the unknown coefficients (coinciding with the nodal values) of the above expansions and by $z \in \mathbb{R}^n$ the nodal values of function z.

By inserting the discretized functions into (5), the cost functional can be written in the form

$$\mathcal{J}(u) = \frac{\beta}{2} u^T K u + |u - z|_{1,M}. \tag{11}$$

Moreover, the discrete Lagrange-functional associated with (6) reads as

$$\mathcal{L}(u, \lambda) = \frac{\beta}{2} u^T K u + \lambda^T M (u - z). \tag{12}$$

The regularized optimality conditions for (12) are given by

$$\begin{cases} \beta K u^* + M \lambda^* = 0, \\ \lambda_i^* = \frac{\lambda_i^* + c(u^* - z)_i}{\max\{1, |\lambda_i^* + c(u^* - z)_i|\}}, \quad \text{for all } i \in N \ \text{ and } c > 0. \end{cases} \tag{13}$$

As in the continuous case, the second condition in (13) is equivalent to

$$\begin{aligned} \lambda_i^* &= \frac{(u^* - z)_i}{|(u^* - z)_i|}, && \text{in } I^* = \left\{ i \in N | (u^* - z)_i \neq 0 \right\}, \\ |\lambda_j^*| &\leq 1, && \text{in } J^* = \left\{ j \in N | (u^* - z)_j = 0 \right\}. \end{aligned} \tag{14}$$

Finally, we remind that the Lagrange multiplier λ^* is unique, because it can be uniquely solved from the first equation in (13) with the unique u^*.

The discrete cost functional corresponding to the L^2-formulation (9) is given as

$$\bar{\mathscr{J}}(u) = \frac{\beta}{2} u^T K u + (u - z)^T M (u - z)$$

with the optimality condition

$$\beta K \tilde{u}^* + 2M(\tilde{u}^* - z) = 0.$$

For later use, we define

$$P_S v = \begin{cases} v_i, & \text{for} \quad i \in S, \\ 0, & \text{for} \quad i \notin S, \end{cases}$$

to be the projection onto the given index subset $S \subset N$. The corresponding restriction, containing *only* the values on S is denoted by R_S. In addition, we denote for a vector $v \in \mathbb{R}^n$, matrix $A \in \mathrm{Mat}^n$, and index subsets $S \subset N$ and $T \subset N$, $v_S = P_S v$, $A_S = P_S A$, and $A_{ST} = P_S A P_T$.

3. Numerical Realization: Algorithms and Convergence

Next, we propose and analyze an active-set method for minimizing the discretized, nonsmooth cost functional (11). The method is based on an outer iteration, where an active set representing the current knowledge of the nonsmooth components is determined. Then, during an inner iteration, a constrained, reduced optimization problem is solved with sufficient precision [7], [12], [14], [15].

Algorithm 1: Basic Active-Set Method

Step 0^o Initialize $(u^0, \lambda^0) \in \mathbb{R}^n \times \mathbb{R}^n$. Choose $c > 0$ and set $k = 0$.

Step 1^o Determine

$$\begin{aligned} J = J^k &= \{j \in N : |\lambda_j^k + c(u^k - z)_j| \leq 1\} \text{ (active)}, \\ I = I^k &= N \setminus J^k \text{ (inactive)}. \end{aligned} \tag{15}$$

If $k > 1$ and $J^k = J^{k-1}$, then STOP; the solution is u^k.

Step 2^o Let (u^{k+1}, λ^{k+1}) be the solution of

$$\min_u \mathscr{J}_I(u) = \frac{\beta}{2} u^T K u + |u - z|_{1,M_I} \quad \text{subj. to} \quad u_j = z_j, \text{ for all } j \in J^k, \tag{16}$$

as described in Lemma 3.1.

Step 3^o Set $k = k + 1$ and go to Step 1^o.

Let us first consider the solvability of Step 2^o. We recall that actually we have $\mathscr{J}_I(u) = \mathscr{J}(u)$ for all u satisfying the constraint $P_J(u - z) = 0$.

Lemma 3.1: *There exists a unique solution* $(\bar{u}, \bar{\lambda})$ *for problem (16), which satisfies the optimality conditions*

$$\beta K\bar{u} + M\bar{\lambda} = 0, \quad \begin{cases} |\bar{\lambda}_i| \leq 1 \text{ and } \bar{\lambda}_i(\bar{u} - z)_i = |(\bar{u} - z)_i|, \text{ for all } i \in I, \\ \bar{u}_j = z_j, \text{for all } j \in J. \end{cases} \tag{17}$$

Proof: Unique solvability of (16) is clear. The basic optimality condition for the solution $\bar{u}$ reads as

$$\langle \beta K\bar{u}, v - \bar{u} \rangle + |v - z|_{1,M_I} - |\bar{u} - z|_{1,M_I} \geq 0, \quad \text{for all } P_J(v - z) = 0. \tag{18}$$

This can be expressed as an equality by introducing Lagrange multipliers for the nonsmooth part of the cost functional and for the constraint. Because the two index subsets I and J are disjoint, these multipliers together form one vector for the whole Lagrangian variable λ. Then, the corresponding Lagrangian for (16) is of the form

$$\mathscr{L}_I(u, \lambda) = \frac{\beta}{2} u^T K u + \lambda^T M_I(u - z) + \lambda^T P_J(u - z), \tag{19}$$

where λ on I coincides with the (regularized) subdifferential and on J with the multiplier for the equality constraint. The saddle-point conditions for the solution of (19) are given by

$$\begin{cases} \beta K\bar{u} + MP_I\bar{\lambda} + P_J\tilde{\lambda} = 0, \\ |\bar{\lambda}_i| \leq 1 \text{ and } \bar{\lambda}_i(\bar{u} - z)_i = |(\bar{u} - z)_i|, \quad \text{for all } i \in I, \\ P_J(\bar{u} - z) = 0 \quad \text{on } J. \end{cases} \tag{20}$$

By choosing $\bar{\lambda}_j = M_j^{-1}\tilde{\lambda}_j$, for all $j \in J$, we have the optimality condition in the required form. Finally, $\bar{\lambda}$ is unique, because it can be uniquely solved from $\beta K\bar{u} + M\bar{\lambda} = 0$.

3.1. Convergence Analysis

Next, we prove the convergence of the basic algorithm using a technique similar to the one in [7].

The first step towards a convergence analysis of any active-set method is to study the behavior of the two index sets J^k and I^k of the algorithm [12], [14], [15]. For this purpose, we introduce the following disjoint decomposition:

$$J^k = \{j \in N : |\lambda_j^k + c(u^k - z)_j| \leq 1\} = J_1^k \cup J_2^k,$$

where

$$J_1^k = \{j \in J^k : |\lambda_j^{k+1}| \leq 1\}, \quad J_2^k = \{j \in J^k : |\lambda_j^{k+1}| > 1\}.$$

Similarly,

$$I^k = \{i \in N : |\lambda_i^k + c(u^k - z)_i| > 1\} = I_1^k \cup I_2^k,$$

where

$$I_1^k = \{i \in I^k : (u^{k+1} - z)_i \neq 0\}, \quad I_2^k = \{i \in I^k : (u^{k+1} - z)_i = 0\}.$$

It then follows directly from the definition of Step 1^o that we have, for all $c > 0$, the next sets as

$$J^{k+1} = J_1^k \cup I_2^k, \quad I^{k+1} = I_1^k \cup J_2^k. \tag{21}$$

From (21) and Lemma 3.1, one can immediately deduce the following proposition.

Proposition 3.1: *Let* (u^{k+1}, λ^{k+1}), for $k \geq 1$, *be the solution obtained at step* 2^o *of Algorithm 1.*

(i) If $J^{k+1} = J^k$, then the pair (u^{k+1}, λ^{k+1}) *satisfies the optimality conditions (13), (14). Especially,* $u^{k+1} = u^*$, *i.e., the unique minimizer of (11).*

(ii) $u_j^k = z_j$ *for all* $j \in J^k$, *i.e.,* $P_J(u^k - z) = 0$ *for* $J = J^k$.

For the convergence proof, we also need the following lemma.

Lemma 3.2: *If* $u^{k+1} \neq u^k$, *then* $\mathscr{J}(u^{k+1}) < \mathscr{J}(u^k)$, *for all* $k \geq 1$.

Proof: For convenience, we use along this proof the notations $J = J^k$ and $I = I^k$. The optimality condition (18) in the proof of Lemma 3.1 and the fact that also $P_J(u^k - z) = 0$ give

$$\langle \beta K u^{k+1}, u^k - u^{k+1} \rangle + |u^k - z|_{1,M_I} - |u^{k+1} - z|_{1,M_I} \geq 0. \tag{22}$$

Moreover, since both u^k and u^{k+1} coincide with z on J, we have

$$\begin{aligned}
\mathscr{J}(u^k) - \mathscr{J}(u^{k+1}) &= \frac{\beta}{2} u^{k,T} K u^k + |u^k - z|_{1,M} - \frac{\beta}{2} u^{k+1,T} K u^{k+1} - |u^{k+1} - z|_{1,M} \\
&= \frac{\beta}{2} u^{k,T} K u^k - \frac{\beta}{2} u^{k+1,T} K u^{k+1} - \langle \beta K u^{k+1}, u^k - u^{k+1} \rangle \\
&\quad + \langle \beta K u^{k+1}, u^k - u^{k+1} \rangle + |u^k - z|_{1,M_I} - |u^{k+1} - z|_{1,M_I} \\
&\geq \frac{\delta}{2} \|u^k - u^{k+1}\|^2 > 0, \quad \text{if } u^{k+1} \neq u^k,
\end{aligned} \tag{23}$$

where $\delta = \beta \min \sigma(K) > 0$ for $\sigma(K)$ denoting the spectrum of the SPD matrix K.

□

Remark 3.1: *The proof of Lemma 3.2 illustrates the effect of the "coercivity parameter"* β *to the overall behavior of the basic Algorithm 1.*

Now, we are ready to state the actual convergence result.

Theorem 3.1: *Algorithm 1 is convergent.*

Proof: First of all, from Proposition 3.1, we know that if $J^{k+1} = J^k$, for some $k \geq 1$, then the pair (u^{k+1}, λ^{k+1}) satisfies the optimality conditions (13), (14). Especially, $u^{k+1} = u^*$, i.e., the unique minimizer of (11).

Assume that $J^{k+1} \neq J^k$. Then, we also have $u^{k+1} \neq u^k$ (due to (15) and the fact that $\bar{u}$ defines $\bar{\lambda}$ uniquely in Lemma 3.1.), and Lemma 3.2 shows that $\mathscr{J}(u^{k+1}) < \mathscr{J}(u^k)$. Hence, $\mathscr{J}(u)$ is decreasing as long as $J^{k+1} \neq J^k$. Because there exists only a finite number of possible active index sets J^k, we must have $J^{k+1} = J^k$ after a finite number of steps. □

3.2. Derivation of Implemented Algorithm

In the actual implementation, Step 2^o of Algorithm 1 is replaced with an inner iteration consisting of a Newton-like search direction followed by a line search. We derive and describe these procedures next.

3.2.1. Description and Analysis of the Inner Iteration

The optimality condition (17) in Lemma 3.1 can be decomposed into block form with respect to two index subsets I and J as follows:

$$\beta \begin{bmatrix} K_{II} & K_{IJ} \\ K_{JI} & K_{JJ} \end{bmatrix} \begin{bmatrix} u_I \\ u_J \end{bmatrix} + \begin{bmatrix} M_I & 0 \\ 0 & M_J \end{bmatrix} \begin{bmatrix} \lambda_I \\ \lambda_J \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}. \tag{24}$$

In the actual realization of Step 2^o, we use the decomposition (24) to obtain a Newton-like step for (17) taking into account the given constraint $P_J(u - z) = 0$ and the definition of the Lagrange multiplier λ on I.

Algorithm 2: Inner Iteration

Step 2.1 Initialize $\tilde{u}^0 = \begin{cases} u_i^k, & i \in I, \\ z_j, & j \in J. \end{cases}$ For $l = 1, \ldots, L_{\max}$, do

(i) Set $\tilde{u}_j = z_j$ for all $j \in J$ and $\tilde{\lambda}_i^l = (\tilde{u}^{l-1} - z)_i / |(\tilde{u}^{l-1} - z)_i|$ for all $i \in I$ such that $\tilde{\lambda}_i^l = 0$ for $\tilde{u}_i^{l-1} = z_i$.

(ii) Solve

$$K_{II}\tilde{u}_I = -\left(\frac{1}{\beta} M_I \tilde{\lambda}_I^l + K_{IJ}\tilde{u}_J\right). \tag{25}$$

(iii) Define $\tilde{d}^{l-1} = \tilde{u} - \tilde{u}^{l-1}$ and determine $\tilde{u}^l = \tilde{u}^{l-1} + t\tilde{d}^{l-1}$, where t is obtained using a line search.

(iv) If stopping criterion for the inner iteration is satisfied, then STOP.

Step 2.2 Take $u^{k+1} = \tilde{u}^l$ and $\lambda_I^{k+1} = \tilde{\lambda}_I^l$ with the final l in Step 2.1.

Update $\lambda_j^{k+1} = -\beta(Ku^{k+1})_j / M_j$ for all $j \in J$.

Remark 3.2:

(i) Due to the projection of u^k in the initialization of Step 2.1, $\tilde{u}^0$ may not coincide with u^k so that the pairs $(\mathscr{J}(\tilde{u}^0),\ \partial\mathscr{J}(\tilde{u}^0))$ and $(\mathscr{J}(u^k),\ \partial\mathscr{J}(u^k))$ are not necessarily the same.

(ii) Because of the explicit update of λ_I in Step 2.1 (i) and the line search in Step 2.1 (iii), $\tilde{\lambda}_I^l$ is not necessarily the sign of $(\tilde{u}^l - z)_I$. This implies that the behavior of the index subsets as stated in (21) may not be valid in the same form anymore.

(iii) After Step 2.2 of Algorithm 2, the pair $(u_J^{k+1}, \lambda_J^{k+1})$ satisfies the linear part of the optimality condition (17) in Lemma 3.1.

From Remark 3.2, we conclude that the proposed active-set method is semi-implicit in the sense that the Lagrange-multiplier does not satisfy (17) on I unless the inner iteration in Algorithm 2 is solved exactly. However, the given form of Algorithm 2 can be justified as follows:

Theorem 3.2: *If $I \neq \emptyset$, then $\tilde{d}^{l-1}$ in Algorithm 2 gives a descent direction for $\mathscr{J}_I(\tilde{u}^{l-1})$.*

Proof: For simplicity, we write $d = \tilde{d}^{l-1}$ and $u = \tilde{u}^{l-1}$, for $l \geq 1$. Assume first that the reduced optimization problem (16) is smooth, i.e., $(u-z)_i \neq 0$ for all $i \in I$. Then, the gradient $g = \nabla\mathscr{J}_I(u)$ is well-defined and single-valued. Let $R_I g$ denote the restriction of the gradient on I. Then a direct calculation shows that d satisfies

$$\beta R_I K R_I d = -R_I g. \tag{26}$$

Due to Step 2.1 (iii) $R_J d = 0$, so that

$$[\nabla\mathscr{J}_I(u)]^T d = -\frac{1}{\beta}(R_I g)^T [R_I K R_I]^{-1} R_I g < 0, \tag{27}$$

because $R_I K R_I$ is positive definite and $I \neq \emptyset$. Thus, d gives a descent direction for $\mathscr{J}_I(u) = \mathscr{J}_I(\tilde{u}^{l-1})$.

Assume next that there exists at least one index $i \in I$ for which $(u-z)_i = 0$, so that problem (16) becomes non-smooth. Let us, instead of choosing $\lambda_i = 0$ for $u_i = z_i$ at Step 2.1 (i), choose an arbitrary subgradient $\xi \in \partial\mathscr{J}_I(u)$ such that $|\lambda_i| \leq 1$ for $u_i = z_i$. Then, exactly as above, a direct calculation shows that d satisfies

$$\beta R_I K R_I d = -R_I \xi.$$

Moreover, due to the same arguments as above, we have

$$\xi^T d = -\frac{1}{\beta}(R_I \xi)^T [R_I K R_I]^{-1} R_I \xi < 0.$$

Since this holds for any subgradient $\xi \in \partial \mathscr{J}_I(u)$, d gives a descent direction for $\mathscr{J}_I(u) = \mathscr{J}_I(\tilde{u}^{l-1})$ [18]. □

Remark 3.3: Notice that the proof of Theorem 3.2 shows that during the inner iterations a descent direction is always obtained *assuming that system (25) is solved exactly*. Hence, the reduced optimization problem in Step 2^o of the basic Algorithm 1 can be solved exactly with a proper line search so that then (21) and Theorem 3.1 are valid. In particular, in this case u^k already coincides with z on J. In practise, the linear system (25) is solved iteratively using the method described in Sect. 3.2.2.

Next, we describe the line search. For this purpose, we use a combination of quadratic interpolation and the Armijo rule [1]:

Algorithm 3: Line Search

(i) Fix scalars $s > 0$, $\gamma \in (0,1)$, and $\sigma \in (0, \frac{1}{2})$. Choose a minimal step length $\underline{t}$ and a termination criterion $\varepsilon > 0$. Let u be the current solution, $g \in \partial \mathscr{J}(u)$ the corresponding subgradient (defined here by $\beta K u + M\lambda$ for $\lambda_i = \text{sign}((u-z)_i)$), and d the given search direction.

(ii) If $d^T g > 0$, then stop; d is not a descent direction. If $d^T g > -\varepsilon$, then skip the line search and return the step length and next solution as $(t, v) = (0, u)$.

(iii) Take $t = 1$ and $v = u + d$. If v satisfies the Wolfe condition

$$\mathscr{J}(u) - \mathscr{J}(v) \geq -\sigma t s g^T d, \tag{28}$$

then return the step length and next solution (t, v).

(iv) Search t using quadratic interpolation with $\mathscr{J}(u)$, $d^T g$ and $\mathscr{J}(v)$. If $(t, u + td)$ satisfy the Wolfe condition, return them.

(v) Armijo rule: Search $t = \gamma^{m_k} s$, where m_k is the first nonnegative integer for which the Wolfe condition is satisfied. If $t \geq \underline{t}$, then return $(t, u + td)$; else return $(0, u)$.

Finally, we have chosen the following initialization and stopping criterion for the inner iteration:

Initialization: Choose $u^0 = z$ and solve λ^0 from the optimality system $\beta K u^0 + M\lambda^0 = 0$.

Termination of inner iteration: If the line search returns $t = 0$ or $l = L_{\max}$ or the relative cost function error is "small enough", i.e.,

$$r_{\mathscr{J}} = \frac{\mathscr{J}(\tilde{u}^l) - \mathscr{J}(\tilde{u}^{l-1})}{\max\{\varepsilon, \mathscr{J}(\tilde{u}^{l-1})\}} \leq \varepsilon. \tag{29}$$

3.2.2. Solution of Linear System (25)

A crucial step in the active-set algorithm is the solution of the reduced linear system in Step 2.1 (ii) of Algorithm 2:

$$(Ku)_i = f_i \quad \text{for } i \in I, \quad u_j = z_j \quad \text{for } j \in J, \tag{30}$$

which must be solved once for every inner iteration. The solution of the linear system (30) in the present case is especially difficult, because the geometry arising from the index subset I is arbitrary, i.e., it has no regularity. Fortunately, in [16], we developed and thoroughly tested an efficient multigrid method suitable for such nonregular (stochastic) geometries. Being a multilevel method with restriction and prolongation operators our current implementation is suitable for mesh sizes of the form $h = 2^{-\bar{m}}$, for $1 \leq \bar{m} \leq 10$, in 2D.

Hence, we solve the reduced linear system (30) iteratively by using a multigrid preconditioned conjugate gradient (CG) method with the termination criterion

$$\max_{1 \leq i \leq N} |(f - Ku)_i| \leq \varepsilon. \tag{31}$$

Due to the inexact solving of the linear system we further modify the update formula of λ_i in Algorithm 2. Namely, we replace the sign-function with the piecewise linear approximation

$$\lambda_i = \frac{(u - z)_i}{\max\{\varepsilon, |(u - z)_i|\}}, \quad \text{for all } i \in I. \tag{32}$$

Finally, we note that in Algorithm 3, (31), and (32), we use the same parameter ε.

4. Numerical Experiments

In this section, we present the results of numerical experiments that were computed using the proposed algorithms. The general purposes of the experiments were to study the convergence of the algorithm with the proposed choice of free parameters, compare the restoration properties between L^1- and L^2-formulations, and propose and test a heuristic method for determining the regularization parameter β in (11).

All experiments were performed on an HP9000/J280 workstation (180 MHz PA8000 CPU) and the algorithms were implemented with Fortran-77. The finite element mesh sizes were taken as $h = 2^{-\bar{m}}$ for an integer $\bar{m}$. The number of discretization points for the test problems is given by $n = n_1^2$ for $n_1 = h^{-1} - 1$. This means that the total number of unknowns in our experiments vary from 3 969 for $\bar{m} = 6$ to 1 046 529 for $\bar{m} = 10$.

In the presented experiments, the following choices were made for the free parameters: $c = 10, L_{\max} = 6, \underline{t} = 10^{-4}$, and $s = 1.0$, $\gamma = 0.5$, and $\sigma = 0.25$. Naturally, some trial and error tests were carried out for obtaining these values, especially for $L_{\max}$ and γ. The active-set algorithm seemed to be quite insensitive with respect to the value of c around the chosen value. Moreover, the values of $\underline{t}$, s, and σ are supposed to be near to the usual ones suggested on other occasions (cf. [1]).

4.1. Implemented Algorithms

First, we recall a compact presentation of the actual active-set algorithm as derived in Sect. 3.2.

Algorithm 4: Implemented Active-Set Method

Step 0^o: Choose $c > 0$ and $\varepsilon > 0$; initialize $u^0 = z$ and λ^0 according to $\beta K u^0 + M\lambda^0 = 0$. Set $k = 0$.

Step 1^o: Determine the active set $J = J^k = \{j \in N : |\lambda_j^k + c(u^k - z)_j| \leq 1\}$ and set $I = I^k = N \setminus J^k$.

Step 2^o: Project $u^k = z$ on J and initialize $\tilde{u}^0 = u^k$. For $l = 1, \dots, L_{\max}$, do

(i) Set $\tilde{\lambda}^l = (\tilde{u}^{l-1} - z)/\max\{\varepsilon, \tilde{u}^{l-1} - z\}$ on I.

(ii) Set $\tilde{u}_J = z_J$ and solve $\tilde{u}_I$ from $K_{II}\tilde{u}_I = -\left(\frac{1}{\beta}M_I\tilde{\lambda}_I^l + K_{IJ}\tilde{u}_J\right)$ using the methodology described in Sect. 3.2.2.

(iii) Define $\tilde{d}^{l-1} = \tilde{u} - \tilde{u}^{l-1}$ and update $\tilde{u}^l = \tilde{u}^{l-1} + t\tilde{d}^{l-1}$, where t is determined using Algorithm 3.

(iv) If the *termination of inner iteration* -conditions as described in Sect. 3.2.1. are valid, then terminate Step 2^o.

Step 3^o:

(i) If Step 2^o was terminated with $l = 1$ and $t = 0$, i.e. no decrease of the cost functional during the inner iteration was obtained, then stop the algorithm. The solution is u^k.

(ii) Otherwise, take $u^{k+1} = \tilde{u}^l$, $\lambda_I^{k+1} = \tilde{\lambda}_I^l$ with the final l in Step 2^o, and update $\lambda_j^{k+1} = -\beta(Ku^{k+1})_j/M_j$ for all $j \in J$. Set $k = k + 1$ and go to Step 1^o.

Notice that the stopping criterion above for the outer iteration yields $J^{k+1} = J^k$ similarly to Algorithm 1.

As a reference method for solving the discrete nonsmooth optimization problem (11), we use the Uzawa method which is also based on the regularized optimality conditions in (13).

Algorithm 5: Uzawa Method

Step 0^o: Initialize $k = 0$, $(u^0, \lambda^0) = (z, 0)$, and choose a "small enough" $c > 0$.

Step 1^o: Solve u^{k+1} from

$$\beta K u^{k+1} = -M\lambda^k.$$

Step 2^o: If $|u^{k+1} - u^k|_\infty \leq \varepsilon$, then STOP; the solution is u^{k+1}.

Step 3^o: Update

$$\lambda_i^{k+1} = \frac{\lambda_i^{k+1} + c(u^{k+1} - z)_i}{\max\{1, |\lambda_i^{k+1} + c(u^{k+1} - z)_i|\}}, \quad \text{for all } i \in N,$$

set $k = k + 1$, and go to Step 1^o.

In Step 1^o, we apply a FFT-based solver for inverting the discrete Laplacian.

4.2. Analytic Example

We use the following basic example for testing the algorithms and their restoration properties.

Example 4.1: Let $\Omega = (0, 1)^2$ be the unit square in $\mathbb{R}^2$. Take the true image to be the pointwise values on the given mesh of $z^* = z_1 z_2$ for

$$z_1(x, y) = \sin(2\pi x)\sin(2\pi y)$$

and

$$z_2(x, y) = \begin{cases} \sin(\pi y), & \text{for} \quad x \le \frac{1}{2}, \\ \sin(\pi y) + (x - \frac{1}{2}), & \text{for} \quad \frac{1}{2} < x \le 1 \ . \end{cases}$$

Notice that $z^* \in H^1(\Omega)$ (cf. [23]).

The noisy image z is generated from z^* by adding two types of noise: outliers and normally distributed (white) noise. Let $O \subset N$ be a subset of the whole index set N that will contain the indices for outliers. We define the locations of outliers using the following procedure:

(1) Choose the desired percentage p_l of linewise outliers. Set to O L-shaped and W-shaped sets of indices (that can overlap each other) in random locations such that $\text{card}(O) \simeq \frac{p_l}{100} N$ (up to the nearest integer), where $\text{card}(\cdot)$ denotes the cardinality (number of points) of an index subset.

(2) Choose the desired amount n_s of ball-shaped (domainwise) outliers for which each one occupies approximately 0.5 percent of $\text{card}(N)$ (up to the nearest integer). Add n_s index subsets defining a ball with centre at random location into O such that these balls do not overlap with the previous linewise outliers.

(3) Choose the desired percentage p_p of pointwise outliers. Add $\frac{p_p}{100} N$ number of indices (up to the nearest integer) at random locations into O such that they do not overlap with the previous outliers.

The resulting pattern of outliers is illustrated in Fig. 1 (right). Here, all random locations are generated using the NAG-routine G05DYF. Altogether, if we choose $p_l = n_s = p_p = 1$, then we have about 2.5 percent of all pixels in the image occupied by the ourliers.

Finally, the noisy image z is formed from z^* by taking

$$z_i = z_i^* + O_i \mathcal{U}_i + (1 - O_i)\mathcal{N}_i,$$

where $\mathcal{U}$ denotes the pseudo-random uniform distribution over the interval $[-2, 2]$ (NAG-routine G05DAF) and $\mathcal{N}$ the pseudo-random normal distribution with mean 0 and standard deviation σ (NAG-routine G05DDF). The obtained noisy image z is illustrated in Fig. 1 (left).

4.3. Numerical Results

4.3.1. Convergence Studies

In Tables 1–2, we compare the convergence of the Uzawa method and the active-set method (ASM). In the tables, "Its." denotes the total number of Uzawa iterations and "CPU" contains the elapsed CPU time. Moreover, "AS-Its." includes the number of active-set (outer) iterations, "Inner-Its." the total number of inner iterations according to Algorithm 2, "PCG-Its." tells the total number of preconditioned CG-iterations, and "Fun.Evals." contains the total number of function evaluations in the line search Algorithm 3. CPU times in Tables 1–2 are further illustrated in Fig. 3.

In these experiments, we have chosen $\sigma = 0.1$ and $\varepsilon = 10^{-7}$. Notice that due to the fixed termination criterion, the accuracy of the final cost functional $\mathcal{J}(u^*)$ is decreasing along the mesh size (as suggested by the number of iterations taken by the algorithms).

We draw the following conclusions from Tables 1–2:

(i) ASM was always convergent with the proposed choices of free parameters. This was also true for the examples in Sect. 4.3.2.

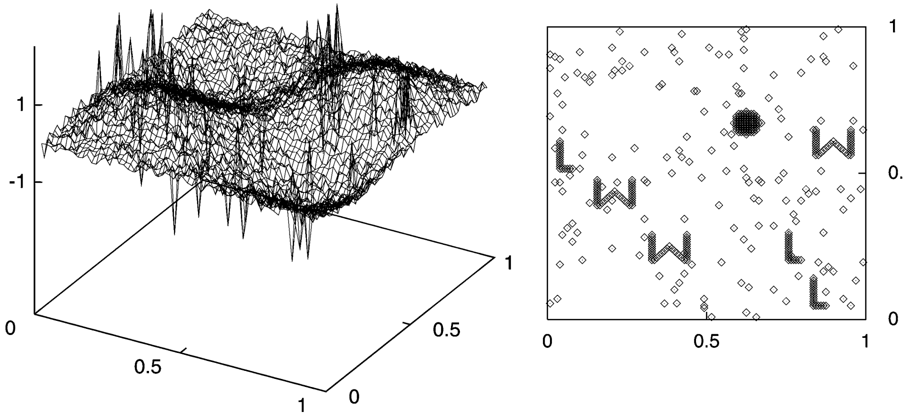

Fig. 1. Noise pattern with $p_l = n_s = p_p = 1$ and $\sigma = 0.1$: noisy image z for $\bar{m} = 6$ (left), locations of outliers for $\bar{m} = 7$ (right)

Table 1. Uzawa ($c = 0.01$) and ASM for $\beta = 5 \cdot 10^{-3}$, and $p_l = n_s = p_p = 1$

	$\bar{m}$	6	7	8	9	10
UZAWA	$\mathcal{J}(u^*)$	$12.65 \cdot 10^{-2}$	$13.56 \cdot 10^{-2}$	$13.73 \cdot 10^{-2}$	$13.80 \cdot 10^{-2}$	
	Its.	27 689	36 402	31 025	21 261	
	CPU	103.9	536.3	2 036.4	7 158.5	
ASM	$\mathcal{J}(u^*)$	$12.65 \cdot 10^{-2}$	$13.56 \cdot 10^{-2}$	$13.73 \cdot 10^{-2}$	$13.80 \cdot 10^{-2}$	$13.82 \cdot 10^{-2}$
	AS-Its.	23	18	13	9	7
	Inner-Its.	84	56	37	31	30
	PCG-Its.	734	555	386	342	351
	Fun.Evals.	558	311	141	88	73
	CPU	1.9	9.1	34.7	144.5	599.2

Table 2. Uzawa (c optimal, i.e., largest possible yielding fastest convergence; sought by hand) and ASM for $p_l = n_s = p_p = 1$

	$\bar{m}/\beta$	$6/10^{-4}$	$7/5 \cdot 10^{-4}$	$8/10^{-3}$	$9/5 \cdot 10^{-3}$	$10/10^{-2}$
UZAWA	$\mathcal{J}(u^*)$	$2.53 \cdot 10^{-2}$	$8.86 \cdot 10^{-2}$	$10.64 \cdot 10^{-2}$	$13.80 \cdot 10^{-2}$	
	c	$3 \cdot 10^{-3}$	$2 \cdot 10^{-2}$	$4 \cdot 10^{-2}$	$2 \cdot 10^{-1}$	
	Its.	7 219	10 799	15 436	7 303	
	CPU	26.8	157.7	962.7	2 463.9	
ASM	$\mathcal{J}(u^*)$	$2.53 \cdot 10^{-2}$	$8.86 \cdot 10^{-2}$	$10.64 \cdot 10^{-2}$	$13.80 \cdot 10^{-2}$	$16.99 \cdot 10^{-2}$
	AS-Its.	10	42	18	9	9
	Inner-Its.	48	165	59	31	34
	PCG-Its.	361	1 424	625	342	396
	Fun.Evals.	372	1 367	353	88	80
	CPU	0.6	25.2	58.1	146.3	684.2

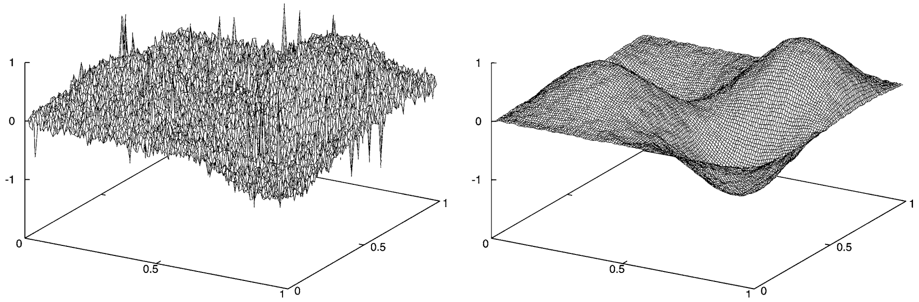

Fig. 2. Noisy image z for $\bar{m} = 7, p_l = n_s = p_p = 1$, and $\sigma = 0.1$ (left) and its reconstruction u^* with $\beta = 3 \cdot 10^{-3}$ (right)

(ii) ASM was always significantly faster than Uzawa, although with a proper choice of c, which depends on both the size and structure (here form of noise) of the problem, the run time of Uzawa could be substantially decreased. For Uzawa, the larger β is the larger c can be used, so that the improvement of coercivity of the original optimization problem can decrease the run time of suitably designed Uzawa method. For ASM, there seems to be no direct connection between the coercivity of the problem and the efficiency of the algorithm (cf. Table 2) and Fig. 3 shows a linear growth in CPU time with respect to the number of unknowns.

(iii) Concerning the convergence of the active-set method we can comment that without the projection $u^k = z$ on J in the initialization of the inner iterations

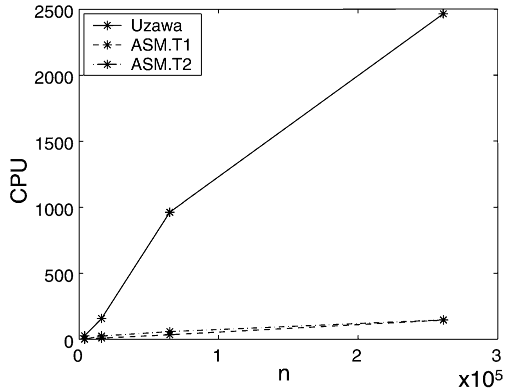

Fig. 3. CPU times for the Uzawa method in Table 2 and ASM in Tables 1 and 2 for $\bar{m} \in \{6, 7, 8, 9\}$

in Algorithm 2, it may happen that the search direction $\tilde{d}^{l-1}$ is NOT descent, i.e., $d^T g > 0$.

4.3.2. Restoration Studies

Here, we study the denoising and the resulting restoration properties of the proposed formulations. The error between the true image z^* and its reconstruction u is measured by using the following quantities:

mean square deviation: $e_m(u) = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(u - z^*)_i^2}$,

absolute maximum: $e_a(u) = \max_{i \in N} |(u - z^*)_i|$,

combination: $e_{ma}(u) = e_m(u) \cdot e_a(u)$.

Usually, the reconstruction capability is tested using only $e_m(u)$. However, e.g., in medical diagnostics, the actual image values may be as important as the average behaviour. The proposed measures reflect also the characteristic features of the L^2- and L^1-formulations: Roughly speaking, the L^2-formulation yields locally smoothed averaging of the noisy image while the L^1-formulation yields locally smoothed median values. Therefore, the L^2-formulation should favor $e_m(u)$ as an error indicator, while for the L^1-formulation, $e_a(u)$ should be a more suitable measure. By the product of these two, we form a simplest possible compound measure for the overall quality of the recovered image.

Hence, Table 3 contains the reconstruction errors for the following cases:

(1) $e_\alpha(u_{L^2})$, $\alpha = m/a/ma$, contains the error measures for the L^2-formulation.

(2) $e_\alpha(u^*)$, $\alpha = m/a/ma$, corresponds to the L^1-formulation with the whole active-set algorithm for the stopping criterion $\varepsilon = 10^{-7}$ (cf. Tables 1–2).

(3) $e_\alpha(u^2)$, $\alpha = m/a/ma$, gives the errors after only TWO iterations of the active-set algorithm.

To study the full denoising potential of different approaches, the reconstruction errors, given in Table 3, have been computed with the corresponding "optimal" value β_α of the regularization parameter β, i.e., with the value that yields the minimum error $e_\alpha(u)$ (β_α is sought iteratively by hand). The linear L^2-problem (10) is solved using a FFT-based solver. An example of the original and reconstructed images is given in Fig. 2.

Let us draw the following conclusions concerning the restoration properties of different algorithms and formulations according to Tables 3 and 4:

Full versus partial ASM: By means of $e_m(u)$, only two iterations of ASM gives usually smaller error than the more accurate solving of the optimization problem. However, by means of $e_a(u)$ and $e_{ma}(u)$, the full algorithm is better and hence more robust, especially with larger images ($\bar{m} = 8$ or 9) with and without outliers. The reason for the success of two iterations with $e_m(u)$ is due to the characteristic behaviour of the ASM algorithm: Namely, the active set J^1 for the second iteration tends to be smaller than the true set J^* which contains the components of the solution being in contact with the noisy observation z. Therefore, in average we can obtain smaller reconstruction error with the incomplete ASM, although the full algorithm is more accurate pointwise.

L^1- versus L^2-formulation: With only Gaussian noise, the L^2-formulation is slightly better than the L^1-formulation for all measured error quantities. On the contrary,

Table 3. Obtained errors for β_α, $\alpha = m/a/ma$, with $p_l = n_s = p_p = 0$ and $\sigma = 0.1$ (Gaussian noise) on the left column; $p_l = n_s = p_p = 1$ and $\sigma = 0.1$ (Gaussian noise with ~2.5% pointwise, linewise, and subdomain-wise outliers) on the right column

$\bar{m}$	6		7		8		9	
Noise	Gaussian	Inc. outl.	Gaussian	Inc. outl.	Gaussian	Inc. outl.	Gaussian	Inc. outl.
$e_m(u_L^2)$	$2.6 \cdot 10^{-2}$	$4.3 \cdot 10^{-2}$	$1.8 \cdot 10^{-2}$	$2.8 \cdot 10^{-2}$	$1.1 \cdot 10^{-2}$	$1.7 \cdot 10^{-2}$	$6.7 \cdot 10^{-3}$	$1.0 \cdot 10^{-2}$
$e_a(u_L^2)$	$8.8 \cdot 10^{-1}$	$2.3 \cdot 10^{-1}$	$7.1 \cdot 10^{-2}$	$2.1 \cdot 10^{-1}$	$4.3 \cdot 10^{-2}$	$1.0 \cdot 10^{-1}$	$2.7 \cdot 10^{-2}$	$6.0 \cdot 10^{-2}$
$e_{ma}(u_L^2)$	$2.3 \cdot 10^{-3}$	$1.2 \cdot 10^{-2}$	$1.3 \cdot 10^{-3}$	$9.2 \cdot 10^{-3}$	$4.6 \cdot 10^{-4}$	$2.5 \cdot 10^{-3}$	$1.9 \cdot 10^{-4}$	$8.9 \cdot 10^{-4}$
$e_m(u^*)$	$2.9 \cdot 10^{-2}$	$2.9 \cdot 10^{-2}$	$2.0 \cdot 10^{-2}$	$2.0 \cdot 10^{-2}$	$1.3 \cdot 10^{-2}$	$1.3 \cdot 10^{-2}$	$8.1 \cdot 10^{-3}$	$7.9 \cdot 10^{-3}$
$e_a(u^*)$	$1.1 \cdot 10^{-1}$	$9.5 \cdot 10^{-2}$	$7.0 \cdot 10^{-2}$	$1.0 \cdot 10^{-1}$	$4.5 \cdot 10^{-2}$	$5.0 \cdot 10^{-2}$	$3.0 \cdot 10^{-2}$	$3.3 \cdot 10^{-2}$
$e_{ma}(u^*)$	$3.2 \cdot 10^{-3}$	$2.8 \cdot 10^{-3}$	$1.4 \cdot 10^{-3}$	$2.1 \cdot 10^{-3}$	$5.7 \cdot 10^{-4}$	$6.6 \cdot 10^{-4}$	$2.4 \cdot 10^{-4}$	$2.6 \cdot 10^{-4}$
$e_m(u^2)$	$2.8 \cdot 10^{-2}$	$2.9 \cdot 10^{-2}$	$2.0 \cdot 10^{-2}$	$2.0 \cdot 10^{-2}$	$1.2 \cdot 10^{-2}$	$1.3 \cdot 10^{-2}$	$8.2 \cdot 10^{-3}$	$7.9 \cdot 10^{-3}$
$e_a(u^2)$	$1.1 \cdot 10^{-1}$	$9.7 \cdot 10^{-2}$	$7.0 \cdot 10^{-2}$	$1.0 \cdot 10^{-1}$	$6.5 \cdot 10^{-2}$	$5.8 \cdot 10^{-2}$	$4.5 \cdot 10^{-2}$	$5.4 \cdot 10^{-2}$
$e_{ma}(u^2)$	$3.0 \cdot 10^{-3}$	$2.9 \cdot 10^{-3}$	$1.4 \cdot 10^{-3}$	$2.0 \cdot 10^{-3}$	$7.9 \cdot 10^{-4}$	$8.3 \cdot 10^{-4}$	$4.5 \cdot 10^{-4}$	$4.3 \cdot 10^{-4}$

Table 4. Comparison of obtained regularization parameters β_m and β^* for $p_l = n_s = p_p = 0$ and σ =0.1 (rows G); $p_l = n_s = p_p = 1$ and $\sigma = 0.1$ (rows O)

		6		7		8		9	
Case	$\bar{m}$	β_m	β^*	β_m	β^*	β_m	β^*	β_m	β^*
G	u_{L^2}	$9 \cdot 10^{-4}$	$8 \cdot 10^{-4}$	$5 \cdot 10^{-4}$	$7 \cdot 10^{-4}$	$3 \cdot 10^{-4}$	$5 \cdot 10^{-4}$	$2 \cdot 10^{-4}$	$3 \cdot 10^{-4}$
	u^*	$4 \cdot 10^{-3}$	$2 \cdot 10^{-4}$	$2 \cdot 10^{-3}$	$4 \cdot 10^{-3}$	$2 \cdot 10^{-3}$	$2 \cdot 10^{-3}$	$9 \cdot 10^{-4}$	$1 \cdot 10^{-3}$
	u^2	$5 \cdot 10^{-3}$	$2 \cdot 10^{-4}$	$3 \cdot 10^{-3}$	$3 \cdot 10^{-3}$	$2 \cdot 10^{-3}$	$3 \cdot 10^{-3}$	$9 \cdot 10^{-4}$	$1 \cdot 10^{-3}$
O	u_{L^2}	$2 \cdot 10^{-3}$	$1 \cdot 10^{-3}$	$9 \cdot 10^{-4}$	$1 \cdot 10^{-3}$	$6 \cdot 10^{-4}$	$9 \cdot 10^{-4}$	$3 \cdot 10^{-4}$	$5 \cdot 10^{-4}$
	u^*	$4 \cdot 10^{-3}$	$2 \cdot 10^{-4}$	$3 \cdot 10^{-3}$	$3 \cdot 10^{-3}$	$2 \cdot 10^{-3}$	$2 \cdot 10^{-3}$	$9 \cdot 10^{-4}$	$1 \cdot 10^{-3}$
	u^2	$4 \cdot 10^{-3}$	$2 \cdot 10^{-4}$	$3 \cdot 10^{-3}$	$5 \cdot 10^{-3}$	$2 \cdot 10^{-3}$	$2 \cdot 10^{-3}$	$9 \cdot 10^{-4}$	$9 \cdot 10^{-4}$

with outliers the L^1-formulation yields better reconstruction for all measures, especially by means of $e_a(u)$ and $e_{ma}(u)$ (see Fig. 4). Hence, in general, reconstruction using the L^1-formulation is more robust than with the L^2-formulation (cf. intrinsic assumptions on noise distribution and their consequences in Sect. 1). For both formulations, the optimal regularization parameter β_α seems to be quite insensitive to the existence of outliers. This suggests that the optimal value of β is determined by the amount of Gaussian noise contained in the image. Finally, let us remark that an "optimal" ratio $\beta_\alpha(u^*)/\beta_\alpha(u_{L^2})$, $\alpha = m/a/ma$, seems to be four rather than two, which is the value that could be expected by a comparison of the optimality conditions (7) and (10) for the two formulations.

Size of problem: The larger problem we have the better reconstruction we obtain in all cases (cf. Fig. 4). The smaller (i.e., the sparser) the problem is, the more superior is the L^1-formulation for the data containing outliers. We note also that when the size of the problem increases, the difference between errors $e_{ma}(u_{L^2})$ and $e_{ma}(u^*)$ gets smaller. This observation is in agreement with the breakdown point properties of statistical estimates corresponding to L^2- and L^1-norms [25].

Behavior of error measures: According to numerical tests, in almost all cases all the error measures $e_\alpha(u)$, $\alpha = m/a/ma$, are strictly convex with respect to β. We found one case for $(\beta_a(u^2), e_a(u^2))$ ($\bar{m} = 9$ in Table 3 for $p_l = n_s = p_p = 1$ and $\sigma = 0.1$), where this was not the case. This illustrates one side effect of inexact solving of the optimization problem when computing u^2.

4.3.3. Choice of the Regularization Parameter

Next, we propose a heuristic method for choosing the regularization parameter β in a "nearly optimal" way with respect to the reconstruction error. Since a comparison of different error measures was done in the previous section, we concentrate here on one error measure, $e_m(u)$, only. We remind that in true applications, one has usually no *a priori* knowledge of the amount and shape of the error distribution contained in the given image z. Therefore, we can only try to

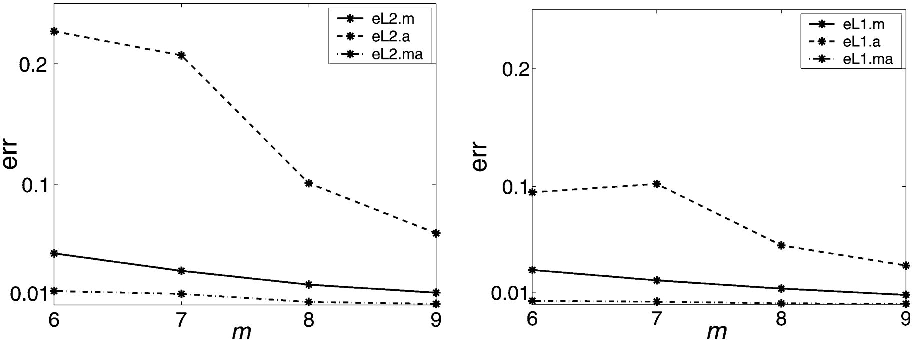

Fig. 4. Errors for the L^2-formulation (u_{L^2}, left) and L^1-formulation (u^*, right) in Table 3 for $p_l = n_s = p_p = 1$ and $\sigma = 0.1$

determine β using computable quantities not depending on the noise statistics or the true image. The present technique yields a slight improvement to the one presented in [13], in the sense that through (35) the best value of β is more distinguishable.

From the given formulations, we see that the cost functional to be minimized consists of two parts

$$\mathscr{J}(u) = r_\beta(u) + f(u), \tag{33}$$

where $r_\beta(u)$ represents the regularization $\int_\Omega \frac{\beta}{2}|\nabla u|^2 dx$ and $f(u)$ the fitting with respect to the given data z, i.e., $f(u) = \int_\Omega |u - z| dx$ in the L^1-case or $f(u) = \int_\Omega |u - z|^2 dx$ in the L^2-case. The choice of the regularization parameter β determines the balance between the two terms in (33).

We use the notation β^* for the regularization parameter which will be obtained using the proposed heuristic method, to distinguish it from the hand-tuned "optimal" one β_m of the previous section. Our suggestion for choosing β^* is based on the concept of sensitivity of $r_\beta(u)$ with respect to the regularization parameter. Assume that we have solved the optimization problem with some reasonable amount of regularization parameters

$$0 < \beta_0 < \beta_1 < \ldots < \beta_{n_\beta - 1} < \beta_{n_\beta} \tag{34}$$

with corresponding solutions $u_0, u_1, \ldots, u_{n_\beta-1}, u_{n_\beta}$ for $u_i = u_i^*$ or $u_i = u_i^2$ or $u_i = u_{L_2,i}$ (we use the notations introduced in the previous section). We propose to choose for the regularization parameter the value β^*, which maximizes the (forward difference-like) quantity

$$\frac{\partial r_{\beta_i}}{\partial \beta} = \frac{r_{\beta_{i+1}}(u_{i+1}) - r_{\beta_i}(u_i)}{\beta_{i+1} - \beta_i}. \tag{35}$$

To illustrate our suggestion, we give plots of the reconstruction error $e_m(u)$ (bottom) and $\frac{\partial r_{\beta_i}}{\partial \beta}$ (top) for four test cases in Figs. 5–6. The computational test values of the regularization parameter β were chosen as

$$\begin{aligned}\beta_0 = 10^{-4} < 2 \cdot 10^{-4} < \ldots < 9 \cdot 10^{-4} < 10^{-3} < 2 \cdot 10^{-3} < \ldots \\ < 9 \cdot 10^{-3} < 10^{-2} = \beta_{n_\beta},\end{aligned}$$

so that we have $n_\beta = 18$. We point out that in the figures, a logarithmic scale is used for the x-axis. In these figures, both the pointwise minimum of $e_m(u)$ and the pointwise maximum of $\frac{\partial r_{\beta_i}}{\partial \beta}$ are marked. In Figs. 5–6, the results are given for u_{L_2} and u^*, respectively. In the plots on the left, we have $p_l = n_s = p_p = 0$ and on the right, $p_l = n_s = p_p = 1$. Moreover, we have chosen $\sigma = 0.1$, so that the examples correspond to the results presented in Table 3. From the figures, we see that the β^* maximizing $\frac{\partial r_{\beta_i}}{\partial \beta}$ gives a good approximation of β_m minimizing $e_m(u)$.

The heuristic is tested more thoroughly in Table 4. As can be seen, we obtain values of β^* that are very close to the "optimal" choices β_m. Only for $\bar{m} = 6$,

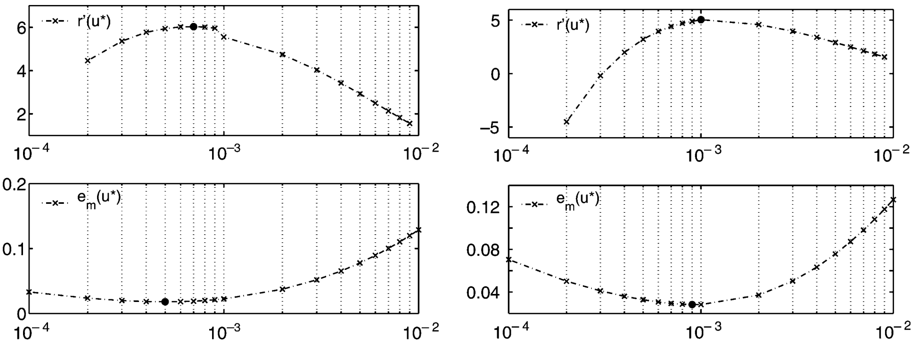

Fig. 5. Error plots for u_{L^2} and $\bar{m} = 7$

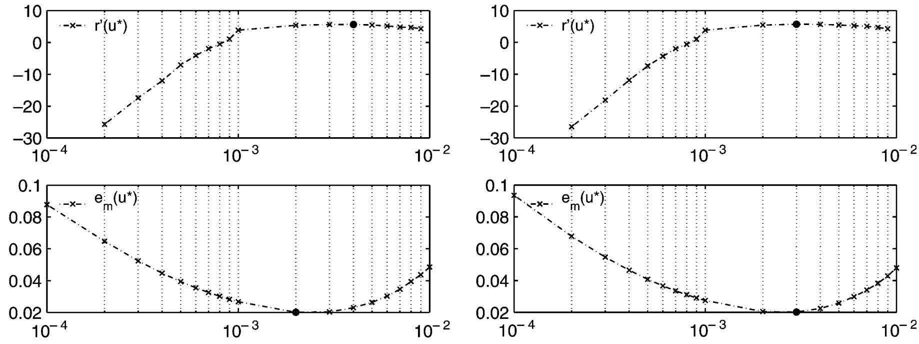

Fig. 6. Error plots for u^* and $\bar{m} = 7$

the heuristic does not seem to work in the L_1-case. From Fig. 7, we see that $\frac{\partial r_{\beta_i}}{\partial \beta}$ oscillates for u^* and $\bar{m} = 6$ when $\beta < 10^{-3}$. Near the best value of β, however, the plot behaves nicely and the maximum gives a good approximation of β_m. This suggests that the search region for β needs to be chosen appropriately. To conclude, the difference between the "optimal" β_m and the value β^* obtained using the heuristic method was generally between 0–3 units on the given grid of β values. A very good agreement was obtained with u^* for $\bar{m} \geq 7$.

To shortly summarize our findings, the proposed heuristic yields a localization of the effective values of the regularization parameter, without *a priori* knowledge about structure or amount of noise. It is effective for various error distributions and different formulations. For accuracy reasons, however, the technique needs good *a priori* knowledge of the search interval and a fine enough grid of β values to be used.

4.4. Real Example: Denoising Lens-paper Burn Data

Let us shortly consider a real example of denoising lens-paper burn data [19]. Example involves slow-combustion (smouldering) front propagating in a sheet of

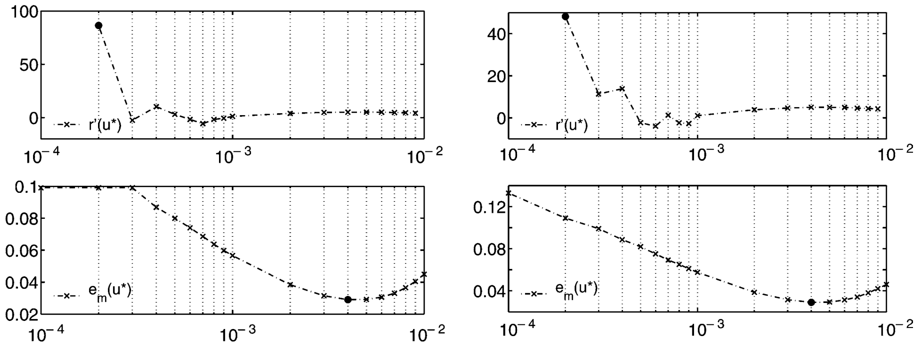

Fig. 7. Error plots for u^* and $\bar{m} = 6$

paper as described more thoroughly in Sect. 1. In these experiments, the propagation of the emerging one-dimensional front was recorded with a CCD-camera system. In Fig. 8 (top), front profiles of a lens-paper burn are shown. Each horizontal curve represents a one-dimensional front at time t. We considered the data as a two-dimensional image. Outliers in the resulting image were caused by, e.g., ash particles detached from the front.

In Fig. 8 (bottom), front profiles are shown after denoising (filtering) the data (top) using ASM. Here, $\beta = 0.01$ was chosen by hand. Formulation (5) is ideal for

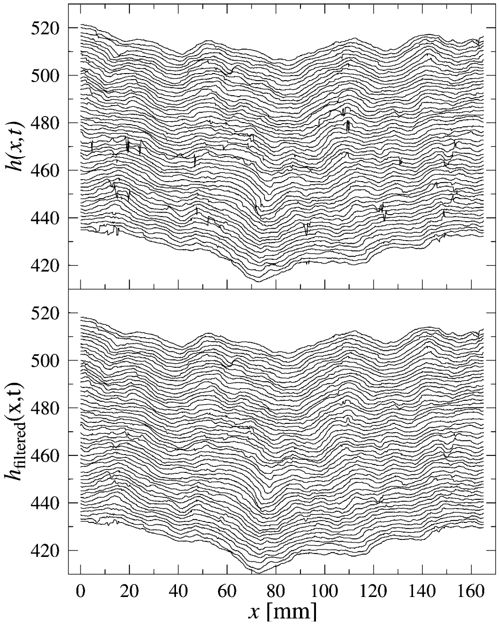

Fig. 8. Front profiles of a lens-paper burn before $(h(x,t))$ and after $(h_{\text{filtered}}(x,t))$ filtering the data [19]

this type of problems. ASM removed the outliers while the data stayed almost untouched elsewhere.

References

[1] Bertsekas, D. P.: Constrained optimization and Lagrange multiplier methods. New York: Academic Press 1982.

[2] Chambolle, A., Lions, P. L.: Image recovery via total variation minimization and related problems Numer. Math. *76*, 167–188 (1997).

[3] Gonzalez, R. C., Woods, R. E.: Digital image processing. Addison-Wesley 1993.

[4] Huber, P. J.: Robust statistics. Wiley Series in Probability and Mathematical Statistics. New York: Wiley 1981.

[5] Ito, K., Kunisch, K.: Augmented Lagrangian formulation of non-smooth, convex optimization in Hilbert spaces. In: Lecture Notes in Pure and Applied Mathematics. Control of Partial Differential Equations and Applications, vol. 174 (E. Casas, ed.), pp 107–117. New York: Marcel Dekker 1995.

[6] Ito, K., Kunisch, K.: Augmented Lagrangian-SQP-methods in Hilbert spaces and application to control in the coefficients problems. SIAM J. Optim. *6*, 96–125 (1996).

[7] Ito, K., Kunisch, K.: An active set strategy based on the augmented Lagrangian formulation for image restoration. Math. Model. Num. Anal. (M2AN) *33*, 1–21 (1999).

[8] Jain, A.: Fundamentals of digital image processing. New Jersey: Prentice Hall 1989.

[9] Kardar, M., Parisi, G., Zhang, Y.-C.: Dynamic scaling of growing interfaces. Phys. Rev. Lett. *56*, 889–892 (1986).

[10] Kärkkäinen, T.: An equation error method to recover diffusion from the distributed observation, Inverse Problems *13*, 1033–1051 (1997).

[11] Kärkkäinen, T., Heikkola, E.: Robust formulations for training multilayer perceptrons. Neural Comput. *16*, 837–862 (2004).

[12] Kärkkäinen, T., Kunisch, K., Tarvainen, P.: Augmented Lagrangian active set methods for obstacle problems. J. Optim. Theory Appl. *119*, 499–533 (2003).

[13] Kärkkäinen, T., Majava, K.: Determination of regularization parameter in monotone active set method for image restoration. In: Proc. 3rd European Conf. on Numerical Mathematics and Advanced Applications (Neittaanmäki, P., Tiihonen, T., Tarvainen, P., ed.), pp. 641–648. Singapore: World Scientific 2000.

[14] Kärkkäinen, T., Majava, K.: Non-monotone and monotone active-set methods for image restoration, Part 1: Convergence analysis. J. Optim. Theory Appl. *106*, 61–80 (2000).

[15] Kärkkäinen, T., Majava, K.: Non-monotone and monotone active-set methods for image restoration, Part 2: Numerical results. J. Optim. Theory Appl. *106*, 81–105 (2000).

[16] Kärkkäinen, T., Toivanen, J.: Building blocks for odd-even multigrid with applications to reduced systems. J. Comp. Appl. Math. *131*, 15–33 (2001).

[17] Majava, K., Kärkkäinen, T.: Active-set algorithms for non-smooth denoising problems. In: Proc. 4th European Congress on Computational Methods in Applied Sciences and Engineering (Neittaanmäki, P., Rossi, T., Korotov, S., Onate, E. Périaux, J., Knörzer, D., eds.). CD-ROM 2004.

[18] Mäkelä, M. M., Neittaanmäki, P.: Non-smooth optimization. Analysis and algorithms with applications to optimal control. Singapore: World Scientific 1992.

[19] Maunuksela, J., Myllys, M., Merikoski, J., Timonen, J., Kärkkäinen, T., Welling, M., Wijngaarden, R.: Determination of the stochastic evolution equation from noisy experimental data. Eur. Phys. J. *B33*, 193–202 (2003).

[20] Moretti, D., Babiloni, F., Carducci, F., Cincotti, F., Remondini, R., Rossini, P., Salinari, S., Babiloni, C.: Computerized processing of EEG-EOG-EMG artifacts for multi-centric studies in EEG oscillations and event-related potentials. Int. J. Psychophysiol. *47*, 199–216 (2003).

[21] Nikolova, M.: Minimizers of cost-functions involving nonsmooth data-fidelity terms. Application to the processing of outliers. SIAM J. Numer. Anal. *40*, 965–994 (2002).

[22] Ollila, E.: Sign and rank covariance matrices with applications to multivariate analysis. PhD thesis, University of Jyväskylä 2002.

[23] Quarteroni, A., Valli, A.: Numerical approximation of partial differential equations. Springer Series in Computational Mathematics, vol. 23. Berlin Heidelberg: Springer 1994.

[24] Rao, C. R.: Methodology based on the l_1-norm, in statistical inference. Sankhya: The Indian Journal of Statistics *50*, 289–313 (1988).

[25] Rousseeuw, P., Leroy, A.: Robust regression and outlier detection. Wiley Series in Probability and Mathematical Statistics. New York: Wiley 1987.

[26] Rudin, L. I., Osher, S., Fatemi, E.: Nonlinear total variation based noise removal algorithms. Physica *D60*, 259–268 (1992).

T. Kärkkäinen
University of Jyväskylä
Department of Mathematical
Information Technology
P.O. Box 35 (Agora)
40014 University of Jyväskylä
Finland
e-mail: tka@mit.jyu.fi

K. Kunisch
University of Graz
Institute for Mathematics
Heinrichstr. 36
8010 Graz
Austria
e-mail: karl.kunisch@kfunigraz.ac.at

K. Majava
University of Jyväskylä
Department of Mathematical
Information Technology
P.O. Box 35 (Agora)
40014 University of Jyväskylä
Finland
e-mail: majkir@mit.jyu.fi