Computing (2007) 81: 137–160
DOI 10.1007/s00607-007-0246-y
Printed in The Netherlands

Computing

# Signal and image approximation with level-set constraints

**C. Schnörr**

Department of Mathematics and Computer Science, University of Heidelberg, Heidelberg, Germany

Received 27 January 2007; Accepted 21 August 2007; Published online 6 November 2007
© Springer-Verlag 2007

**Summary**

We present a novel variational approach to signal and image approximation using filter statistics (histograms) as constraints. Given a set of linear filters, we study the problem to determine the closest point to given data while constraining the level-sets of the filter outputs. This criterion and the constraints are formulated as a bilevel optimization problem. We develop an algorithm by representing the lower-level problem through complementarity constraints and by applying an interior-penalty relaxation method. Based on a decomposition of the penalty term into the difference of two convex functions, the resulting algorithm approximates the data by solving a sequence of convex programs. Our approach allows to model and to study the generation of image structure through the interaction of two convex processes for spatial approximation and for preserving filter statistics, respectively.

*AMS Subject Classifications:* 68U10; 65K05; 65K10; 90C33.

*Keywords:* level-sets; image approximation; equilibrium constraints; complementarity constraints; DC-programming.

## 1. Introduction

Filter statistics play an important role in both natural and computational vision systems [8], [12], [20]. It has been shown, for instance, that the statistics (histograms) of bandpass filter outputs, collected over a large image database, are significant for natural scences and can be accurately described by parametric families of probability distributions.

A related subject that is relevant to our present work concerns the use of filter statistics for representing *subclasses* of images. Zhu and Mumford [22] showed in their seminal work impressive image restoration results by employing filter statistics. A significant difference of their approach to established denoising methods using

Correspondence: Christoph Schnörr, Department of Mathematics and Computer Science, University of Heidelberg, 69120 Heidelberg, Germany (E-mail: christoph.schnoerr@iwr.uni-heidelberg.de)

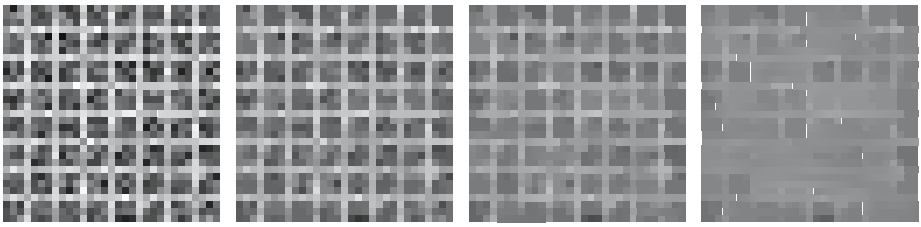

**Fig. 1.** TV-Denoising fails for non-smooth image structure. A grid image superimposed with noise, shown left as image $d$, and three minimizers $u(\alpha)$ of $\|u - d\|^2 + \alpha \mathrm{TV}(u)$ for increasing values of $\alpha$, computed with Chambolle's projection algorithm [5]. Without prior knowledge about image structure, denoising is not possible. The approach introduced in this paper exploits filter statistics as prior knowledge for discriminating structure and noise through image approximation with level-set constraints – see Figs. 5 and 8

TV-regularization [17] as well as to modern related approaches to image decomposition [5], [3], is the ability to *generate* image structure which is not possible when using the latter convex variational models (Fig. 1).

The nonconvex approach of Mumford and Shah involves two phases, learning and inference. In the learning phase, a Gibbs distribution

$$p(u) = \frac{1}{Z} \exp\left(-E(u)\right), \tag{1}$$

defined on the image space $\mathbb{R}^n$ together with coarsely quantized coordinates, is determined by maximizing the entropy of $p$ subject to the constraint that samples $u \sim p(u)$ reproduce given filter statistics on the average [23]. Stochastic sampling was applied to cope with this difficult optimization problem. Inference, on the other hand, amounts to approximate given image data $d$ by a function $u$ using the learned energy function $E(u)$ as a regularizing term. Because the latter is highly nonconvex, stochastic sampling was applied, too, for computing a local minimizer [22].

The objective of this paper is to present a quite different variational approach to solving a similar constrained approximation problem. Rather than encoding the filter statistics as prior information by a probability distribution that has to be learned beforehand, we bypass this entire learning step and exploit the prior knowledge *directly* by imposing hard constraints[1] on the sizes of level-sets of filter outputs. Furthermore, we devise a *deterministic* algorithm that computes an approximation by solving a sequence of convex programs. The main purpose is the ability to study directly the formation of image structure through approximation in image spaces constrained by empirical distributions. Another aspect is that an *explicit* representation of the statistical knowledge is used for data processing, enabling to replace or to revise this knowledge quickly if an overall task is calling for, or if novel data are observed. This is not possible, however, when the available knowledge is *implicitly* represented by a probability distribution (1) defined over the entire image space, that has to be determined by a time-consuming prior learning process.

[1] Our approach also allows to weaken these constraints in order to cope with *uncertain* prior knowledge – see Remark 2.2. We do not exploit this modification in the present paper, however.

From the viewpoint of mathematical programming, our problem formulation involves a bilevel optimization problem which is interesting by its own in view of current research. This problem belongs to the class of mathematical programs with equilibrium constraints (MPECs) [14] that cannot be treated using methods developed for conventional nonlinear programs, because standard constraint qualifications fail at any feasible point [18]. Accordingly, a substantial amount of research has been devoted to cope with this difficulty by modifying established interior-point methods, see [18], [19], [11], [1], [15], [13] and references therein. In this work, we apply to our problem the interior-penalty approach investigated in [11], [13] together with a DC-programming[2] strategy [2] for solving a nonconvex relaxation of our approach. While we analyze the two inner loops of our variational method, that is the sequence of convex programs and its relationship to the nonconvex relaxation, an analysis of the consequences of the aforementioned DC-decomposition on the overall convergence, in view of the results in [11], [13], is beyond the scope of this paper.

*Organisation.* We introduce some notation and state formally the constrained image approximation problem in Sect. 2. In Sects. 3 and 4, we reformulate the problem as a MPEC and as a complementarity-constrained program, respectively, followed by the interior-penalty relaxation in Sect. 5. We apply a particular DC-decomposition to the penalty term and state our final algorithm. Properties of this algorithm with respect to both the convex inner loop and the nonconvex interior-penalty relaxation are stated in Sect. 6. Finally, we report details of our implementation in Sect. 7 and discuss some numerical examples for illustrating our approach.

*Notation.* Bars $|\cdot|$ applied to a finite set denotes the number of elements of the set. $\Omega$ denotes the domain of a given signal or image. We identify it with the sampling grid of size $n$, $|\Omega| = n$, corresponding to given signal or image measurements.

Vectors are column vectors, and $\cdot^\top$ transposes them. For simplicity, we often write $u = (v, w)$ for the column vector $u = (v^\top, w^\top)^\top$ obtained by concatenating vectors $v$ and $w$. $\langle v, w\rangle$ as well as $v^\top w$ denote the Euclidean inner product, and $\|\cdot\|$ the Euclidean norm. $e_n = (1, 1, \ldots, 1)^\top \in \mathbb{R}^n$ is the one-vector of dimension $n$, and $I$ the identity matrix whose dimension will be clear from the context. We use subscripts for enumerating vectors, e.g., $p_1$, $p_2$, and brackets for denoting components of vectors and matrices, e.g., $p_1(i)$, $D(i, j)$. The symbol $\otimes$ denotes the Kronecker product [9].

$I_C(v)$ denotes the indicator function of a closed convex set $C$, i.e., $I_C(v) = 0$ if $v \in C$, and $I_C(v) = \infty$ if $v \notin C$. We set $\mathbb{B} = [0, 1]$.

## 2. Problem formulation

We introduce some notation regarding filters and level-sets, followed by the definitions of two approximation problems.

[2] DC stands for "*d*ifference of *c*onvex functions" – see, e.g., [10]

### 2.1 Linear filters and level-sets

Let $\Omega = \{1, 2, \ldots, n\}$ be the sampling grid with positions indexed by $x \in \Omega$. For any vector $u \in \mathbb{R}^n$ defined on $\Omega$, that is for a given signal or for a vectorized image with values $u(x)\,,\ x \in \Omega$, we denote with

$$\Omega(z; u) := \left\{x \in \Omega \;\middle|\; u(x) \leq z\right\}, \tag{2}$$

the level-set of $u$ with respect to level $z \in \mathbb{R}$.

Furthermore, we assume to be given a set of FIR-filters $\mathcal{J}$, represented by matrices $H^j\,,\ j \in \mathcal{J}$, such that the linear mapping $H^j u$ corresponds to the convolved function $u(x)$. Typically, each $H^j$ is a circulant and sparse matrix, or block-circulant in the case of images. Level-sets of the *filtered* function $u$ are denoted with

$$\Omega^j(z; u) := \left\{x \in \Omega \;\middle|\; \left(H^j u\right)(x) \leq z\right\}. \tag{3}$$

For a given set $\mathcal{I}$ of levels $\{z_i\}_{i \in \mathcal{I}}$, we associate with each filter $j$ a subset $\mathcal{I}_j \subset \mathcal{I}$ of relevant levels $\{z_i\}_{i \in \mathcal{I}_j}$. Thus, all relevant level-sets are given by

$$\left\{\Omega^j(z_i; u)\right\}_{i \in \mathcal{I}_j,\, j \in \mathcal{J}}. \tag{4}$$

We also consider *intersections* of level-sets. To this end, let $\mathcal{K} = \{\mathcal{K}_1, \mathcal{K}_2, \ldots\}$ denote a family of subsets of index pairs $(i, j),\ i \in \mathcal{I},\ j \in \mathcal{J}$, that is each $\mathcal{K}_k \in \mathcal{K}$ has the form $\mathcal{K}_k \subseteq \mathcal{I} \times \mathcal{J}$. We require all elements of each $\mathcal{K}_k$ to differ from each other, but a filter $j$ or level $i$ may be part of several elements of a set $\mathcal{K}_k$. As a result, $\mathcal{K}$ represents the family of intersections of level-sets

$$\left\{\bigcap_{(i,j) \in \mathcal{K}_k} \Omega^j(z_i; u)\right\}_{\mathcal{K}_k \in \mathcal{K}}. \tag{5}$$

(4) may be regarded as a special of (5) with $|\mathcal{K}_k| = 1,\ \forall k$.

### 2.2 Level-set constrained approximation

We distinguish two problems according to whether constraints are imposed in terms of (4) or (5), respectively.

*Separate filter statistics.* Given a signal (or image) $d$, we wish to compute the closest signal (or image) $u$ subject to fixed sizes of the level-sets (4):

$$\min_{u \in \mathbb{R}^n} \frac{1}{2}\|u - d\|^2, \tag{6a}$$

$$\text{s.t.} \quad \left|\Omega^j(z_i; u)\right| = \omega_i^j, \quad \forall i \in I_j, \quad \forall j \in J. \tag{6b}$$

The numbers $\omega_i^j$ are assumed to be given. They represent the filter statistics of the signal (or image) class from which $d$ is observed, and are computed from samples of this class beforehand.

The constraints (6b) impose a difficult *combinatorial* problem: for each location $x$, the single value $u(x)$ decides simultaneously whether $x$ belongs to the level-sets $\Omega^j(z_i; u)$. Moreover, due to overlapping supports of the filters $H^j$, this decision interacts with those at adjacent positions in a local neighbourhood of $x$. Consequently, despite local quantities are only involved in (6), the overall problem is intrinsically non-local, akin to Markov random field inference. Unlike the latter, however, we determine real-valued functions $u : \Omega \to \mathbb{R}$ with a deterministic algorithm, and do not artificially quantize the range into discrete "labels". In this sense, problem (6) may be regarded as *hybrid* (continuous/discrete).

*Joint filter statistics.* We generalize problem (6) by fixing the size of *intersections* of various *collections* of level-sets (5):

$$\min_{u \in \mathbb{R}^n} \frac{1}{2}\|u - d\|^2, \tag{7a}$$

$$\text{s.t.} \quad \left| \bigcap_{(i,j) \in \mathcal{K}_k} \Omega^j(z_i; u) \right| = \omega_k, \quad \forall \mathcal{K}_k \in \mathcal{K}. \tag{7b}$$

As in the previous section, we assume that the numbers $\omega_k$ have been computed beforehand from examples of a particular image class, and are therefore given.

A significant difference between problems (6) and (7) is that the constraints (7b) represent *conjunctions*. At any location $x$, and for any level/filter pair $(i, j) \in \mathcal{K}_k$, only those values $u(x)$ contribute to satisfying the corresponding constraint in (7b), that make $x$ an element of *all* level-sets indexed by $\mathcal{K}_k$.

**Remark 2.1:** *Problems* (6) *and* (7) *involve a very simple objective function. It may be replaced by alternative functions, like the more robust* $\|\cdot\|_1$*-norm for instance, without essentially altering the overall basic problem structure as long as this function is convex.*

**Remark 2.2:** *Using lower and upper bounds for sizes of* (*intersections of*) *level-sets in* (6b) *and* (7b)*, respectively, may turn out to be more useful in applications, in order to capture the variability of a signal* (*or image*) *class. This requires minor modifications only of our approach to be developed below. In this work, however, we only focus on equality constraints.*

## 3. Equilibrium constraints

We reformulate (6) and (7) as hierarchical problems involving two convex optimization problems and represent the lower-level problem by an equivalent variational inequality. By this, we classify these problems as instances of mathematical programs with equilibrium constraints (MPECs) and prepare the development of a numerical algorithm in subsequent sections.

*Separate filter statistics.* In order to make more explicit the dependency of the constraints (7b) on $u$, we introduce for each index pair $(i, j)$ an auxiliary vector $v_{i,j} \in \mathbb{R}^n$ with components

$$v_{i,j}(x) := \begin{cases} 1 & \text{if } \left(H^j u\right)(x) < z_i, \\ \in [0,1] & \text{if } \left(H^j u\right)(x) = z_i, \\ 0 & \text{if } \left(H^j u\right)(x) > z_i. \end{cases} \tag{8}$$

Thus, the size of the corresponding level-set is

$$\langle e_n, v_{i,j} \rangle = \left|\Omega^j(z_i; u)\right|. \tag{9}$$

The mapping $u \to v_{i,j}$ defined in (8) is given by the *linear program* (*LP*)

$$v_{i,j} \in \operatorname*{argmin}_{\tilde{v} \in \mathbb{B}^n} \langle H^j u - z_i e_n, \tilde{v} \rangle. \tag{10}$$

In order to reformulate problem (6), we use the shorthand

$$H_{i,j}(u) := H^j u - z_i e_n \tag{11}$$

and the number of all level-sets

$$n_l := \sum_{j \in \mathcal{J}} |\mathcal{I}_j| \tag{12}$$

to define vectors

$$v \in \mathbb{R}^{n_l \cdot n}, \tag{13a}$$

$$H(u) := Hu - z \otimes e_n \in \mathbb{R}^{n_l \cdot n}, \tag{13b}$$

$$\omega \in \mathbb{R}^{n_l} \tag{13c}$$

by stacking together for all $i, j$ the vectors $v_{i,j}$, $H_{i,j}(u)$ defined in (8) and (11), respectively, and the numbers $\omega_i^j$ on the right-hand side of (6). Furthermore, we define the matrix

$$E_v := \begin{pmatrix} e_n^\top & 0^\top & \dots & 0^\top \\ 0^\top & e_n^\top & \dots & 0^\top \\ \vdots & \vdots & \ddots & \vdots \\ 0^\top & 0^\top & \dots & e_n^\top \end{pmatrix} \in \mathbb{R}^{n_l \times (n_l \cdot n)} \tag{14}$$

and denote with

$$v \in \mathrm{SOL}\left(\mathbb{B}^{n_l \cdot n}, H(u)\right) \tag{15}$$

the fact that $v$ satisfies the variational inequality

$$\langle H(u), \tilde{v} - v \rangle \geq 0, \qquad \forall \tilde{v} \in \mathbb{B}^{n_l \cdot n}, \tag{16}$$

that is equivalent to (10) holding for all indices $i, j$.

Using this notation, problem (6) reads

$$\min_{u,v} \frac{1}{2}\|u-d\|^2, \tag{17a}$$

$$\text{s.t.} \quad E_v v = \omega, \tag{17b}$$

$$v \in \mathrm{SOL}\left(\mathbb{B}^{n_l \cdot n}, H(u)\right). \tag{17c}$$

This formulation shows that problem (6) is an instance of a MPEC [14]. It also provides the basis for the design of numerical algorithms for computing $u$.

*Joint filter statistics.* Our objective is to make explicit the dependency of the constraints (7b) on $u$. To this end, we note that pixel location $x = l$ belongs to the intersection of the level-sets (7b) – recall the notation (11)) – if

$$\max\left\{H_{i,j}(u)\right\}_{(i,j)\in\mathcal{K}_k} \leq 0. \tag{18}$$

To extract this information from $u$, we define local vectors

$$h_{k,l}(u) := \left\{\left(H_{i,j}(u)\right)(l)\right\}_{(i,j)\in\mathcal{K}_k} \in \mathbb{R}^{|\mathcal{K}_k|} \tag{19}$$

that collect the $l$th components of all vectors $H_{i,j}(u)$, $(i,j) \in \mathcal{K}_k$, and compute the left-hand side of (18) as solution to the LP

$$\min_{\mathbb{R}} \lambda_k(l), \quad \lambda_k(l)\, e_{|\mathcal{K}_k|} - h_{k,l}(u) \geq 0. \tag{20}$$

Concatenating all numbers $\lambda_k(l)$ and vectors $h_{k,l}(u)$ for $l = 1, \ldots, n$, to obtain vectors $\lambda_k \in \mathbb{R}^n$, $h_k(u) \in \mathbb{R}^{n|\mathcal{K}_k|}$, the $k$-th constraint (7b) can be expressed as

$$\langle e_n, v_k\rangle = \omega_k, \tag{21}$$

where

$$v_k \in \operatorname*{argmin}_{\tilde{v}\in\mathbb{B}^n} \langle \lambda_k, \tilde{v}\rangle \tag{22}$$

and every component of $\lambda_k$ solves (20). Concatenating in turn for $k = 1, \ldots, |\mathcal{K}|$ the numbers and vectors $\omega_k, v_k, \lambda_k, h_k(u)$, resulting in

$$\omega \in \mathbb{R}^{|\mathcal{K}|}, \quad v \in \mathbb{R}^{n|\mathcal{K}|}, \quad \lambda \in \mathbb{R}^{n|\mathcal{K}|}, \quad h(u) \in \mathbb{R}^{n\sum_k |\mathcal{K}_k|}, \tag{23}$$

and defining

$$\Lambda(u) := \left\{\lambda \in \mathbb{R}^{n|\mathcal{K}|} \,\middle|\, \lambda_k(l) \text{ satisfies (20)},\ l = 1, \ldots, n,\ k = 1, \ldots, |\mathcal{K}|\right\}, \tag{24}$$

the reformulation of problem (7) reads

$$\min_{u,v,\lambda} \frac{1}{2}\|u-d\|^2, \tag{25a}$$

$$\text{s.t.} \quad E_v v = \omega, \tag{25b}$$

$$\lambda \in \Lambda(u), \tag{25c}$$

$$v \in \mathrm{SOL}\left(\mathbb{B}^{n|\mathcal{K}|}, \lambda\right). \tag{25d}$$

Comparing (25) with (17) reveals that the additional intersections in (7) relative to (6) do not essentially alter the structure of the MPEC (17).

## 4. Complementarity-constrained programming

In this section, we exploit the convexity of the lower level problems in (17) and (25) to represent them by complementarity constraints. This provides the basis for relaxing the combinatorial part of the overall problem and, in turn, will directly lead to the design of a numerical algorithm in the subsequent section.

*Separate filter statistics.* Introducing a dual (multiplier) vector $w \in \mathbb{R}^{n_l \cdot n}$, we consider the KKT-conditions of the lower-level convex program (17c) and transform further problem (17):

$$\min_{u,v,w} \frac{1}{2}\|u - d\|^2, \tag{26a}$$

$$\text{s.t.} \qquad E_v v = \omega, \tag{26b}$$

$$0 \leq \begin{pmatrix} v \\ w \end{pmatrix} \perp \begin{pmatrix} H(u) + w \\ e_{n_l \cdot n} - v \end{pmatrix} \geq 0. \tag{26c}$$

The lower-level problem (17c) now takes the form of a complementarity constraint, i.e., an orthogonality constraint of two vectors in the positive cone. This constraint represents the combinatorial part of the overall problem. Rewriting it in the more common form

$$0 \leq \begin{pmatrix} v \\ w \end{pmatrix} \perp \begin{pmatrix} 0 & I \\ -I & 0 \end{pmatrix} \begin{pmatrix} v \\ w \end{pmatrix} + \begin{pmatrix} H(u) \\ e_{n_l \cdot n} \end{pmatrix} \geq 0 \tag{27}$$

reveals a linear complementarity problem (LCP) (see [6]) parametrized by the upper level variable $u$.

*Joint filter statistics.* In order to generalize formulation (26) to problem (25), we additionally consider the dual programs to (20)

$$\max_{\mathbb{R}^{|\mathcal{K}_k|}} \left\langle \mu_{k,l}, h_{k,l}(u) \right\rangle, \quad \text{s.t.} \quad \langle \mu_{k,l}, e_{|\mathcal{K}_k|} \rangle = 1, \quad \mu_{k,l} \geq 0. \tag{28}$$

By concatenating the local vectors $\mu_{k,l}$ as in the previous section, we obtain

$$\mu \in \mathbb{R}^{n \sum_k |\mathcal{K}_k|}. \tag{29}$$

Accordingly, the equality constraints in (28) define the system

$$E_\mu \mu = e_{n|\mathcal{K}|}, \quad E_\mu \in \mathbb{R}^{(n|\mathcal{K}|) \times (n \sum_k |\mathcal{K}_k|)}, \tag{30}$$

whereas the inequalities in (20) define the system

$$E_\lambda \lambda - h(u) \geq 0, \quad E_\lambda \in \mathbb{R}^{(n \sum_k |\mathcal{K}_k|) \times (n|\mathcal{K}|)}. \tag{31}$$

Using this notation, problem (25) becomes

$$\min_{u,v,w,\mu,\lambda} \frac{1}{2}\|u-d\|^2, \tag{32a}$$

$$\text{s.t.} \qquad E_v v = \omega, \tag{32b}$$

$$E_\mu \mu = e_{n|\mathcal{K}|}, \tag{32c}$$

$$0 \le \begin{pmatrix} v \\ w \\ \mu \end{pmatrix} \perp \begin{pmatrix} \lambda + w \\ e_{n|\mathcal{K}|} - v \\ E_\lambda \lambda - h(u) \end{pmatrix} \ge 0. \tag{32d}$$

## 5. Relaxation and optimization

Like the original level-set constraints (6b) and (7b), the complementarity constraints (26c) and (32d) are still combinatorial and therefore difficult. A range of smoothing and relaxation techniques have been proposed in the literature in order to weaken such constraints and to cope with them numerically – see, e.g., [18], [1], [15], and [13] and references therein.

In the following, we adopt an interior-penalty approach [11], [13], encouraged by the numerical results reported by Leyffer et al. [13]. We additionally propose in the subsequent section a decomposition of the corresponding penalty term into the difference of two convex functions, enabling us to conveniently compute a minimizer by a sequence of convex programs.

### *5.1 Interior-penalty relaxation*

The basic idea worked out in this section is to weakly incorporate the complementarity constraints by complementing the constraints with slack variables, and by adding a penalty term to the objective function. We distinguish again the two problems (6) and (7).

*Separate filter statistics.* We include additional slack variables

$$s \in \mathbb{R}^{n_l \cdot n}, \quad t \in \mathbb{R}^{n_l \cdot n}, \tag{33}$$

and, using (13), propose the following relaxation of problems (6), (17) and (26), respectively:

$$\min_{u,v,w,s,t} \left\{ \frac{1}{2}\|u-d\|^2 + \pi\left(\langle v, s\rangle + \langle w, t\rangle\right) \right\}, \qquad \mathbb{R} \ni \pi > 0, \tag{34a}$$

$$\text{s.t.} \qquad E_v v = \omega, \tag{34b}$$

$$Hu + w - s = z \otimes e_n, \tag{34c}$$

$$v + t = e_{n_l \cdot n}, \tag{34d}$$

$$v, w, s, t \ge 0. \tag{34e}$$

Note that all constraints are convex. The penalty term only, weighted by a parameter $\pi$, is nonconvex. It will give rise to a sequential convex approximation technique introduced in the subsequent section.

*Joint filter statistics.* Regarding problems (7), (25) and (32), respectively, we define similarly

$$s \in \mathbb{R}^{n|\mathcal{K}|}, \quad t \in \mathbb{R}^{n|\mathcal{K}|}, \quad \nu \in \mathbb{R}^{n\sum_k |\mathcal{K}_k|}, \tag{35}$$

and the matrix $S$

$$h(u) =: SH(u) \tag{36}$$

mapping the vector $H(u)$ defined by (13b), to the vector $h(u)$ defined by (23), as specified by Eq. (19).

The relaxed problem then reads:

$$\min_{u,v,w,\mu,s,t,\nu,\lambda} \left\{ \frac{1}{2}\|u-d\|^2 + \pi\left(\langle v, s\rangle + \langle w, t\rangle + \langle \mu, \nu\rangle\right)\right\}, \quad \mathbb{R} \ni \pi > 0, \tag{37a}$$

$$\text{s.t.} \qquad E_v v = \omega, \tag{37b}$$

$$E_\mu \mu = e_{n|\mathcal{K}|}, \tag{37c}$$

$$\lambda + w - s = 0, \tag{37d}$$

$$v + t = e_{n|\mathcal{K}|}, \tag{37e}$$

$$E_\lambda \lambda - SHu - \nu = -S(z \otimes e_n), \tag{37f}$$

$$v, w, \mu, s, t, \nu \geq 0. \tag{37g}$$

Again this problem is convex up to the penalty term that is weighted by a parameter $\pi$.

### *5.2 DC-Decomposition and sequential convex optimization*

In the following, we decompose the inner product between two vectors $p_1$, $p_2$ into the difference of two convex functions

$$\langle p_1, p_2\rangle = \frac{1}{2}\begin{pmatrix} p_1 \\ p_2 \end{pmatrix}^\top \begin{pmatrix} 0 & I \\ I & 0 \end{pmatrix} \begin{pmatrix} p_1 \\ p_2 \end{pmatrix} \tag{38a}$$

$$= \frac{1}{4}\begin{pmatrix} p_1 \\ p_2 \end{pmatrix}^\top \left[ \underbrace{\begin{pmatrix} c_\pi I & I \\ I & c_\pi I \end{pmatrix}}_{:=P^+} - \underbrace{\begin{pmatrix} c_\pi I & -I \\ -I & c_\pi I \end{pmatrix}}_{:=P^-} \right] \begin{pmatrix} p_1 \\ p_2 \end{pmatrix}, \quad \mathbb{R} \ni c_\pi > 1. \tag{38b}$$

**Remark 5.1:** *Both matrices $P^+$ and $P^-$ are positive definite.*

We apply this decomposition together with a linearization of the second concave term defined in terms of $P^-$, to the nonconvex penalty terms in (34) and (37), respectively. Our approach is a problem-specific instance of the general DC-programming approach developed by An and Tao [2], [21].

*Separate filter statistics.* Setting

$$p = \begin{pmatrix} p_1 \\ p_2 \end{pmatrix}, \quad p_1 = \begin{pmatrix} v \\ w \end{pmatrix}, \quad p_2 = \begin{pmatrix} s \\ t \end{pmatrix}, \tag{39}$$

we propose to minimize (34) by the sequence of *convex* programs

$$\left(u^{k+1}, p^{k+1}\right) \in \operatorname*{argmin}_{u,p} \left\{ \frac{1}{2}\|u - d\|^2 + \frac{\pi}{4}\left\langle p, P^+ p - 2P^- p^k \right\rangle \right\}, \tag{40a}$$

$$\text{s.t.} \qquad A \begin{pmatrix} u \\ p \end{pmatrix} = b, \quad p \geq 0. \tag{40b}$$

The linear system (40b) is given by

$$A = \begin{pmatrix} 0 & E_v & 0 & 0 & 0 \\ H & 0 & I & -I & 0 \\ 0 & I & 0 & 0 & I \end{pmatrix}, \qquad b = \begin{pmatrix} \omega \\ z \otimes e_n \\ e_{n_l \cdot n} \end{pmatrix}. \tag{41}$$

*Joint filter statistics.* Setting

$$p = \begin{pmatrix} p_1 \\ p_2 \end{pmatrix}, \quad p_1 = \begin{pmatrix} v \\ w \\ \mu \end{pmatrix}, \quad p_2 = \begin{pmatrix} s \\ t \\ \nu \end{pmatrix}, \tag{42}$$

we propose to minimize (37) by the sequence of *convex* programs

$$\left(u^{k+1}, p^{k+1}, \lambda^{k+1}\right) \in \operatorname*{argmin}_{u,p,\lambda} \left\{ \frac{1}{2}\|u - d\|^2 + \frac{\pi}{4}\left\langle p, P^+ p - 2P^- p^k \right\rangle \right\}, \tag{43a}$$

$$\text{s.t.} \qquad A \begin{pmatrix} u \\ p \\ \lambda \end{pmatrix} = b\,, \quad p \geq 0. \tag{43b}$$

The linear system (43b) is given by

$$A = \begin{pmatrix} 0 & E_v & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & E_\mu & 0 & 0 & 0 & 0 \\ 0 & 0 & I & 0 & -I & 0 & 0 & I \\ 0 & I & 0 & 0 & 0 & I & 0 & 0 \\ -SH & 0 & 0 & 0 & 0 & 0 & -I & E_\lambda \end{pmatrix}, \qquad b = \begin{pmatrix} \omega \\ e_{n|\mathcal{K}|} \\ 0 \\ e_{n|\mathcal{K}|} \\ -S(z \otimes e_n) \end{pmatrix}. \tag{44}$$

Properties of (40) and (43) as well as details of our implementation will be reported in the following two sections.

**Remark 5.2:** *Note that $\lambda$ is missing in the objective function* (43a)*. This slightly complicates our analysis and numerical implementation presented below.*

## 6. Properties

In the next two sections, we analyse the sequential convex approximations (40b) and (43b) of the nonconvex programs (34) and (37). Next we comment on the solvability of the original problems (26) and (32) by the nonconvex relaxations (34) and (37), for an increasing sequence of the penalty parameter.

### *6.1 Global optimality and uniqueness (inner loop)*

**Lemma 6.1:** *The constraint systems* (40b) *and* (43b) *satisfy the Mangasarian Fromovitz constraint Qualification (MFCQ) at every feasible point.*

*Proof:* The rows of the matrix $A$ in (41) are linearly independent. Furthermore, inspection of the constraints in (34) shows that $v \neq 0$, and that $v(i)$ and $t(i)$ cannot both vanish for any $i$. Thus, for any feasible $(u, p)$ (recall the definition (39) of $p$), there exists a vector $q = (q_u, q_p)$ with $Aq = 0$, and with $q_p(i) > 0$ if $p(i) = 0$. The same reasoning applies to the matrix $A$ in (44) and to the constraints in (37), for any feasible point $(u, p, \lambda)$. □

**Proposition 6.2:** *Each of the convex programs* (40) *and* (43) *has a unique global minimum at each iteration step k.*

*Proof:* The assertion is immediate in the case of (40) because the Hessian

$$\begin{pmatrix} I & 0 \\ 0 & \frac{\pi}{2} P^+ \end{pmatrix}$$

is positive definite (cf. Remark 5.1). Regarding (43) (cf. Remark 5.2), let $f$ denote the objective function (43a), $L$ the corresponding Lagrangian, $C$ the feasible set, $y = (u, p, \lambda)$ a stationary point, $\mathcal{T}(y; C)$ the tangent cone at $y$, and $\mathcal{C}(y; C, \nabla f) = \mathcal{T}(y; C) \cap \{\nabla f(y)\}^{\perp}$ the critical cone at $y$. If the Hessian $\nabla^2_{yy} L$ of the Lagrangian with respect to the primal variables is strictly copositive on $\mathcal{C}$, then Lemma 6.1 and [7, corollary 3.3.20] imply that $y$ is an isolated, strong local minimizer. The assertion then follows from the convexity of (43).

$\nabla^2_{yy} L$ is strictly copositive iff $q = (q_u, q_p, q_\lambda) = 0$ is the unique global minimum of

$$\min_{q \in \mathcal{C}} \frac{1}{2} \langle q, \nabla^2_{yy} L\, q \rangle.$$

The expression

$$\left\langle q, \nabla^2_{yy} L\, q \right\rangle = \|q_u\|^2 + \frac{\pi}{2} \langle q_p, P^+ q_p \rangle$$

and Remark 5.1 show that for any minimizer $q_u = 0$ and $q_p = 0$. Consequently, because $q \in \mathcal{C}$ implies $q \in \mathcal{T}$ and in turn $Aq = 0$, equation (37d) and $q_p = 0$ show that $q_\lambda = 0$. □

### 6.2 Convergence and local optimality (outer loop)

Let $C$ denote the feasible set of (34). Using (38) and (39), we write (34) as DC-program

$$\min_{u,p} \{g(u, p) - h(u, p)\}, \tag{45}$$

with convex proper and lower-semicontinuous functions

$$g(u, p) = \frac{1}{2}\|u - d\|^2 + \frac{\pi}{4}\langle p, P^+ p\rangle + I_C(p), \tag{46}$$

$$h(u, p) = \frac{\pi}{4}\langle p, P^- p\rangle. \tag{47}$$

The iteration (40) corresponds to the two-step process

$$(0, q_h^k) \in \partial h(u^k, p^k) \quad (u^{k+1}, p^{k+1}) \in \partial g^*(0, q_h^k) \tag{48a}$$

$$= \{\nabla h(u^k, p^k)\} = \operatorname*{argmin}_{u,p} \left\{g(u, p) - \langle q_h^k, p\rangle\right\}, \tag{48b}$$

where $g^*$ is the conjugate function to $g$, and $\partial h(u^k, p^k)$ and $\partial g^*(0, q_h^k)$ denote the subgradients of $h$ and $g^*$ at their arguments, respectively, (cf. [16, chap. 11]). This iterative process is obviously well-defined for any feasible point $(u, p)$.

The formulation (48) is a special instance of a general algorithm ("simplified DCA") developed and investigated in [21] and [2].

**Proposition 6.3:** *The sequence* $(u^k, p^k)$ *defined by* (48) *converges, and the limit point* $(\overline{u}, \overline{p})$ *satifies the necessary local optimality conditions with respect to problem* (45)

$$\partial h(\overline{u}, \overline{p}) \subset \partial g(\overline{u}, \overline{p}) \ . \tag{49}$$

*Proof:* In [21] it is shown that the sequence $g(u^k, p^k) - h(u^k, p^k)$ is decreasing. Because the objective function is finite and the sequence $(u^k, p^k)$ is bounded, [21, theorem 3.7] implies convergence to a critical point $(\overline{u}, \overline{p})$ defined by

$$\partial g(\overline{u}, \overline{p}) \cap \partial h(\overline{u}, \overline{p}) \neq \emptyset \ .$$

The assertion follows from $\partial h(\overline{u}, \overline{p}) = \{\nabla h(\overline{u}, \overline{p})\}$ and $\nabla h(\overline{u}, \overline{p}) \in \partial g(\overline{u}, \overline{p}) \Leftrightarrow (\overline{u}, \overline{p}) \in \partial g^* (\nabla h(\overline{u}, \overline{p}))$ by [16, proposition 11.3]. □

The same reasoning applies to the sequence $(u^k, p^k, \lambda^k)$ generated by (43), taking into account the proof of Proposition 6.2 regarding boundedness.

We summarize the result:

**Corollary 6.4:** *The sequences of global optima of the convex programs* (40) *and* (43) *converge, and the limit points satisfy the necessary local optimality conditions with respect to the nonconvex problems* (34) *and* (37).

### 6.3 Stationarity (MPEC)

In our experiments, we wish to solve the approximation problems (6) and (7) through sequences of minimizers of the *nonconvex* relaxations (34) and (37), defined by an increasing sequence $\pi^l$ of the penalty parameter. General studies of the convergence of penalty methods to various classes of stationary points of MPECs [18] include the work of Hu and Ralph [11] and Leyffer et al. [13]. As mentioned at the end of Sect. 1, a corresponding analysis of our approach is beyond the scope of this paper and will be presented elsewhere.

We confine ourselves to pointing out an assumption that is most crucial (cf. [13, theorem 3.4]): The penalty term $\langle p_1, p_2 \rangle$ must converge to zero and exhibit a certain behaviour relative to the sequence $\pi^l$ of the penalty parameter.

In view of these assumptions, we conduct our experiments with a *finite* increasing sequence $\pi^l$ and conclude "success" of the approximation if the penalty term vanishes, and "failure" otherwise.

## 7. Implementation and numerical examples

We first report some details of our implementation regarding (40) and (43). Next, we illustrate the approaches (6) and (7) by a few numerical examples.

### 7.1 Implementation details

*Interior point method.* At each iteration step $k$ in (40) or (43), we solve the convex program using the standard logarithmic barrier function for the positivity constraints, and by applying a feasible Newton method to the resulting equality-constrained convex problem [4]. In the case of (43), the barrier problem with parameter $\tau$ reads

$$\left\{u^k, p^{k+1}, \lambda^{k+1}\right\} \in \operatorname*{argmin}_{u,p,\lambda} \left\{\tau \left(\frac{1}{2}\|u-d\|^2 + \frac{\pi}{4}\left\langle p, P^+p - 2P^-p^k\right\rangle\right) - \sum_i \log p(i)\right\}, \tag{50a}$$

$$\text{s.t.} \qquad A\begin{pmatrix} u \\ p \\ \lambda \end{pmatrix} = b. \tag{50b}$$

*Computing a descent direction.* A key step of this approach is to determine a feasible search direction by solving the KKT-system associated with the linearized optimality conditions for (50). We detail next how to do this efficiently, and how to cope with the (minor) complication pointed out in Remark 5.2.

To this end, we use temporarily the notation $y = (u, p)$ and $f(y)$ for the objective function (50a), and – omitting the iteration index $k$ – rewrite (50):

$$\min_{y,\lambda} f(y)\,, \quad \text{s.t.} \quad \begin{pmatrix} A_y & A_\lambda \end{pmatrix}\begin{pmatrix} y \\ \lambda \end{pmatrix} = b. \tag{51}$$

Let $\zeta$ denote the dual vector corresponding to the constraints (50b). Then the KKT-system for computing a descent direction $(\Delta y, \Delta\lambda)$ reads

$$\begin{pmatrix} \nabla^2 f(y) & 0 & A_y^\top \\ 0 & 0 & A_\lambda^\top \\ A_y & A_\lambda & 0 \end{pmatrix} \begin{pmatrix} \Delta y \\ \Delta\lambda \\ \zeta \end{pmatrix} = \begin{pmatrix} -\nabla f(y) \\ 0 \\ 0 \end{pmatrix}. \tag{52}$$

The computation proceeds by eliminating $(\Delta y, \Delta\lambda)$ and substitution to solve for $\zeta$, which in turn gives $(\Delta y, \Delta\lambda)$. The elimination step is slightly complicated by the singularity of the upper-left $2 \times 2$ block-matrix in (52). We just focus on this problem here and refer to [4] for the remaining details of the algorithm that we applied without further modifications.

**Proposition 7.1:** *The variables* $(\Delta y, \Delta\lambda)$ *in* (52) *can be eliminated by applying to the corresponding components on the right-hand side the matrix*

$$\begin{pmatrix} \tau^{-1}I & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & Q_{1,1} & 0 & 0 & Q_{2,1} & 0 & 0 & Q_{2,1} \\ 0 & 0 & Q_{1,2} & 0 & 0 & Q_{2,2} & 0 & -Q_{1,2} \\ 0 & 0 & 0 & Q_{1,3} & 0 & 0 & Q_{2,3} & 0 \\ 0 & Q_{2,1} & 0 & 0 & Q_{3,1} & 0 & 0 & Q_{3,1} \\ 0 & 0 & Q_{2,2} & 0 & 0 & Q_{3,2} & 0 & -Q_{2,2} \\ 0 & 0 & 0 & Q_{2,3} & 0 & 0 & Q_{3,3} & 0 \\ 0 & Q_{2,1} & -Q_{1,2} & 0 & Q_{3,1} & -Q_{2,2} & 0 & I + Q_{1,2} + Q_{3,1} \end{pmatrix},$$

*where the three diagonal matrices* $Q_i = \operatorname{diag}(Q_{i,1}, Q_{i,2}, Q_{i,3})$, $i = 1, 2, 3$, *are given by* (59) *and* (57).

*Proof:* See Appendix. □

*Parameter values.* We normalize the data $d$ to the range $d(x) \in [0, 1], \forall x \in \Omega$. Then we choose $c_\pi = 1.5$ in (38) and start the iterations (40) or (43) with an iteration parameter $l = 0$ and $u^0 = d$, $\tau^0 = 1$, $\pi^0 = 1$. After convergence of the sequence of convex programs, the barrier parameter $\tau$ and the complementarity parameter are updated: $\tau^{l+1} = 2\tau^l$, $\pi^l = 1 + 9(l/20)^5$. Next we increase the iteration counter $l$ by one and continue to solve the sequence of convex programs. We terminate this process at $l = 20$.

The increasing weight $\pi$ of the penalty term (see Fig. 2), together with the barrier parameter, gradually enforces the complementarity constraints in (26) and (32), respectively. We check this by monitoring the value of the penalty term. Figure 3 shows a typical behaviour of the penalty term that decreases during the iteration to a small value close to zero. This in turn confirms that the level-set constraints in (6) and (7) are satisfied.

### *7.2 Numerical examples*

The objective of this section is to illustrate by numerical examples some key properties of the approaches (6) and (7).

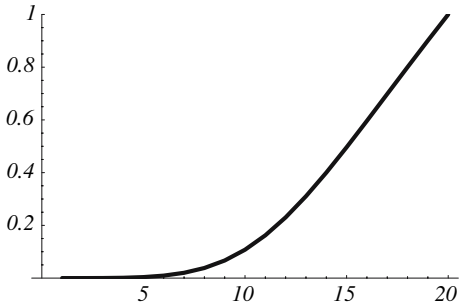


**Fig. 2.** Schedule $\log_{10}(\pi^l)$ of the complementarity penalty parameter $\pi$ for enforcing the level-set constraints in 20 iteration steps of the outer loop

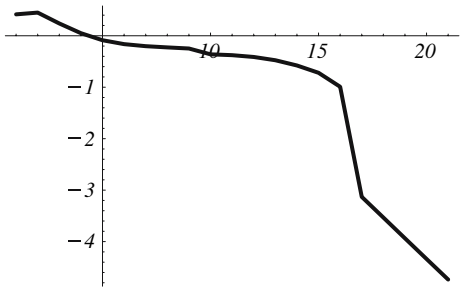


**Fig. 3.** Typical behaviour of the penalty term $\log_{10}(\max_i\{p_1(i)p_2(i)\})$ during the iteration. Its final value in the experiments is $\approx 10^{-6}$

Although some of the experiments discussed below required to solve MPECs involving about 20.000 variables which is more than tenfold the problem sizes considered in [13], our current research implementation does not allow for processing whole images. To this end, the sparse structure of the constraint matrix $A$ has to be exploited during the Newton search steps, e.g. for computing $(A^\top H^{-1}A)^{-1}$ with $H^{-1}$ given by Proposition 7.1. Thus, the main focus of the discussion below is on the validation of our variational approach.

*Separate filter statistics.* Figure 4 shows two experiments for illustrating the *combinatorial* aspect of the approximation problem. We wish to compute the closest point $u$ to a given arbitrary noise signal $d$ subject to the constraint that the filtered solution $Hu$ vanishes at all but eight locations $x$. The *discrete* decisions *where* these optimal locations are, is *simultaneously* made together with approximating $d$ by $u$. The only difference between the two experiments is the respective filter $H$, which was the simplest 2-tap derivative filter and the 1D-Laplacian, respectively. As a consequence, we compute a piecewise constant approximation in the first case and a piecewise linear approximation in the second. The bottom panels show the decisions made by the algorithm in terms of $Hu$.

The reader might correctly point out that the previous experiment can be easily solved using dynamic programming. This, however, is no longer possible (with a

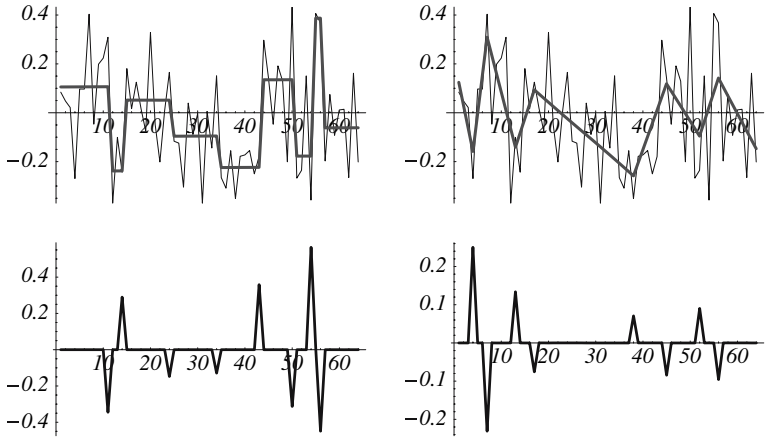


**Fig. 4.** (*Top*) A piecewise constant and a piecewise linear approximation of an arbitrary (noise) signal, respectively, computed by using two level-set constraints that, in each case, enforce $Hu$ to vanish at all but eight locations. $H$ was the 1D-gradient filter for the experiment shown on the left, and the 1D-Laplacian on the right. (*Bottom*) The corresponding filtered solutions $Hu$. The locations with nonzero values are automatically determined as part of the approximation

reasonable complexity) for *two*- and *higher*-dimensional signals. Figure 5 shows two experiments using three primitive filters and six level-set constraints as described in the figure caption. The results demonstrate the ability to restore *non-smooth* signals and, related to that, to *generate* image structure consistent with both the constraints and the input data. This latter fact is remarkable in view that the algorithm merely solves a sequence of *convex* programs.

*Joint filter statistics.* We present two further numerical examples illustrating the approach (7) involving *intersecting* level-set constraints.

Figure 6, top-left, shows a signal composed of two structures. Filtering this signal with a 1D-Laplacian characterizes these structures locally by negative and positive values corresponding to convex and concave parts (Fig. 6, top-right). Furthermore, the levels $z_1 = 0$ and $z_2 = 1$ of the signal *coincide* with the levels $z_3 = -1/2$ and $z_4 = +1/2$ of the filtered signal.

Consequently, adopting the notation introduced in Sect. 2.1, we define three levels

$$\mathcal{I} = \{1, 2, 3\}, \quad z_i \in \{-1, -1/2, 0\}, \; i \in \mathcal{I},$$

and four filters, the signed identity filters and Laplacians[3],

$$\mathcal{J} = \{1, 2, 3, 4\}, \quad H^1 = I, \quad H^2 = L, \quad H^3 = -I, \quad H^4 = -L,$$

[3] *Signed* filters are used to express constraints of the form $\left[\left(H^j u\right)(x) \geq z_i\right]$ as $\left[\left(-H^j u\right)(x) \leq -z_i\right]$ (cf. Definition (3)).

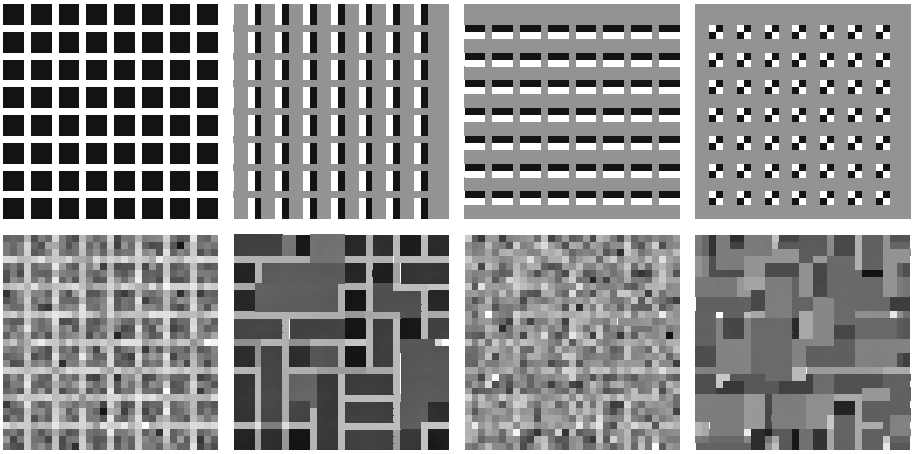

**Fig. 5.** (*Top, from left to right*) An image and its partial derivatives $\partial_{x_1}$, $\partial_{x_2}$, and $\partial^2_{x_1 x_2}$. The sizes of level-set of positive, zero and negative values, respectively, i.e., the number of *white*, *gray* and *black* pixels (not the values!), were imposed separately for the three filters as level-set constraints. (*Bottom, two figures on the left*) A noisy input image (*left*) and its constrained approximation. This result shows that level-set constrained approximation enables *non-smooth* restorations. (*Bottom, two figures on the right*) A very noisy image and its constrained approximation. This result shows that while the constraints do not suffice to restore the grid, the level-set constrained approximation *generates* image structure compatible with both the constraints and the input data

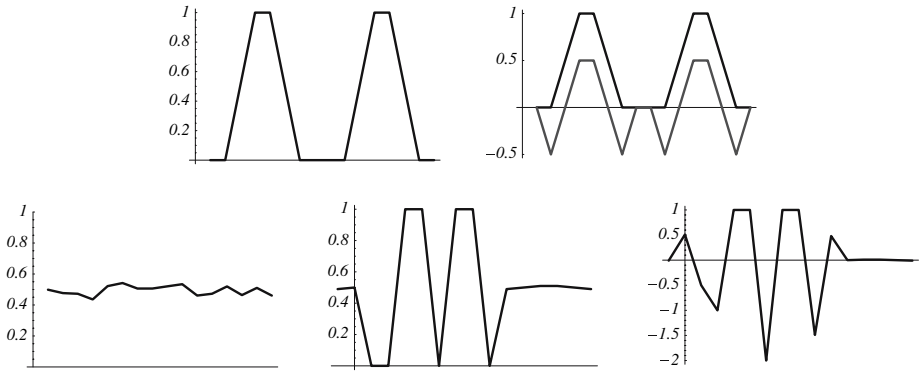


**Fig. 6.** (*Top-left*) A simple signal with two local structures. (*Top-right*) The signal shown left (*black curve*) together with its 1D-Laplacian filtered version (*blue curve*) indicating the local signal structure (convexities, concavities) through positive and negative values. (*Bottom-left*) An arbitrary noisy input signal $d$. (*Bottom-center*) The approximation $u$ of the noisy signal subject to *conjunctions* of level-set constraints as specified in the text. Based on the random input $d$, its approximation $u$ exhibits the local structure encoded by the constraints at random locations. (*Bottom-right*) The Laplacian-filtered approximation $Lu$. This result illustrates that intersecting level-set constraints (joint filter statistics) provide tight restrictions of the signal space and lead to the *formation* of local signal structure

as well as constraints with respect to the intersection of level-sets of signals filtered with $H^j$, $j \in \mathcal{J}$,

$$\begin{aligned}\mathcal{K} &= \{\mathcal{K}_1, \mathcal{K}_2\},\\ &= \{\{(3,1),(2,2)\}, \{(1,3),(2,4)\}\}.\end{aligned}$$

Rewriting the constraints above explicitly as conjunctions, we obtain for (7b)[4]

$$\left|\left\{x \in \Omega \,\middle|\, (u(x) \leq 0) \;\wedge\; \left((Lu)(x) \leq -\frac{1}{2}\right)\right\}\right| = \omega_1, \tag{53a}$$

$$\left|\left\{x \in \Omega \,\middle|\, (u(x) \geq 1) \;\wedge\; \left((Lu)(x) \geq +\frac{1}{2}\right)\right\}\right| = \omega_2, \tag{53b}$$

with $\omega$ computed from the original signal (Fig. 6, top-left). We include also the constraint that the gradient of the approximation $u$ vanishes within an *unknown* subset having a specified size. Thus, temporarily adopting the symbol $G$ for the gradient filter, we additionally impose

$$\left|\left\{x \in \Omega \,\middle|\, (Gu)(x) \leq -0.01\right\}\right| = \omega_3\,, \quad \left|\left\{x \in \Omega \,\middle|\, (Gu)(x) \geq 0.01\right\}\right| = \omega_4, \tag{54}$$

by enlarging $\mathcal{I}$, $\mathcal{J}$, $\mathcal{K}$ accordingly.

The lower panel of Fig. 6 illustrates how tightly these constraints restrict the entire signal space. On the left, Fig. 6 depicts an arbitrary input signal $d$. The approximation $u$ subject to the constraints detailed above is shown in the center. This result illustrates how the local signal structure is *enforced* by jointly constraining the level-sets of two filters, *and localized* through approximating the input signal. The figure on the right shows $Lu$ satisfying exactly the intersecting level-set constraints. Note that outside the intersection set, that is within $\{x \in \Omega \,|\, 0 < u < 1\}$, the filtered signal $Lu$ is *not* constrained and thus may admit additional local signal structure, depending on other constraints being active at such locations. In our example, this occurs at the positive peaks close to the boundary.

Figure 7, top, shows another signal comprising the local structure from the previous example (Fig. 6) at random locations, and a corrupted input signal generated by adding noise. The lower panel shows on the left the reconstruction using the same constraints as for the previous example, with $\omega$ changed accordingly. While the local structures are restored, the remaining degrees of freedom discussed above also lead to undesired local structures that are not present in the original signal. The result on the right shows that this additional structure can be avoided by specifying the size of an unknown subset where the gradient of $u$ vanishes *and* $u(x) = 0$. With other words, we replace the constraint (54) with an *intersecting* level-set constraint analogous to (53).

The key observation to be made here is that the approach (7) allows to approximate (or restore) signals based on knowledge about local signal *structure*, rather than just smoothness. This knowledge is encoded through filter statistics without the need to specify any locations beforehand, and expressed in terms of sizes of intersections of level-sets. It is the constrained approximation process only that decides *where* this knowledge applies best to given data.

[4] The symbol $\wedge$ means "and".

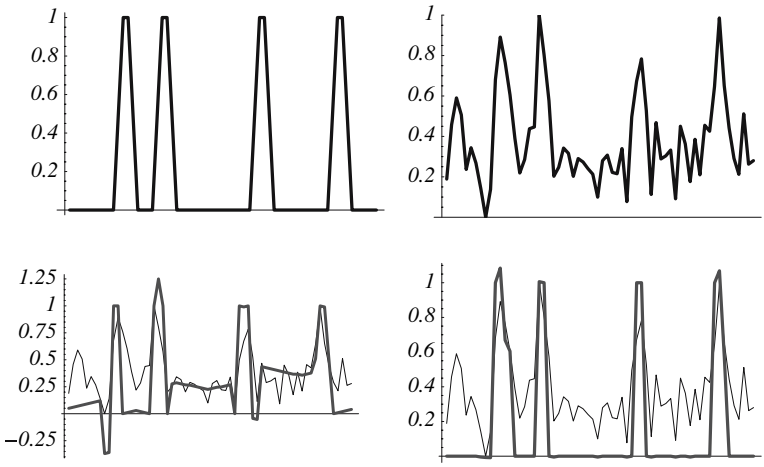


**Fig. 7.** (*Top-left*) A signal comprising four local structures analogous to Fig. 6 (*top-left*). (*Top-right*) The signal $d$ corrupted by noise and normalized to the range [0, 1]. (*Bottom-left*) The approximation $u$ using the same constraints as for the experiment shown in Fig. 6. Restored structures emerge at the correct locations. (*Bottom-right*) Adding a single intersection constraint (see text) results in a good reconstruction of the original signal. This result illustrates that level-set constraints allow for the restoration of heavily corrupted signals based on prior knowledge about local signal *structure* rather than smoothness

Figure 8 provides another example. The constraints are defined in terms of the two filters

$$H^1 = I\ , \qquad H^2 = \begin{pmatrix} -1 & +1 & -1 \\ +1 & +1 & +1 \\ -1 & +1 & -1 \end{pmatrix}\ ,$$

and their response to the grid-image depicted in Fig. 5, top-left. Filter $H^2$ maps the "background" (value 0, black) to $\{0, -1\}$ and the "foreground" (value 1, white) to $\{0, 1, 3, 5\}$. The two constraints imposed are

$$\left| \left\{ x \in \Omega \,\middle|\, (u(x) \leq 0) \ \wedge \ \left( \left( H^2 u \right)(x) \leq 0 \right) \right\} \right| = \omega_1,$$

$$\left| \left\{ x \in \Omega \,\middle|\, (u(x) \geq 1) \ \wedge \ \left( \left( H^2 u \right)(x) \geq 0 \right) \right\} \right| = \omega_2.$$

Note that these two constraints merely *order* the set of feasible images $u$. For example, $H^2 u$ attains *multiple* positive values in the set $\{x \in \Omega \,|\, u(x) \geq 0\}$. Because the spatial support of $H^2$ overlaps at any location $x$ with adjacent pixel positions and, therefore, only *contextual* decisions are possible, the restoration of structured non-smooth signals simply becomes a consequence of spatial approximation (Fig. 8).

Finally, Fig. 9 shows another numerical experiment with a real image patch for validating our approach. The objective was to *generate* image structure in order to obtain some naturally looking contrast enhancement, that is without destroying the object's "shape". This is not straightforward to do by modifying image intensities directly. On the other hand, specifying plausible constraints with respect to intensity

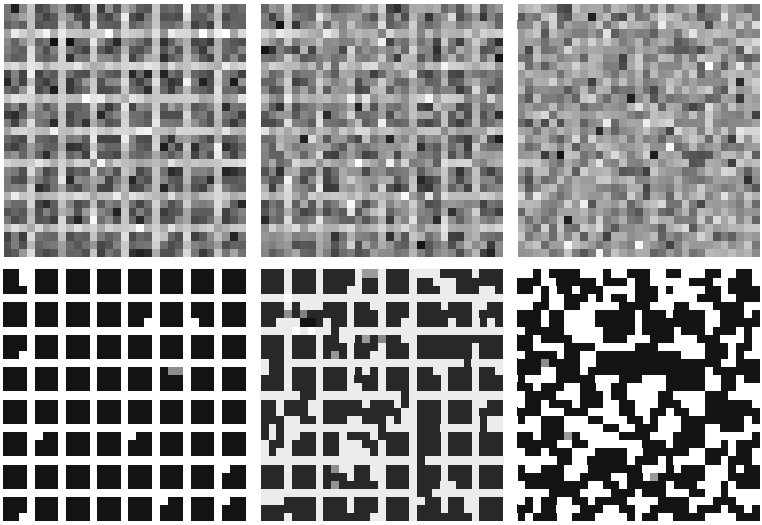

**Fig. 8.** (*Top, from left to right*) An image similar to Fig. 5 (*bottom-left*), a very noisy image, and pure noise. (*Bottom, from left to right*) The corresponding approximations based on constraints specified in the text. The results illustrate the *intersecting* level-set constraints considerably restrict the image space and allow for the restoration of non-smooth signals from noisy input data. The constraints are far from defining the original signal uniquely, however. Rather, they define a *class* of images satisfying the constraints

*derivatives* is a lot easier. For instance, choosing a level of medium magnitude 0.1, we simply measured the sizes of the corresponding level-sets of the partial derivatives, that is the level-sets of the *filtered* patches:

$$\left\{ x \in \Omega \,\middle|\, \left| \left( \partial_{x_i} d \right)(x) \right| \gtreqless 0.1 \right\}, \quad i = 1, 2.$$

Doubling the level to 0.2 and approximating the patch $d$ by $u$ subject to these four constraints, results in a desired contrast enhancement by automatically modifying the intensities of $d$ so as to satisfy the level-set constraints (Fig. 9, bottom left).

## 8. Conclusion

We presented a novel variational approach to image approximation subject to filter statistics represented by level-set constraints. The key modeling step involved is to embody the combinatorial part of the overall approximation problem – localization of (intersections of) level-sets – as the interaction of two convex optimization problems.

Our further work will focus on numerical issues enabling the processing of larger data sets, and on determining optimal filters for encoding a given image class.

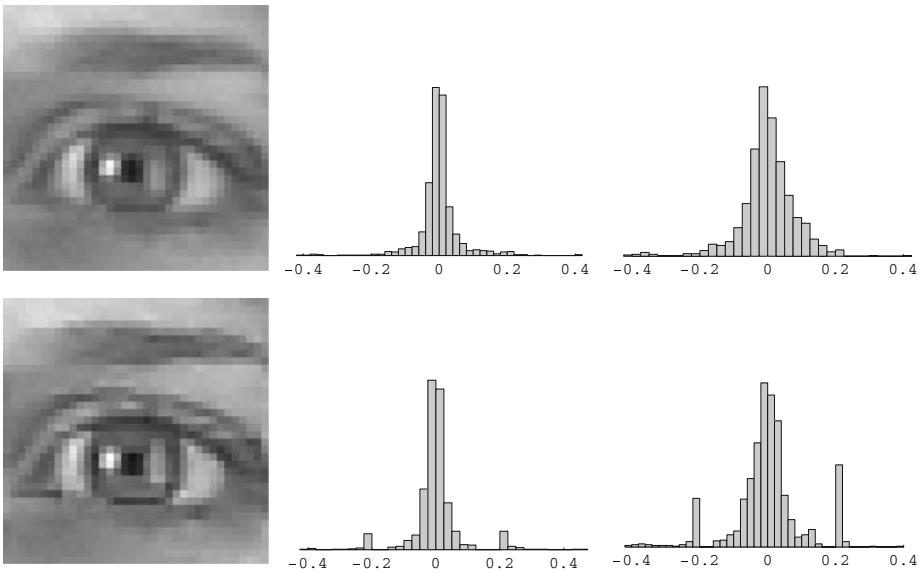


**Fig. 9.** (*Top, from left to right*) An original real image patch together with histograms of the partial derivatives $\partial_{x_1}$ and $\partial_{x_2}$. (*Bottom, from left to right*) Approximation of the image patch, subject to using *modified* histograms of the outputs of *derivative* filters as constraints (see text), results in a naturally looking contrast enhancement

## Appendix A: Proof of Proposition 7.1

We use the following general inversion formula for an arbitrary block-matrix in terms of the Schur complement $S_C$ of a regular matrix $C$:

$$\begin{pmatrix} A & B^\top \\ B & C \end{pmatrix}^{-1} = \begin{pmatrix} S_C^{-1} & -S_C^{-1} B^\top C^{-1} \\ -C^{-1} B S_C^{-1} & C^{-1} + C^{-1} B S_C^{-1} B^\top C^{-1} \end{pmatrix}, \quad S_C = A - B^\top C^{-1} B. \tag{55}$$

*Proof:* We first compute the inverse of the Hessian of the objective function (50). It has the form

$$\nabla^2 f(y) = \begin{pmatrix} \tau I & 0 & 0 \\ 0 & D_1 & c I \\ 0 & c I & D_2 \end{pmatrix} \tag{56}$$

with diagonal matrices

$$D_1 = c c_\pi I + P_1^{-2}, \quad D_2 = c c_\pi I + P_2^{-2}, \quad c = \tau \frac{\pi}{2}, \tag{57}$$

and $P_i = \mathrm{diag}(p_i)$, $i = 1, 2$. An elementary computation yields

$$\left(\nabla^2 f(y)\right)^{-1} = \begin{pmatrix} \tau^{-1} I & 0 & 0 \\ 0 & Q_1 & Q_2 \\ 0 & Q_2 & Q_3 \end{pmatrix}, \tag{58}$$

with diagonal matrices $Q_1$, $Q_2$, $Q_3$ and entries

$$Q_1(i,i) = \frac{D_2(i,i)}{D_1(i,i)D_2(i,i) - c^2}, \tag{59a}$$

$$Q_2(i,i) = \frac{c}{c^2 - D_1(i,i)D_2(i,i)}, \tag{59b}$$

$$Q_3(i,i) = \frac{D_1(i,i)}{D_1(i,i)D_2(i,i) - c^2}, \quad \forall i. \tag{59c}$$

Next, using the third block of equations of the system (52), we consider the *equivalent* system

$$\begin{pmatrix} \nabla^2 f(y) + A_y^\top H A_y & A_y^\top H A_\lambda & A_y^\top \\ A_\lambda^\top H A_y & A_\lambda^\top H A_\lambda & A_\lambda^\top \\ A_y & A_\lambda & 0 \end{pmatrix} \begin{pmatrix} \Delta y \\ \Delta\lambda \\ \zeta \end{pmatrix} = \begin{pmatrix} -\nabla f(y) \\ 0 \\ 0 \end{pmatrix}, \tag{60}$$

and choose $H$ to be the following block-matrix conforming to the structure of $A$ in (44):

$$H = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & I & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix}, \quad I \in \mathbb{R}^{(n\,|\mathcal{K}|)\times(n\,|\mathcal{K}|)}. \tag{61}$$

Inserting $H$ into (60) and computing the upper-left $2 \times 2$ block-matrix, we obtain

$$\begin{pmatrix} E & F^\top \\ F & G \end{pmatrix}, \quad E = \nabla^2 f(y) + F^\top G^{-1} F. \tag{62}$$

Because $\nabla^2 f(y)$ is regular by (58), the Schur complement $S_G = E - F^\top G^{-1} F$ and, in turn, the upper-left $2 \times 2$ block-matrix in (60) are regular as well.

The assertion of Proposition 7.1 follows from inserting $(A_y, A_\lambda) = A$ (44) and $H$ (61) into (62) and applying to the result (58) and (55). □

## References


[1] Anitescu, M.: On using the elastic mode in nonlinear programming approaches to mathematical programs with complementarity constraints. SIAM J Optim **15**(4), 1203–1236 (2005)

[2] An, L. T. H., Tao, P. D.: The DC (difference of convex functions) programming and DCA revisited with DC models of real world nonconvex optimization problems. Ann Oper Res **133**, 23–46 (2005)

[3] Aujol, J.-F., Gilboa, G., Chan, T., Osher, S.: Structure-texture image decomposition – modeling, algorithms, parameter selection. Int J Comp Vision **67**(1), 111–136 (2006)

[4] Boyd, S., Vandenberghe, L.: Convex optimization. Cambridge University Press, London (2004)

[5] Chambolle, A.: An algorithm for total variation minimization and applications. J Math Imaging Vis **20**, 89–97 (2004)

[6] Cottle, R. W., Pang, J.-S., Stone, R. E.: The linear complementarity problem. Academic, Dublin (1992)

[7] Facchinei, F., Pang, J.-S.: Finite-dimensional variational inequalities and complementarity problems, vol. I. Springer, New York (2003)

[8] Field, D. J.: Wavelets, vision, the statistics of natural scenes. Philos Trans R Soc Lond A **357**, 2527–2542 (1999)
[9] Graham, A.: Kronecker products and matrix calculus with applications. Wiley, New York (1981)
[10] Horst, R., Thoai, N. V.: DC programming: overview. J Optim Theory Appl **103**(1), 1–43 (1999)
[11] Hu, X. M., Ralph, D.: Convergence of a penalty method for mathematical programming with complementarity constraints. J Optim Theory Appl **123**(2), 365–390 (2004)
[12] Lee, A. B., Pedersen, K. S., Mumford, D.: The nonlinear statistics of high-contrast patches in natural images. Int J Comp Vision **54**, 83–103 (2003)
[13] Leyffer, S., Lopez-Calva, G., Nocedal, J.: Interior methods for mathematical programs with complementarity constraints. SIAM J Optim **17**(1), 52–77 (2006)
[14] Luo, Z.-Q., Pang, J.-S., Ralph, D.: Mathematical programs with equilibrium constraints. Cambridge University Press, London (1996)
[15] Raghunathan, A. U., Biegler, L. T.: An interior point method for mathematical programs with complementarity constraints (MPCCs). SIAM J Optim **15**(3), 720–750 (2005)
[16] Rockafellar, R. T., Wets, R. J.-B.: Variational analysis. In: Grundlehren der math. Wissenschaften, vol. 317. Springer, New York (1998)
[17] Rudin, L., Osher, S., Fatemi, E.: Nonlinear total variation based noise removal algorithms. Phys D **60**, 259–268 (1992)
[18] Scheel, H., Scholtes, S.: Mathematical program with complementarity constraints: stationarity, optimality and sensitivity. Math Oper Res **25**, 1–22 (2000)
[19] Scholtes, S.: Convergence properties of a regularization scheme for mathematical programs with complementarity constraints. SIAM J Optim **11**(4), 918–936 (2001)
[20] Srivastava, A., Liu, X., Grenander, U.: Universal analytical forms for modeling image probabilities. IEEE Trans Pattern Anal Mach Intell **24**(9), 1200–1214 (2002)
[21] Tao, P. D., An, L. T. H.: A D.C. optimization algorithm for solving the trust-region subproblem. SIAM J Optim **8**(2), 476–505 (1998)
[22] Zhu, S. C., Mumford, D.: Prior learning and gibbs reaction-diffusion. IEEE Trans Pattern Anal Mach Intell **19**(11), 1236–1250 (1997)
[23] Zhu, S. C., Wu, Y., Mumford, D.: Filters, random fields and maximum entropy (FRAME). Towards a unified theory for texture modeling. Int J Comp Vision **27**(2), 107–126 (1998)