SIAM J. IMAGING SCIENCES
Vol. 8, No. 3, pp. 2133–2159

© 2015 Society for Industrial and Applied Mathematics

Sparsity Based Methods for Overparameterized Variational Problems*

R. Giryes†, M. Elad‡, and A. M. Bruckstein‡

Abstract. Two complementary approaches have been extensively used in signal and image processing leading to novel results, the sparse representation methodology and the variational strategy. Recently, a new sparsity based model has been proposed, the cosparse analysis framework, which may potentially help in bridging sparse approximation based methods to the traditional total-variation minimization. Based on this, we introduce a sparsity based framework for solving overparameterized variational problems. The latter has been used to improve the estimation of optical flow and also for general denoising of signals and images. However, the recovery of the space varying parameters involved was not adequately addressed by traditional variational methods. We first demonstrate the efficiency of the new framework for one dimensional signals in recovering a piecewise linear and polynomial function. Then, we illustrate how the new technique can be used for denoising and segmentation of images.

Key words. sparsity, variational methods, total variation, overparameterization, cosparsity, compressed sensing, denoising, line segmentation, image segmentation

AMS subject classifications. 47N10, 35A15, 49N45, 65M20, 65J22, 68U10, 94A12, 65D18

DOI. 10.1137/140998585

1. Introduction.

Many successful signal and image processing techniques rely on the fact that the given signals or images of interest belong to a class described by a certain a priori known model. Given the model, the signal is processed by estimating the "correct" parameters of the model. For example, in the sparsity framework the assumption is that the signals belong to a union of low dimensional subspaces [6, 37, 4, 40]. In the variational strategy, a model is imposed on the variations of the signal, e.g., its derivatives are required to be smooth [52, 54, 10, 61].

Though both sparsity based and variational based approaches are widely used for signal processing and computer vision, they are often viewed as two different methods with little in common between them. One of the well-known variatonal tools is the total-variation (TV) regularization, used mainly for denoising and inverse problems. It can be formulated as [54]

$$\min_{\tilde{\mathbf{f}}} \left\| \mathbf{g} - \mathbf{M}\tilde{\mathbf{f}} \right\|_2^2 + \lambda \left\| \nabla \tilde{\mathbf{f}} \right\|_1, \tag{1}$$

*Received by the editors December 5, 2014; accepted for publication (in revised form) August 10, 2015; published electronically September 30, 2015. The research leading to these results has received funding from the European Research Council under European Unions Seventh Framework Program, ERC grant agreement 320649. This research was partially supported by AFOSR, ARO, NSF, ONR, and NGA.
http://www.siam.org/journals/siims/8-3/99858.html

†Department of Electrical and Computer Engineering, Duke University, Durham, NC 27708 (raja.giryes@duke.edu).
‡Department of Computer Science, The Technion - Israel Institute of Technology, Haifa, 32000, Israel (elad@cs.technion.ac.il, freddy@cs.technion.ac.il).

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 01/08/18 to 132.68.36.165. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

Downloaded 01/08/18 to 132.68.36.165. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

where $\mathbf{g} = \mathbf{M}\mathbf{f} + \mathbf{e} \in \mathbb{R}^m$ are the given noisy measurements, $\mathbf{M} \in \mathbb{R}^{m \times d}$ is a measurement matrix, $\mathbf{e} \in \mathbb{R}^m$ is an additive (typically white Gaussian) noise, λ is a regularization parameter, $\mathbf{f} \in \mathbb{R}^d$ is the original unknown signal to be recovered, and $\nabla \mathbf{f}$ is its gradients vector.

The anisotropic version of (1), which we will use in this work, is

$$\min_{\tilde{\mathbf{f}}} \left\| \mathbf{g} - \mathbf{M}\tilde{\mathbf{f}} \right\|_2^2 + \lambda \left\| \mathbf{\Omega}_{\text{DIF}} \tilde{\mathbf{f}} \right\|_1, \tag{2}$$

where $\mathbf{\Omega}_{\text{DIF}}$ is the finite-difference operator that returns the derivatives of the signal. In the one dimensional (1D) case it applies the filter $[1, -1]$, i.e.,

$$\mathbf{\Omega}_{\text{DIF}} = \mathbf{\Omega}_{\text{1D-DIF}} = \begin{bmatrix} 1 & -1 & 0 & \dots & \dots & 0 \\ 0 & 1 & -1 & \dots & \dots & 0 \\ \vdots & \ddots & \ddots & \ddots & \ddots & \vdots \\ 0 & 0 & \ddots & 1 & -1 & 0 \\ 0 & 0 & \dots & \dots & 1 & -1 \end{bmatrix}. \tag{3}$$

For images it returns the horizontal and vertical derivatives using the filters $[1, -1]$ and $[1, -1]^T$, respectively. Note that for 1D signals there is no difference between (1) and (2) as the gradient equals the derivative. However, in the two dimensional (2D) case (1) considers the sum of gradients (square root of the squared sum of the directional derivatives), while (2) considers the absolute sum of the directional derivatives, approximated by finite differences.

Recently, a very interesting connection has been drawn between the TV minimization problem and the sparsity model. It has been shown that (2) can be viewed as an ℓ_1-relaxation technique for approximating signals that are sparse in their derivatives domain, i.e., after applying the operator $\mathbf{\Omega}_{\text{DIF}}$ on them [48, 46, 33]. Such signals are said to be cosparse under the operator $\mathbf{\Omega}_{\text{DIF}}$ in the analysis (co)sparsity model [46].

Notice that the TV regularization is only one example from the variational framework. Another recent technique, which is the focus of this paper, is the overparameterization idea, which represents the signal as a combination of known functions weighted by space-variant parameters of the model [49, 50].

Let us introduce this overparameterized model via an example. If a 1D signal $\mathbf{f}$ is known to be piecewise linear, its ith element can be written as $\mathbf{f}(i) = \mathbf{a}(i) + \mathbf{b}(i)i$, where $\mathbf{a}(i)$ and $\mathbf{b}(i)$ are the local coefficients describing the local line curve. As such, the vectors $\mathbf{a}$ and $\mathbf{b}$ should be piecewise *constant*, with discontinuities in the same locations. Each constant interval in $\mathbf{a}$ and $\mathbf{b}$ corresponds to one linear segment in $\mathbf{f}$. When put in matrix-vector notation, $\mathbf{f}$ can be written alternatively as

$$\mathbf{f} = \mathbf{a} + \mathbf{Z}\mathbf{b}, \tag{4}$$

where $\mathbf{Z} \in \mathbb{R}^{d \times d}$ is a diagonal matrix with the values $1, 2, \dots, d$ on its main diagonal. For images this parameterization would similarly be $\mathbf{f}(i,j) = \mathbf{a}(i,j) + \mathbf{b}_1(i,j)i + \mathbf{b}_2(i,j)j$.

This strategy is referred to as "overparameterization" because the number of representation parameters is larger than the signal size. In the above 1D example, while the original signal contains d unknown values, the recovery problem that seeks $\mathbf{a}$ and $\mathbf{b}$ has twice as many

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 01/08/18 to 132.68.36.165. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

variables. Clearly, there are many other parameterization options for signals, beyond the linear one. Such parameterizations have been shown to improve the denoising performance of the solution of the problem posed in (1) in some cases [49], and to provide very high quality results for optical flow estimation [50, 53].

1.1. Our contribution. The true force behind overparameterization is that while it uses more variables than needed for representing the signals, these are often more naturally suited to describe its structure. For example, if a signal is piecewise linear then we may impose a constraint on the overparameterization coefficients $\mathbf{a}$ and $\mathbf{b}$ to be piecewise constant.

Note that piecewise constant signals are sparse under the $\mathbf{\Omega}_{\mathrm{DIF}}$ operator. Therefore, for each of the coefficients we can use the tools developed in the analysis sparsity model [33, 34, 28, 51, 9]. However, in our case, $\mathbf{a}$ and $\mathbf{b}$ are jointly sparse, i.e., their change points are collocated and therefore an extension is necessary.

Constraints on the structure in the sparsity pattern of a representation have already been analyzed in the literature. They are commonly referred to as joint sparsity models, and those are found in the literature, both in the context of handling groups of signals [15, 60, 59, 63, 42, 26], or when considering blocks of nonzeros in a single representation vector [64, 24, 57, 23, 3]. We use these tools to extend the existing analysis techniques to handle the block sparsity in our overparameterized scheme.

In this paper we introduce a general sparsity based framework for solving overparameterized variational problems. As the structure of these problems enables segmentation while recovering the signal, we provide an elegant way for recovering a signal from its deteriorated measurements by using an ℓ_0 approach, which is accompanied by theoretical guarantees. We demonstrate the efficiency of the new framework for 1D functions in recovering piecewise polynomial signals. Then we shift our view to images and demonstrate how the new approach can be used for denoising and segmentation.

1.2. Organization. This paper is organized as follows: In section 2 we present the overparameterized variational model with more details. In section 3 we describe briefly the synthesis and analysis sparsity models. In sections 4 and 5 we introduce a new framework for solving overparameterized variational problems using sparsity. Section 4 proposes a recovery strategy for the 1D polynomial case based on the SSCoSaMP technique (defined in section 4) with optimal projections [17, 35, 36]. We provide stable recovery guarantees for this algorithm for the case of an additive adversarial noise and denoising guarantees for the case of a zero-mean white Gaussian noise. In section 5 we extend our scheme beyond the 1D case to higher dimensional polynomial functions such as images. We employ an extension of the GAPN algorithm [45] for block sparsity for this task. In section 6 we present experiments for linear overparameterization of images and 1D signals. We demonstrate how the proposed method can be used for image denoising and segmentation. Section 7 concludes our work and proposes future directions of research.

2. The overparameterized variational framework.

Considering again the linear relationship between the measurements and the unknown signal,

$$\mathbf{g} = \mathbf{M}\mathbf{f} + \mathbf{e}, \tag{5}$$

note that without a prior knowledge on $\mathbf{f}$ we cannot recover it from $\mathbf{g}$ if $m < d$ or $\mathbf{e} \neq 0$.

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 01/08/18 to 132.68.36.165. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

In the variational framework, a regularization is imposed on the variations of the signal $\mathbf{f}$. One popular strategy for recovering the signal in this framework is by solving the following minimization problem,

$$\min_{\tilde{\mathbf{f}}} \left\| \mathbf{g} - \mathbf{M}\tilde{\mathbf{f}} \right\|_2^2 + \lambda \left\| \mathcal{A}(\tilde{\mathbf{f}}) \right\|_p, \tag{6}$$

where λ is the regularization weight and $p \geq 1$ is the type of norm used with the regularization operator $\mathcal{A}$, which is typically a local operator. For example, for $p = 1$ and $\mathcal{A} = \nabla$ we get the TV minimization (2). Another example for a regularization operator is the Laplace operator $\mathcal{A} = \nabla^2$. Other types of regularization operators and variational formulations can be found in [43, 11, 62, 2].

Recently, the overparameterized variational framework has been introduced as an extension to the traditional variational methodology [49, 50, 53, 56]. Instead of applying a regularization on the signal itself, it is applied on the coefficients of the signal under a global parameterization of the space. Each element of the signal can be modeled as $\mathbf{f}(i) = \sum_{j=1}^{n} \mathbf{b}_j(i)\mathbf{x}_j(i)$, where the $\{\mathbf{b}_j\}_{j=1}^n$ are the coefficients vectors and the $\{\mathbf{x}_j\}_{j=1}^n$ contain the parameterization basis functions for the space.

Denoting by $\mathbf{X}_i \triangleq \text{diag}(\mathbf{x}_i)$ the diagonal matrix that has the vector $\mathbf{x}_i$ on its diagonal, we can rewrite the above as $\mathbf{f} = \sum_{j=1}^{n} \mathbf{X}_j \mathbf{b}_j$. With this notation, the overparameterized minimization problem becomes

$$\min_{\tilde{\mathbf{b}}_i, 1 \leq i \leq n} \left\| \mathbf{g} - \mathbf{M} \sum_{i=1}^{n} \mathbf{X}_i \tilde{\mathbf{b}}_i \right\|_2^2 + \sum_{i=1}^{n} \lambda_i \left\| \mathcal{A}_i(\tilde{\mathbf{b}}_i) \right\|_{p_i}, \tag{7}$$

where each coefficient $\mathbf{b}_i$ is regularized separately by the operator $\mathcal{A}_i$ (which can be the same one for all the coefficients).[1]

Returning to the example of a linear overparameterization, we have that $\mathbf{f} = \mathbf{a} + \mathbf{Z}\mathbf{b}$, where in this case $\mathbf{X}_1 = \mathbf{I}$ (the identity matrix) and $\mathbf{X}_2 = \mathbf{Z} = \text{diag}(1, \ldots, d)$, a diagonal matrix with $1, \ldots, d$ on its diagonal. If $\mathbf{f}$ is a piecewise linear function then the coefficient vectors $\mathbf{a}$ and $\mathbf{b}$ should be piecewise constant, and therefore it would be natural to regularize these coefficients with the gradient operator. This leads to the following minimization problem:

$$\min_{\tilde{\mathbf{a}}, \tilde{\mathbf{b}}} \left\| \mathbf{g} - \mathbf{M}\left(\tilde{\mathbf{a}} + \mathbf{Z}\tilde{\mathbf{b}}\right) \right\|_2^2 + \lambda_1 \left\| \nabla \tilde{\mathbf{a}} \right\|_1 + \lambda_2 \left\| \nabla \tilde{\mathbf{b}} \right\|_1, \tag{8}$$

which is a special case of (7). The two main advantages of using the overparameterized formulation are these: (i) the new unknowns have a simpler form (e.g., a piecewise linear signal is treated by piecewise constant unknowns), and thus are easier to recover, and (ii) this formulation leads to recovering the parameters of the signal along with the signal itself.

The overparameterization idea, as introduced in [49, 50, 53, 56], builds upon the vast work in signal processing that refers to variational methods. As such, there are no known guarantees for the quality of the recovery of the signal when using the formulation posed in (8) or its variants. Moreover, it has been shown in [56] that even for the case of $\mathbf{M} = \mathbf{I}$ (and, obviously,

[1] We note that it is possible to have more than one regularization for each coefficient, as practiced in [56].

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 01/08/18 to 132.68.36.165. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

$\mathbf{e} \neq 0$), a poor recovery is achieved in recovering $\mathbf{f}$ and its parameterization coefficients. Note that the same happens even if more sophisticated regularizations are combined and applied on $\mathbf{a}$, $\mathbf{b}$, and eventually on $\mathbf{f}$ [56].

This leads us to look for another strategy to approach the problem of recovering a piecewise linear function from its deteriorated measurement $\mathbf{g}$. Before describing our new scheme, we introduce in the next section the sparsity model that will aid us in developing this alternative strategy.

3. The synthesis and analysis sparsity models. A popular prior for recovering a signal $\mathbf{f}$ from its distorted measurements (as posed in (5)) is the sparsity model [6, 37]. The idea behind it is that if we know a priori that $\mathbf{f}$ resides in a union of low dimensional subspaces, which do not intersect trivially with the null space of $\mathbf{M}$, then we can estimate $\mathbf{f}$ stably by selecting the signal that belongs to this union of subspaces and is the closest to $\mathbf{g}$ [4, 40].

In the classical sparsity model, the signal $\mathbf{f}$ is assumed to have a sparse representation $\boldsymbol{\alpha}$ under a given dictionary $\mathbf{D}$, i.e., $\mathbf{f} = \mathbf{D}\boldsymbol{\alpha}, \|\boldsymbol{\alpha}\|_0 \leq k$, where $\|\cdot\|_0$ is the ℓ_0 pseudonorm that counts the number of nonzero entries in a vector, and k is the sparsity of the signal. Note that each low dimensional subspace in the standard sparsity model, known also as the synthesis model, is spanned by a collection of k columns from $\mathbf{D}$. With this model we can recover $\mathbf{f}$ by solving

$$\min_{\boldsymbol{\alpha}} \|\mathbf{g} - \mathbf{M}\mathbf{D}\boldsymbol{\alpha}\|_2^2 s.t. \|\boldsymbol{\alpha}\|_0 \leq k \tag{9}$$

if k is known, or

$$\min_{\boldsymbol{\alpha}} \|\boldsymbol{\alpha}\|_0 s.t. \|\mathbf{g} - \mathbf{M}\mathbf{D}\boldsymbol{\alpha}\|_2^2 \leq \|\mathbf{e}\|_2^2 \tag{10}$$

if we have information about the energy of the noise $\mathbf{e}$. Obviously, once we get $\boldsymbol{\alpha}$, the desired recovered signal is simply $\mathbf{D}\boldsymbol{\alpha}$. As both of these minimization problems are NP-hard [19], many approximation techniques have been proposed to approximate their solution, accompanied with recovery guarantees that depend on the properties of the matrices $\mathbf{M}$ and $\mathbf{D}$. These include ℓ_1-relaxation [22, 14, 21], known also as LASSO [58], matching pursuit [41], orthogonal matching pursuit (OMP) [13, 18], compressive sampling matching pursuit (CoSaMP) [47], subspace pursuit (SP) [16], iterative hard thresholding (IHT) [5] and hard thresholding pursuit (HTP) [27].

Another framework for modeling a union of low dimensional subspaces is the analysis one [46, 22]. This model considers the behavior of $\boldsymbol{\Omega}\mathbf{f}$, the signal after applying a given operator $\boldsymbol{\Omega}$ on it, and assumes that this vector is sparse. Note that here the zeros are those that characterize the subspace in which $\mathbf{f}$ resides, as each zero in $\boldsymbol{\Omega}\mathbf{f}$ corresponds to a row in $\boldsymbol{\Omega}$ to which $\mathbf{f}$ is orthogonal. Therefore, $\mathbf{f}$ resides in a subspace orthogonal to the one spanned by these rows. We say that $\mathbf{f}$ is cosparse under $\boldsymbol{\Omega}$ with a cosupport Λ if $\boldsymbol{\Omega}_\Lambda \mathbf{f} = 0$, where $\boldsymbol{\Omega}_\Lambda$ is a submatrix of $\boldsymbol{\Omega}$ with the rows corresponding to the set Λ.

The analysis variants of (9) and (10) for estimating $\mathbf{f}$ are

$$\min_{\tilde{\mathbf{f}}} \left\|\mathbf{g} - \mathbf{M}\tilde{\mathbf{f}}\right\|_2^2 s.t. \left\|\boldsymbol{\Omega}\tilde{\mathbf{f}}\right\|_0 \leq k, \tag{11}$$

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

where k is the number of nonzeros in $\mathbf{\Omega f}$, and

$$\min_{\tilde{\mathbf{f}}} \left\|\mathbf{\Omega}\tilde{\mathbf{f}}\right\|_0 s.t. \left\|\mathbf{g}-\mathbf{M}\tilde{\mathbf{f}}\right\|_2^2 \leq \|\mathbf{e}\|_2^2 . \tag{12}$$

As in the synthesis case, these minimization problems are also NP-hard [46] and approximation techniques have been proposed including greedy analysis pursuit (GAP) [46], GAP noise (GAPN) [45], analysis CoSAMP (ACoSaMP), analysis SP, analysis IHT and analysis HTP [33].

4. Overparameterization via the analysis sparsity model. With the sparsity models now defined, we revisit the overparameterization variational problem. If we know that our signal $\mathbf{f}$ is piecewise linear, then it is clear that the coefficients parameters should be piecewise constant with the same discontinuity locations, when linear overparameterization is used. We denote by k the number of these discontinuity locations.

As a reminder, we rewrite $\mathbf{f} = [\mathbf{I}, \mathbf{Z}] \left[\mathbf{a}^T, \mathbf{b}^T\right]^T$. Note that $\mathbf{a}$ and $\mathbf{b}$ are jointly sparse under $\mathbf{\Omega}_{\text{DIF}}$, i.e, $\mathbf{\Omega}_{\text{DIF}}\mathbf{a}$ and $\mathbf{\Omega}_{\text{DIF}}\mathbf{b}$ have the same nonzero locations. With this observation we can extend the analysis minimization problem (11) to support the structured sparsity in the vector $\left[\mathbf{a}^T, \mathbf{b}^T\right]^T$, leading to the following minimization problem:

$$\min_{\mathbf{a},\mathbf{b}} \left\|\mathbf{g}-\mathbf{M}\left[\mathbf{I},\mathbf{Z}\right]\begin{bmatrix}\mathbf{a}\\ \mathbf{b}\end{bmatrix}\right\|_2^2 \quad \text{s.t.} \quad \left\||\mathbf{\Omega}_{\text{DIF}}\mathbf{a}| + |\mathbf{\Omega}_{\text{DIF}}\mathbf{b}|\right\|_0 \leq k, \tag{13}$$

where $|\mathbf{\Omega}_{\text{DIF}}\mathbf{a}|$ denotes applying elementwise the absolute value on the entries of $\mathbf{\Omega}_{\text{DIF}}\mathbf{a}$.

Note that we can have a similar formulation for this problem also in the synthesis framework using the Heaviside dictionary

$$\mathbf{D}_{HS} = \begin{bmatrix} 1 & 1 & \dots & 1 & 1 \\ 0 & 1 & \dots & \ddots & 1 \\ \vdots & 0 & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & 1 & 1 \\ 0 & 0 & \dots & 0 & 1 \end{bmatrix}, \tag{14}$$

whose atoms are step functions of different length. We use the known observation that every 1D signal with k change points can be sparsely represented using $k+1$ atoms from $\mathbf{D}_{HS}$ (k columns for representing the change points plus one for the DC component). One way to observe that is by the fact that $\mathbf{\Omega}_{\text{DIF}}\tilde{\mathbf{D}}_{HS} = \mathbf{I}$, where $\tilde{\mathbf{D}}_{HS}$ is a submatrix of $\mathbf{D}_{HS}$ obtained by removing the last column of $\mathbf{D}_{HS}$ (the DC component). Therefore, one may recover the coefficient parameters $\mathbf{a}$ and $\mathbf{b}$ by their sparse representations $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$, solving

$$\min_{\boldsymbol{\alpha},\boldsymbol{\beta}} \left\|\mathbf{g}-\mathbf{M}\left[\mathbf{I},\mathbf{Z}\right]\begin{bmatrix}\mathbf{D}_{HS} & 0\\ 0 & \mathbf{D}_{HS}\end{bmatrix}\begin{bmatrix}\boldsymbol{\alpha}\\ \boldsymbol{\beta}\end{bmatrix}\right\|_2^2 \quad \text{s.t.} \quad \left\||\boldsymbol{\alpha}| + |\boldsymbol{\beta}|\right\|_0 \leq k, \tag{15}$$

Downloaded 01/08/18 to 132.68.36.165. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 01/08/18 to 132.68.36.165. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

where $\mathbf{a} = \mathbf{D}_{HS}\boldsymbol{\alpha}$ and $\mathbf{b} = \mathbf{D}_{HS}\boldsymbol{\beta}$. This minimization problem can be approximated using block-sparsity techniques such as the group-LASSO estimator [64], the mixed-ℓ_2/ℓ_1 relaxation (extension of the ℓ_1 relaxation) [24, 57], the block OMP (BOMP) algorithm [23], or the extensions of CoSaMP and IHT for structured sparsity [3]. The joint sparsity framework can also be used with (15) [15, 60, 59, 63, 42, 26].

The problem with the above synthesis techniques is twofold: (i) no recovery guarantees exist for this formulation with the dictionary $\mathbf{D}_{HS}$; (ii) It is hard to generalize the model in (9) to higher order signals, e.g., images.

The reason that no theoretical guarantees are provided for the $\mathbf{D}_{HS}$ dictionary is the high correlation between its columns. These create high ambiguity, causing the classical synthesis techniques to fail in recovering the representations $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$. This problem has been addressed in several contributions that have treated the signal directly and not its representation [17, 35, 36, 30, 31, 32].

We introduce an algorithm that approximates the solutions of both (9) and (11) and has theoretical reconstruction performance guarantees for 1D functions $\mathbf{f}$ with matrices $\mathbf{M}$ that are near isometric for piecewise polynomial functions. In the next section we shall present another algorithm that does not have such guarantees but is generalizable to higher order functions.

Though till now we have restricted our discussion only to piecewise linear functions, we turn now to look at the more general case of piecewise 1D polynomial functions of degree n. Note that this method approximates the following minimization problem, which is a generalization of (13) to any polynomial of degree n,

$$\min_{\mathbf{b}_0,\mathbf{b}_1,\ldots,\mathbf{b}_n} \left\| \mathbf{g} - \mathbf{M}\left[\mathbf{I}, \mathbf{Z}, \mathbf{Z}^2, \ldots, \mathbf{Z}^n\right] \begin{bmatrix} \mathbf{b}_0 \\ \mathbf{b}_1 \\ \vdots \\ \mathbf{b}_n \end{bmatrix} \right\|_2^2 \quad \text{s.t.} \quad \left\| \sum_{i=0}^{n} |\boldsymbol{\Omega}_{\text{DIF}}\mathbf{b}_i| \right\|_0 \leq k, \tag{16}$$

where $|\boldsymbol{\Omega}_{\text{DIF}}\mathbf{b}_i|$ is an elementwise operation that calculates the absolute value of each entry in $\boldsymbol{\Omega}_{\text{DIF}}\mathbf{b}_i$.

We employ the signal space CoSaMP (SSCoSaMP) strategy[2] [17, 35] to approximate the solution of (16). This algorithm assumes the existence of a projection that for a given signal finds its closest signal (in the ℓ_2-norm sense) that belongs to the model,[3] where in our case the model consists of piecewise polynomial functions with k jump points. This algorithm, along with the projection required, are presented in Appendix A.

4.1. Recovery guarantees for piecewise polynomial functions. To provide theoretical guarantees for the recovery by SSCoSaMP, we employ two theorems from [35] and [36]. These

[2] In a very similar way we could have used ACoSaMP [33, 28].

[3] Note that in [17, 35] the projection might be allowed to be near optimal in the sense that the projection error is close to the optimal error up to a multiplicative constant.

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 01/08/18 to 132.68.36.165. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

lead to reconstruction error bounds for SSCoSaMP that guarantee stable recovery if the noise is adversarial, and an effective denoising effect if it is zero-mean white Gaussian.

Both theorems rely on the following property of the measurement matrix $\mathbf{M}$, which is a special case of the $\mathbf{D}$-RIP [9] and $\mathbf{\Omega}$-RIP [33].

Definition 4.1. *A matrix $\mathbf{M}$ has a polynomial restricted isometry property (RIP) of order n (P_n-RIP) with a constant δ_k if for any piecewise polynomial function $\mathbf{f}$ of order n with k jumps we have*

$$(1-\delta_k)\left\|\mathbf{f}\right\|_2^2 \leq \left\|\mathbf{M}\mathbf{f}\right\|_2^2 \leq (1+\delta_k)\left\|\mathbf{f}\right\|_2^2. \tag{17}$$

Having the P_n-RIP definition we turn to present the first theorem, which treats the adversarial noise case.

Theorem 4.2 (based on Corollary 3.2 in [35]). *Let $\mathbf{f}$ be a piecewise polynomial function of order n, $\mathbf{e}$ be an adversarial bounded noise, and $\mathbf{M}$ satisfy the P_n-RIP (17) with a constant $\delta_{4k} < 0.046$. Then after a finite number of iterations, SSCoSaMP yields*

$$\left\|\hat{\mathbf{f}}-\mathbf{f}\right\|_2 \leq C\left\|\mathbf{e}\right\|_2, \tag{18}$$

where $C > 2$ is a constant depending on δ_{4k}.

Note that the above theorem implies that we may compressively sense piecewise polynomial functions and achieve a perfect recovery in the noiseless case $\mathbf{e} = 0$. Note also that if $\mathbf{M}$ is a subgaussian random matrix then it is sufficient to use only $m = O(k(n + \log(d))$ measurements [4, 33]. It is worth mentioning that Corollary 3.2 in [35] provides stable recovery guarantees for general sparse vectors under a given dictionary D with the assumption that there exists a near-optimal projection algorithm that projects any vector to its closest sparse vector under the same dictionary. We can deduce Theorem 4.2 from Corollary 3.2 in [35] due to the optimal projection algorithm proposed in Section A.1.

Though the above theorem is important for compressed sensing, it does not guarantee noise reduction, even for the case $\mathbf{M} = \mathbf{I}$, as $C > 2$. The reason for this is that the noise here is adversarial, leading to a worst-case bound. By introducing a random distribution for the noise, one may get better reconstruction guarantees. The following theorem assumes that the noise is randomly Gaussian distributed, this way enabling it to provide effective denoising guarantees.

Theorem 4.3 (based on Theorem 1.7 in [36]). *Assume the conditions of Theorem 4.2 such that $\mathbf{e}$ is a random zero-mean white Gaussian noise with a variance σ^2. Then after a finite number of iterations, SSCoSaMP yields*

$$\left\|\hat{\mathbf{f}}-\mathbf{f}\right\|_2 \leq C\sqrt{(1+\delta_{3k})3k}\left(1+\sqrt{2(1+\beta)\log(nd)}\right)\sigma \tag{19}$$

with probability exceeding $1-\frac{2}{(3k)!}(nd)^{-\beta}$.

The bound in the theorem can be given on the expected error instead of being given only with high probability using the proof technique in [29]. We remark that if we were given an

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 01/08/18 to 132.68.36.165. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

oracle that has foreknowledge of the locations of the jumps in the parameterization, the error we would get would be $O(\sqrt{k}\sigma)$. As the $\log(nd)$ factor in our bound is inevitable [7], we may conclude that our guarantee is optimal up to a constant factor. Note that Theorem 1.7 in [35] provides near-oracle performance guarantees for block-sparse vectors under a given dictionary D with the assumption that there exists a near-optimal projection algorithm that projects any vector to its closest sparse vector under the same dictionary. We can apply the result of Theorem 1.7 in [35], resulting with Theorem 4.3, due to the optimal projection algorithm proposed in Section A.1.

5. Sparsity based overparameterized variational algorithm for high dimensional functions. We now turn to generalize the model in (13) to support other overparameterization forms, including higher dimensional functions such as images. We consider the case where an upper bound for the noise energy is given and not the sparsity k, as is common in many applications. Notice that for the synthesis model, such a generalization is not trivial because while it is easy to extend the $\mathbf{\Omega}_{\text{DIF}}$ operator to high dimensions, it is not clear how to do this for the Heaviside dictionary.

Therefore we consider an overparameterized version of (12), where the noise energy is known and the analysis model is used. Let $\mathbf{X}_1, \dots \mathbf{X}_n$ be matrices of the space variables and $\mathbf{b}_1 \dots \mathbf{b}_n$ their coefficients parameters. For example, in a 2D (image) case of piecewise linear constants, $\mathbf{X}_1$ will be the identity matrix, $\mathbf{X}_2$ will be a diagonal matrix with the values $[1, 2, \dots, d, 1, 2, \dots, d, \dots 1, 2, \dots, d]$ on its main diagonal, and $\mathbf{X}_3$ will similarly be a diagonal matrix with $[1, 1, \dots, 1, 2, 2, \dots, 2, \dots d, d, \dots, d]$ on its main diagonal. Assuming that all the coefficient parameters are jointly sparse under a general operator $\mathbf{\Omega}$, we may recover these coefficients by solving

$$\left[\hat{\mathbf{b}}_1^T, \dots, \hat{\mathbf{b}}_n^T\right]^T = \min_{\tilde{\mathbf{b}}_1,\dots,\tilde{\mathbf{b}}_n} \left\| \sum_{i=1}^{n} \left| \mathbf{\Omega} \tilde{\mathbf{b}}_i \right| \right\|_0 \quad \text{s.t.} \quad \left\| \mathbf{g} - \mathbf{M} \left[\mathbf{X}_1, \dots \mathbf{X}_n\right] \begin{bmatrix} \tilde{\mathbf{b}}_1 \\ \vdots \\ \tilde{\mathbf{b}}_n \end{bmatrix} \right\|_2 \leq \|\mathbf{e}\|_2 . \tag{20}$$

Having an estimate for all these coefficients, our approximation for the original signal $\mathbf{f}$ is $\hat{\mathbf{f}} = [\mathbf{X}_1, \dots, \mathbf{X}_n] \left[\hat{\mathbf{b}}_1^T, \dots, \hat{\mathbf{b}}_n^T\right]^T$.

As the minimization problem in (20) is NP-hard we suggest solving it by a generalization of the GAPN algorithm [45]—the block GAPN (BGAPN). We introduce this extension in Appendix B.

This algorithm aims at finding in a greedy way the rows of $\mathbf{\Omega}$ that are orthogonal to the space variables $\mathbf{b}_1 \dots \mathbf{b}_n$. Notice that once we find the indices of these rows, the set Λ that satisfies $\mathbf{\Omega}_\Lambda \mathbf{b}_i = 0$ for $i = 1 \dots n$ ($\mathbf{\Omega}_\Lambda$ is the submatrix of $\mathbf{\Omega}$ with the rows corresponding to

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

the set Λ), we may approximate $\mathbf{b}_1 \dots \mathbf{b}_n$ by solving

$$\left[\hat{\mathbf{b}}_1^T, \dots, \hat{\mathbf{b}}_n^T\right]^T = \min_{\tilde{\mathbf{b}}_1,\dots,\tilde{\mathbf{b}}_n} \sum_{i=1}^{n} \left\|\mathbf{\Omega}_\Lambda \tilde{\mathbf{b}}_i\right\|_2^2 \quad \text{s.t.} \quad \left\|\mathbf{g} - \mathbf{M}\left[\mathbf{X}_1, \dots \mathbf{X}_n\right] \begin{bmatrix} \tilde{\mathbf{b}}_1 \\ \vdots \\ \tilde{\mathbf{b}}_n \end{bmatrix}\right\|_2 \leq \|\mathbf{e}\|_2 . \tag{21}$$

Therefore, BGAPN approximates $\mathbf{b}_1 \dots \mathbf{b}_n$ by finding Λ first. It starts with a Λ that includes all the rows of $\mathbf{\Omega}$ and then gradually removes elements from it by solving the problem posed in (21) at each iteration and then finding the row in $\mathbf{\Omega}$ that has the largest correlation with the current temporal solution $\left[\hat{\mathbf{b}}_1^T, \dots, \hat{\mathbf{b}}_n^T\right]^T$.

Note that there are no known recovery guarantees for BGAPN of the form we have had for SSCoSaMP before. Therefore, we present its efficiency in several experiments in the next section. As explained in Appendix B, the advantages of BGAPN over SSCoSaMP, despite the lack of theoretical guarantees, are that (i) it does not need foreknowledge of k and (ii) it is easier to use with higher dimensional functions.

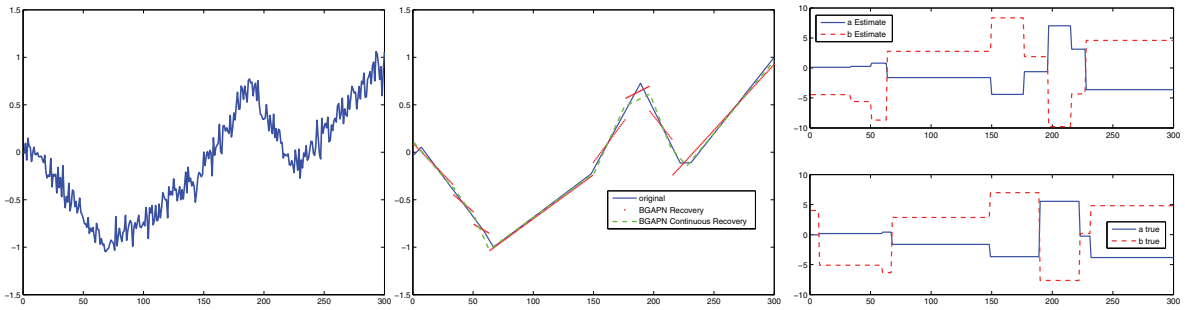
Before we move to the next section we note that one of the advantages of the above formulation and the BGAPN algorithm is the relative ease of adding to it new constraints. For example, we may encounter piecewise polynomial functions that are also continuous. However, we do not have such a continuity constraint in the current formulation. As we shall see in the next section, the absence of such a constraint allows jumps in the discontinuity points between the polynomial segments and therefore it is important to add it to the algorithm to get a better reconstruction.

One possibility for solving this problem is to add a continuity constraint on the jump points of the signal. In Appendix B we present also a modified version of the BGAPN algorithm that imposes such a continuity constraint, and in the next section we shall see how this handles the problem. Note that this is only one example of a constraint that one may add to the BGAPN technique. For example, in images one may add a smoothness constraint on the edges' directions.

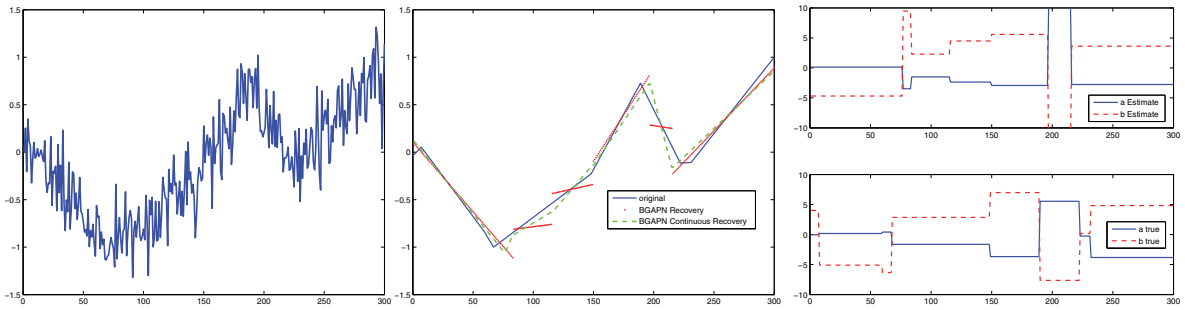
6. Experiments. For demonstrating the efficiency of the proposed method we perform several tests. We start with the 1D case, testing our polynomial fitting approach with the continuity constraint and without it for continuous piecewise polynomials of first and second degrees. We compare these results to the optimal polynomial approximation scheme presented in section 4 and to the variational approach in [56]. We continue with a compressed sensing experiment for discontinuous piecewise polynomials and compare BGAPN with SSCoSaMP. Then we perform some tests on images using BGAPN. We start by denoising cartoon images using the piecewise linear model. We compare our outcome with the one of TV denoising [54] and show that our result does not suffer from a staircasing effect [55]. We compare also to a TV denoising version with overparameterization [49]. Then we show how our framework may be used for image segmentation, drawing a connection to the Mumford–Shah functional [44, 1]. We compare our results with the ones obtained by the popular graph-cuts based segmentation [25].

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 01/08/18 to 132.68.36.165. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php



(a) Noisy function $\sigma = 0.1$ (b) Function recovery for $\sigma = 0.1$ (c) Coefficients parameters recovery for $\sigma = 0.1$



(d) Noisy function $\sigma = 0.25$ (e) Function recovery for $\sigma = 0.25$ (f) Coefficients parameters recovery for $\sigma = 0.25$

Figure 1. *Recovery of a piecewise linear function using the BGAPN algorithm with and without a constraint on the continuity.*

6.1. Continuous piecewise polynomial functions denoising. In order to check the performance of the polynomial fitting, we generate random continuous piecewise linear and second order polynomial functions with 300 samples, 6 jumps, and a dynamic range $[-1, 1]$. Then we contaminate the signal with a white Gaussian noise with a standard deviation from the set $\{0.05, 0.1, 0.15, \ldots, 0.5\}$.

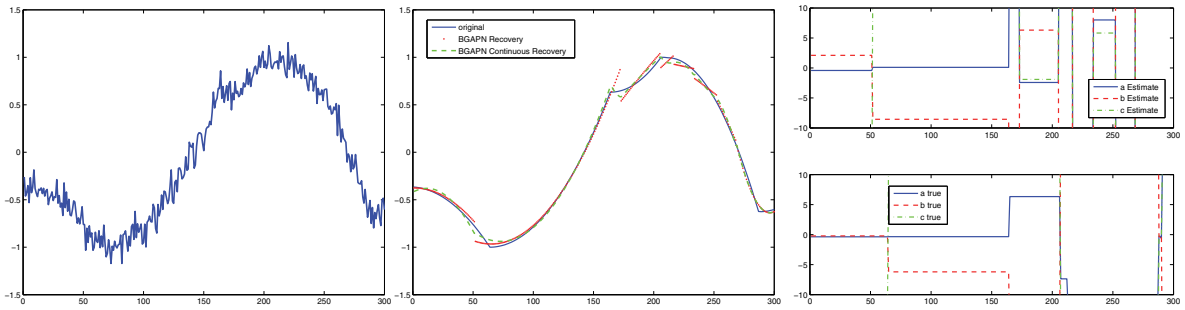
We compare the recovery result of BGAPN with and without the continuity constraint with the one of the optimal approximation.[4] Figures 1 and 2 present BGAPN reconstruction results for the linear and second order polynomial cases, respectively, for two different noise levels. It can be observed that the addition of the continuity constraint is essential for the correctness of the recovery. Indeed, without it we get jumps between the segments. Note also that the number of jumps in our recovery may be different than the one of the original signal as BGAPN does not have preliminary information about it. However, it still manages to recover the parameterization in a good way, especially in the lower noise case.

The possibility of providing a parametric representation is one of the advantages of our method. Indeed, one may achieve good denoising results without using the linear model in terms of MSE using methods such as free-knot spline [39]. However, the approximated

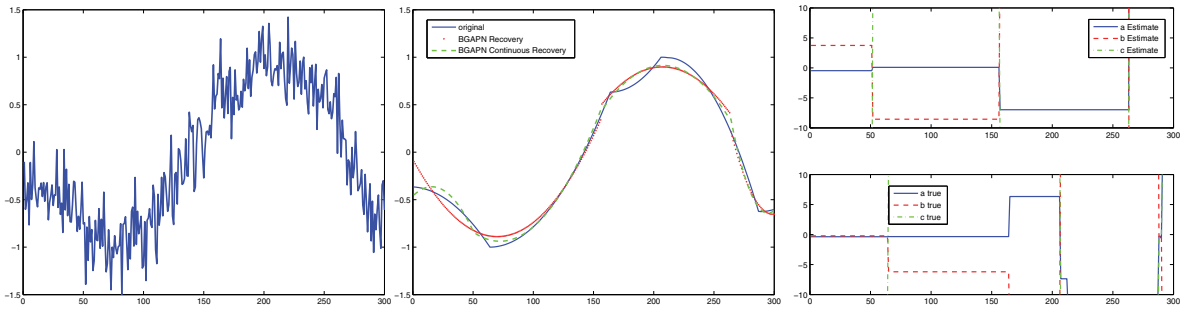
[4]We have done the same experiment with the BOMP algorithm [23], adopting the synthesis framework, with and without the continuity constraint, and observed that it performs very similarly to BGAPN.

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 01/08/18 to 132.68.36.165. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php



(a) Noisy function $\sigma = 0.1$ (b) Function recovery for $\sigma = 0.1$ (c) Coefficients parameters recovery for $\sigma = 0.1$



(d) Noisy function $\sigma = 0.25$ (e) Function recovery for $\sigma = 0.25$ (f) Coefficients parameters recovery for $\sigma = 0.25$

Figure 2. *Recovery of a piecewise second order polynomial function using the BGAPN algorithm with and without a constraint on the continuity.*

function is not guaranteed to be piecewise linear and therefore learning the change points from it is suboptimal. See [56] and the references therein for more details.

To evaluate our method with respect to its MSE we compare it with the optimal approximation for a piecewise polynomial function presented in Appendix A.1. Note that the target signals are continuous while this algorithm does not use this assumption. Therefore, we add the continuity constraint to this method as postprocessing (unlike BGAPN that merges this into its steps). We take the changing points it has recovered and project the noisy measurement $\mathbf{g}$ to its closest continuous piecewise polynomial function with the same discontinuities.

Figure 3 presents the recovery performance of BGAPN and the projection algorithm with and without the continuous constraint. Without the constraint, it can be observed that BGAPN achieves better recovery performance. This is due to the fact that it is not restricted to the number of change points in the initial signal and therefore it can use more points and thus adapt itself better to the signal, achieving lower MSE. However, after adding the constraint in the piecewise linear case the optimal projection achieves a better recovery error. The reason is that, as the optimal projection uses the exact number of points, it finds the changing locations more accurately. Note though that in the case of second order polynomial functions, BGAPN gets better recovery. This happens because this program uses the continuity constraint also within its iterations and not only at the final step, as is the case

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 01/08/18 to 132.68.36.165. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

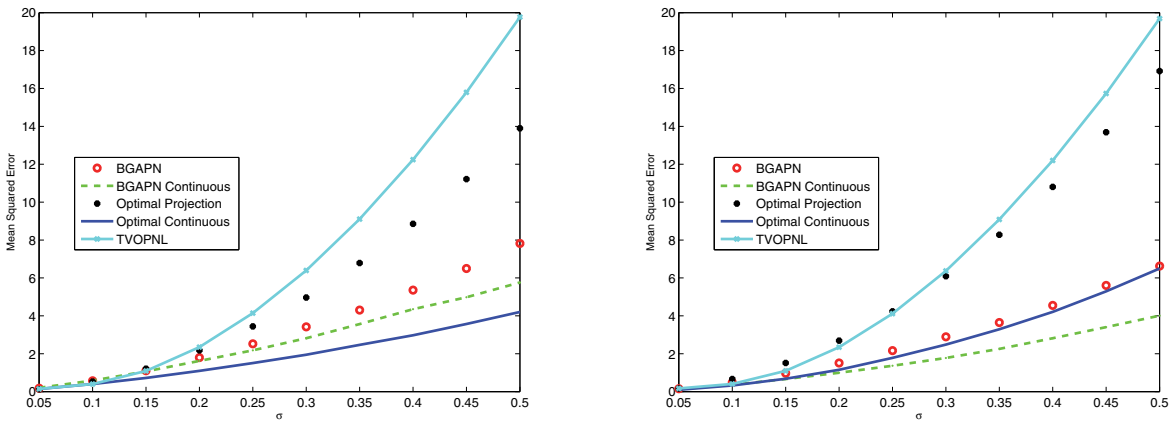


Figure 3. *Mean squared error (MSE) of the recovered piecewise linear functions (left) and piecewise second order polynomial functions (right) as a function of the noise variance σ for the methods BGAPN with and without the continuity constraint and the optimal approximation with and without continuity postprocessing. As a reference we compare this to the TVOPNL approach introduced in* [56].

with the projection algorithm. As the second order polynomial case is more complex than the piecewise linear one, the impact of the usage of the continuity prior is higher and more significant than the information on the number of change points.

We also compare these results to the nonlocal overparameterized TV algorithm (TVOPNL) in [56],[5] which was shown to be better for the task of line segmentation when compared with several alternatives including the ones reported in [49] and [50]. Clearly, our proposed scheme achieves better recovery performance than TVOPNL, demonstrating the supremacy of our line segmentation strategy.

6.2. Compressed sensing of piecewise polynomial functions. We perform also a compressed sensing experiment in which we compare the performance of SSCoSAMP, with the optimal projection, and BGAPN for recovering a second order polynomial function with 6 jumps from a small set of linear measurements. Each entry in the measurement matrix $\mathbf{M}$ is selected from an independently and identically distributed normal distribution and then all columns are normalized to have a unit norm. The polynomial functions are selected as in the previous experiment but with two differences: (i) we omit the continuity constraint and (ii) we normalize the signals to be with a unit norm.

Figure 4 presents the recovery rate (noiseless case $\sigma = 0$) of each program as a function of the number of measurements m. Note that for a very small or large number of samples BGAPN behaves better. However, in the middle range SSCoSaMP achieves a better reconstruction rate. Nonetheless, we may say that their performance is more or less the same.

6.3. Cartoon image denoising. We turn to evaluating the performance of our approach on images. A piecewise smooth model is considered to be a good model for images, and especially to the ones with no texture, i.e., cartoon images [8, 20]. Therefore, we use a

[5]Code provided by the authors.

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 01/08/18 to 132.68.36.165. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

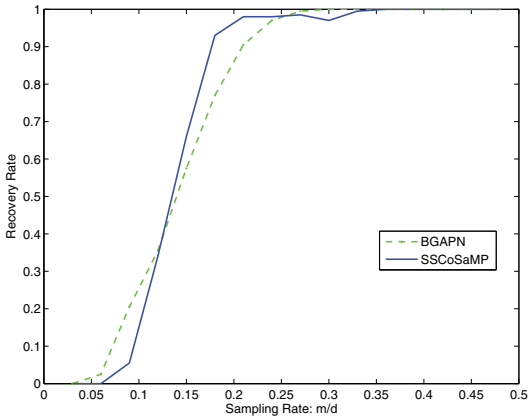


Figure 4. *Recovery rate of piecewise second order polynomial functions as a function of the sampling rate m/d for the methods BGAPN and SSCoSaMP.*

linear overparameterization of the 2D plane and employ the 2D difference operator $\mathbf{\Omega}_{\text{DIF}}$ that calculates the horizontal and vertical discrete derivatives of an image by applying the filters $[1,-1]$ and $[1,-1]^T$ to it. In this case, the problem in (20) turns out to be (notice that $\mathbf{M}=\mathbf{I}$)

$$\left[\hat{\mathbf{b}}_0^T,\hat{\mathbf{b}}_h^T,\hat{\mathbf{b}}_v^T\right]^T = \min_{\tilde{\mathbf{b}}_0,\tilde{\mathbf{b}}_h,\tilde{\mathbf{b}}_v} \left\| \left|\mathbf{\Omega}_{\text{DIF}}\tilde{\mathbf{b}}_0\right|^2 + \left|\mathbf{\Omega}_{\text{DIF}}\tilde{\mathbf{b}}_h\right|^2 + \left|\mathbf{\Omega}_{\text{DIF}}\tilde{\mathbf{b}}_v\right|^2 \right\|_0 \quad \text{s.t.} \quad \left\| \mathbf{g} - [\mathbf{X}_0,\mathbf{X}_h,\mathbf{X}_v] \begin{bmatrix} \tilde{\mathbf{b}}_0 \\ \tilde{\mathbf{b}}_h \\ \tilde{\mathbf{b}}_v \end{bmatrix} \right\|_2 \leq \|\mathbf{e}\|_2 , \tag{22}$$

where $\mathbf{X}_0$, $\mathbf{X}_h,\mathbf{X}_v$ are the matrices that contain the DC, the horizontal, and the vertical parameterizations, respectively, and $\hat{\mathbf{b}}_0,\hat{\mathbf{b}}_h,\hat{\mathbf{b}}_v$ are their corresponding space variables.

We apply this scheme for denoising two cartoon images, *swoosh* and *sign*. We compare our results with the ones of TV denoising [54]. Figures 5 and 6 present the recovery of *swoosh* and *sign* from their noisy version contaminated with an additive white Gaussian noise with $\sigma=20$. Note that we achieve better recovery results than TV and do not suffer from its staircasing effect. We have tuned the parameters of TV separately for each image to optimize its output quality, while we have used the same setup for our method in all the denoising experiments. To get a good quality with BGAPN, we run the algorithm several times with different sets of parameters (which are the same for all images) and then provide as an output the average image of all the runs. Notice that using this technique with TV degrades its results.

To test whether our better denoising is just a result of using overparameterization or an outcome of our new framework, we compare it also to TV with linear overparameterization [49].[6] Notice that while plugging overparameterization directly into TV improves the results in some cases [49], this is not the case with the images here. Therefore, we see that our

[6]Code provided by the authors.

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 01/08/18 to 132.68.36.165. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

(a) Original image (b) Noisy image $\sigma = 20$ (c) BGAPN with $\mathbf{\Omega}_{\mathrm{DIF}}$. PSNR = 40.09 dB

(d) TV recovery. PSNR = 38.95 dB (e) TV OP recovery. PSNR = 37.41 dB

Figure 5. *Denoising of* swoosh *using the BGAPN algorithm with and without diagonal derivatives. Notice that we do not have the staircasing effect that appears in the TV reconstruction. OP stands for overparameterized.*

new framework that links sparsity with overparameterization has an advantage over the old approach that still acts within the variational scheme.

We could use other forms of overparameterization such as cubical instead of planar or add other directions of the derivatives in addition to the horizontal and vertical ones. For example, one may apply our scheme using an operator that also calculates the diagonal derivatives using the filters $\left[\begin{smallmatrix} 1 & 0 \\ 0 & -1 \end{smallmatrix}\right]$ and $\left[\begin{smallmatrix} 0 & 1 \\ -1 & 0 \end{smallmatrix}\right]$. Such choices may lead to an improvement in different scenarios. Future work should focus on learning the overparameterizations and the types of derivatives that should be used for denoising and for other tasks. We believe that such a study has the potential to lead to state-of-the-art results.

6.4. Image segmentation. As a motivation for the task of segmentation we present the denoising of an image with a texture. We continue using the model in (22) and consider the *house* image as an example. Figure 7 demonstrates the denoising result we get for this image. Note that here as well we do not suffer from the staircasing effect that appears in the TV

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 01/08/18 to 132.68.36.165. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

(a) Original image

(b) Noisy image $\sigma = 20$

(c) BGAPN with $\mathbf{\Omega}_{\mathrm{DIF}}$. PSNR = 34.02 dB

(d) TV recovery. PSNR = 33.69 dB

(e) TV OP recovery. PSNR = 31.83 dB

Figure 6. *Denoising of* sign *using the BGAPN algorithm. The results of TV and OP-TV are presented as a reference. OP stands for overparameterized.*

recovery. However, due to the nature of our model we lose the texture and therefore achieve an inferior PSNR compared to the TV denoising.[7]

Though the removal of texture is not favorable for the task of denoising, it makes the recovery of salient edges in the original image easier. In Figure 8 we present the gradient map of our recovered image and the one of the original image. It can be seen that while the gradients of the original image also capture the texture changes, our method finds only the main edges.[8] This motivates us to use our scheme for segmentation.

[7]The lower PSNR we get with our method is because our model is linear and therefore is less capable of adapting itself to the texture. By using a cubic overparameterization we get PSNR which is equal to the one of TV. Note also that for larger noise magnitudes the recovery performance of our algorithm in terms of PSNR becomes better than TV also with the linear model, as in these conditions, we tend to lose the texture anyway.

[8]We have tested our approach also in the presence of a blur operator. The edges in this case are preserved as well.

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 01/08/18 to 132.68.36.165. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

(a) Original image (b) Noisy image $\sigma = 20$ (c) BGAPN with $\mathbf{\Omega}_{\mathrm{DIF}}$. PSNR = 30.77 dB

(d) TV recovery. PSNR = 31.44 dB (e) TV OP recovery. PSNR = 30.6 dB

Figure 7. *Denoising of* house *using the BGAPN algorithm. Notice that we do not have the staircasing effect that appears in the TV reconstruction. Because our model is linear we do not recover the texture and thus we get slightly inferior results compared to TV with respect to PSNR. Note that if we use a cubic overparameterization with BGAPN instead of linear we get PSNR (= 31.81 dB) better than that of TV. OP stands for overparameterized.*

Since our scheme divides the image into piecewise linear regions, we can view our strategy as an approach that minimizes the Mumford–Shah functional [44, 1]. On the other hand, if the image has only two regions, our segmentation result can be viewed as a solution of the Chan–Vese functional with the difference that we model each region by a polynomial function instead of approximating it by a constant [12].

We present our segmentation results for three images, and for each we display the piecewise constant version of each image together with its boundary map. Our segmentation results appear in Figures 9, 10, and 11. We compare our results to the popular graph-cuts based segmentation [25]. Notice that we achieve a comparable performance, where in some places our method behaves better and in others the strategy in [25] provides a better result.

Though we get good segmentation, it is clear that there is still large room for improvement compared to the current state of the art. One direction for improvement is to use more filters

Downloaded 01/08/18 to 132.68.36.165. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

(a) Original image gradients (b) Recovered image gradients

Figure 8. *Gradient map of the clean* house *image and our recovered image from Figure* 7.

within $\mathbf{\Omega}$. Another one is to calculate the gradients of the coefficients parameters and not of the recovered image as they are supposed to be truly piecewise constant. We leave these ideas to future work.

7. Conclusion and future work. This work has presented a novel framework for solving the overparameterized variational problem using sparse representations. We have demonstrated how this framework can be used both for 1D and 2D functions, while a generalization to other higher dimensions (such as three dimensions) is straightforward. We have solved the problem of line fitting for piecewise polynomial 1D signals and then shown how the new technique can be used for compressed sensing, denoising, and segmentation.

Though this work has focused mainly on linear overparameterizations, the extension to other forms is straightforward. However, to keep the discussion as simple as possible, we have chosen to use simple forms of overparameterizations in the experiments section. As a future research, we believe that a learning process should be added to our scheme. It should adapt the functions of the space variables $\mathbf{X}_1, \ldots, \mathbf{X}_n$ and the filters in $\mathbf{\Omega}$ to the signal at hand. We believe that this has the potential to lead to state-of-the-art results in segmentation, denoising, and other signal processing tasks. Combining of our scheme with the standard sparse representation approach may provide the possibility of adding support to images with texture. This will lead to a scheme that works globally on the image for the cartoon part and locally for the texture part. Another route for future work is to integrate our scheme in the state-of-the-art overparameterized based algorithm for optical flow in [50].

Appendix A. The SSCoSaMP algorithm.

For approximating (16), we use a block sparsity variant of SSCoSaMP [36] and adapt it to our model. It is presented in Algorithm 1. Due to the equivalence between $\mathbf{D}_{HS}$ and $\mathbf{\Omega}_{DIF}$, we use the latter in the algorithm.

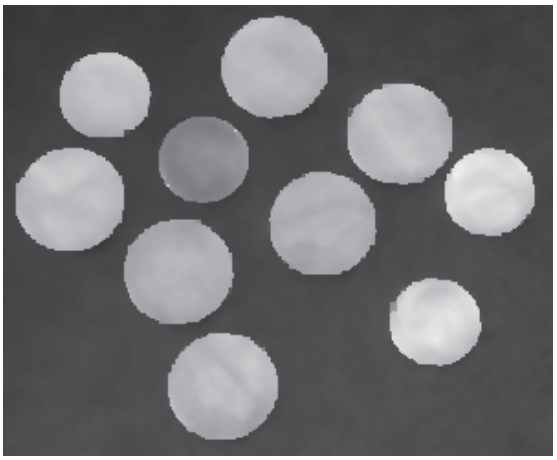
This method uses a projection $S_n(\cdot, k)$ that, given a signal, finds its closest piecewise polynomial functions with k jump points. We calculate this projection using dynamic programming. Our strategy is a generalization of the one that appears in [33, 38] and is presented in the next subsection.

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

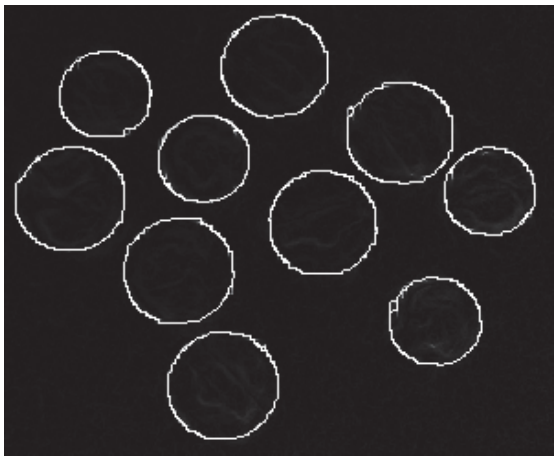
Downloaded 01/08/18 to 132.68.36.165. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

Downloaded 01/08/18 to 132.68.36.165. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

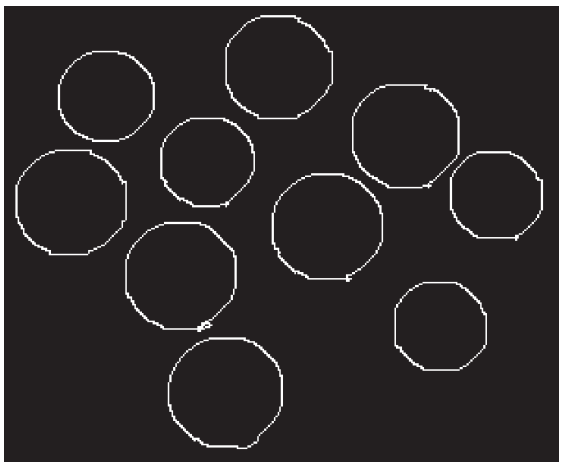
(a) Original image



(b) Piecewise linear version of the image



(c) Image segmentation



(d) Image segmentation using graph cuts [25]

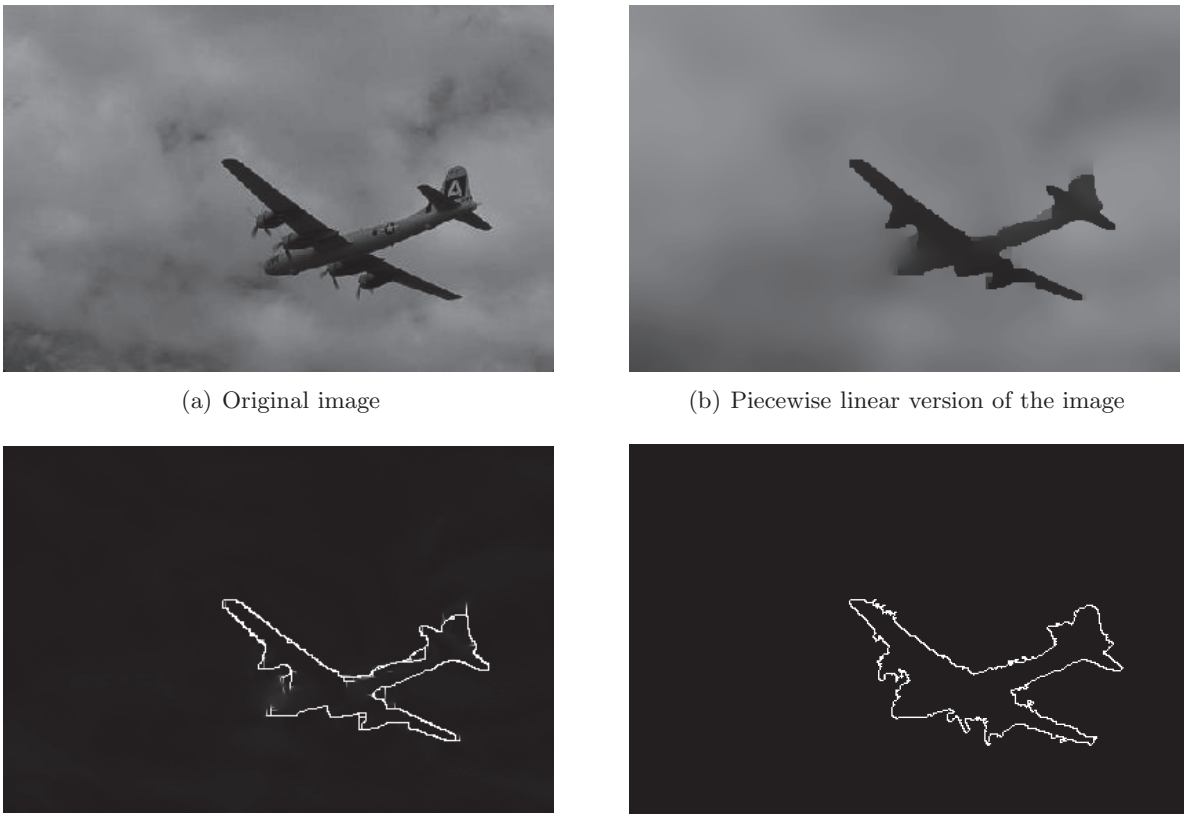
Figure 9. *Piecewise linear version of* coins *image together with the segmentation result. We compare it to the popular graph-cuts based segmentation* [25].

The halting criterion we use in our work in Algorithm 1 is $\left\|\mathbf{g}_r^t\right\|_2 \leq \epsilon$ for a given small constant ϵ. Other options for stopping criteria are discussed in [47].

A.1. Optimal approximation using piecewise polynomial functions. Our projection technique uses the fact that once the jump points are set, the optimal parameters of the polynomial in a segment $[t, l]$ can be calculated optimally by solving a least squares minimization problem

$$\left\| [\mathbf{I}[t,l], \mathbf{Z}[t,l], \ldots, \mathbf{Z}^n[t,l]] \begin{bmatrix} \mathbf{b}_0[t,l] \\ \mathbf{b}_1[t,l] \\ \vdots \\ \mathbf{b}_n[t,l] \end{bmatrix} - \mathbf{g}[t,l] \right\|_2^2, \tag{23}$$

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.



(a) Original image (b) Piecewise linear version of the image

(c) Image segmentation (d) Image segmentation using graph cuts [25]

Figure 10. *Piecewise linear version of* airplane *image together with the segmentation result. We compare it to the popular graph-cuts based segmentation* [25].

where $\mathbf{g}[t,l]$ is the subvector of $\mathbf{g}$ supported by the indices t to l ($t \leq l$) and $\mathbf{Z}^i[t,l]$ is the (square) submatrix of $\mathbf{Z}^i$ corresponding to the indices t to l. We denote by $P_n(\mathbf{g}[t,l])$ the polynomial function we get by solving (23). Indeed, in the case that the size of the segment $[t,l]$ is smaller than the number of parameters, e.g., segment of size one for a linear function, the above minimization problem has infinitely many options for setting the parameters. However, all of them lead to the same result, which is keeping the values of the points in the segment, i.e., having $P_n(\mathbf{g}[t,l]) = \mathbf{g}[t,l]$.

Denote by $S_n(\mathbf{g}[1,\tilde{d}],k)$ the optimal approximation of the signal $\mathbf{g}[1,\tilde{d}]$ by a piecewise polynomial function with k jumps. It can be calculated by solving the following recursive minimization problem

$$\hat{t} = \underset{1\leq t<\tilde{d}}{\operatorname{argmin}} \|S_n(\mathbf{g}[1,t],k-1) - \mathbf{g}[1,t]\|_2^2 + \left\|P_n(\mathbf{g}[t+1,\tilde{d}]) - \mathbf{g}[t+1,\tilde{d}]\right\|_2^2, \tag{24}$$

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 01/08/18 to 132.68.36.165. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

(a) Original image

(b) Piecewise linear version of the image

(c) Image segmentation

(d) Image segmentation using graph cuts [25]

Figure 11. *Piecewise linear version of* man *image together with the segmentation result. We compare it to the popular graph-cuts based segmentation* [25].

and setting

$$S_n(\mathbf{g}[1,\tilde{d}],k) = \begin{bmatrix} S_n(\mathbf{g}[1,\hat{t}],k-1) \\ P_n(\mathbf{g}[\hat{t}+1,\tilde{d}]) \end{bmatrix}. \tag{25}$$

The vectors $S_n(\mathbf{g}[1,t],k-1)$ can be calculated recursively using (24). The recursion ends with the base case $S_n(\mathbf{g}[1,t],0) = P_n(\mathbf{g}[1,t])$.

This leads us to the following algorithm for calculating an optimal approximation for a signal $\mathbf{g}$. Notice that this algorithm provides us also with the parameterization of a piecewise polynomial.

1. Calculate $S_n(\mathbf{g}[1,t],0) = P_n(\mathbf{g}[1,t])$ for $1 \le t \le d$.
2. For $\tilde{k} = 1 : k-1$ do
   - Calculate $S_n(\mathbf{g}[1,\tilde{d}],\tilde{k})$ for $1 \le \tilde{d} \le d$ using (24) and (25).
3. Calculate $S_n(\mathbf{g}[1,d],k)$ using (24) and (25).

Denoting by T the worst case complexity of calculating $P_n(\mathbf{g}[t,l])$ for any pair t,l, we have that the complexity of step (1) is $O(dT)$; of step (2) is $O(kd^2(T+d))$, as the computation of

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 01/08/18 to 132.68.36.165. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

Algorithm 1 Signal Space CoSaMP (SSCoSaMP) for Piecewise Polynomial Functions.

Input: $k, \mathbf{M}, \mathbf{g}, \gamma$, where $\mathbf{g} = \mathbf{M}\mathbf{f} + \mathbf{e}$, $\mathbf{f} = \left[\mathbf{I}, \mathbf{Z}, \mathbf{Z}^2, \dots, \mathbf{Z}^n\right] \left[\mathbf{b}_0^T, \mathbf{b}_1^T, \dots, \mathbf{b}_n^T\right]^T$ is a piecewise polynomial function of order n, $k = \left\|\sum_{i=0}^{n} |\mathbf{\Omega}_{\text{DIF}}\mathbf{b}_i|\right\|_0$ is the number of jumps in the representation coefficients of $\mathbf{f}$, $\mathbf{e}$ is an additive noise, and γ is a parameter of the algorithm. $S_n(\cdot, k)$ is a procedure that approximates a given signal by a piecewise polynomial function of order n with k jumps.

Output: $\hat{\mathbf{f}}$: A piecewise polynomial with $k+1$ segments that approximates $\mathbf{f}$.

- Initialize the jumps' locations $T^0 = \emptyset$, the residual $\mathbf{g}_r^0 = \mathbf{g}$, and set $t = 0$.

while halting criterion is not satisfied **do**

- $t = t + 1$.
- Find the parameterization $\mathbf{b}_{r,0}, \mathbf{b}_{r,1}, \dots, \mathbf{b}_{r,n}$ of the residual's polynomial approximation by calculating $S_n(\mathbf{M}^T\mathbf{g}_r^{t-1}, \gamma k)$.
- Find new temporal jump locations: T_Δ = the support of $\sum_{i=0}^{n} |\mathbf{\Omega}_{\text{DIF}}\mathbf{b}_{r,i}|$.
- Update the jumps locations' indices: $\tilde{T}^t = T^{t-1} \cup T_\Delta$.
- Compute temporal parameters:

$$[\mathbf{b}_{p,0}, \dots, \mathbf{b}_{p,n}] = \text{argmin}_{\tilde{\mathbf{b}}_0,\dots,\tilde{\mathbf{b}}_n} \left\| \mathbf{g} - \mathbf{M}\left[\mathbf{I}, \mathbf{Z}, \mathbf{Z}^2 \dots, \mathbf{Z}^n\right] \begin{bmatrix} \tilde{\mathbf{b}}_0 \\ \tilde{\mathbf{b}}_1 \\ \vdots \\ \tilde{\mathbf{b}}_n \end{bmatrix} \right\|_2^2$$

$$\text{s.t.} \quad (\mathbf{\Omega}_{\text{DIF}}\tilde{\mathbf{b}}_0)_{(\tilde{T}^t)^C} = 0, \ \dots \ (\mathbf{\Omega}_{\text{DIF}}\tilde{\mathbf{b}}_n)_{(\tilde{T}^t)^C} = 0.$$

- Calculate a polynomial approximation of order n: $\mathbf{f}^t = S_n(\left[\mathbf{I}, \mathbf{Z}, \mathbf{Z}^2, \dots, \mathbf{Z}^n\right] \left[\mathbf{b}_{p,0}^T, \dots, \mathbf{b}_{p,n}^T\right]^T, k)$.
- Find new jump locations: T^t = the locations of the jumps in the parameterization of $\mathbf{f}^t$.
- Update the residual: $\mathbf{g}_r^t = \mathbf{g} - \mathbf{M}\mathbf{f}^t$.

end while

- Form final solution $\hat{\mathbf{f}} = \mathbf{f}^t$.

the projection error is of complexity $O(d)$; and of step (3) $O(d(T+d))$. Summing all together we get a total complexity of $O(kd^2(T+d))$ for the algorithm, which is a polynomial complexity since T is polynomial.

Appendix B. The block GAPN algorithm.

For approximating (20), we extend the GAPN technique [45] to block sparsity and adapt it to our model. It is presented in Algorithm 2. Notice that this program, unlike SSCoSaMP, does not assume the knowledge of k or the existence of an optimal projection onto the signals' low dimensional union of subspaces. Note also that it suits a general form of overparameterization and not only 1D piecewise polynomial functions. It is possible to accelerate BGAPN for highly scaled problems by removing from the cosupport several elements at a time instead of one in the *update cosupport* stage.

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 01/08/18 to 132.68.36.165. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

Algorithm 2 The Block GAPN Algorithm.

Input: $\mathbf{M}, \mathbf{g}, \mathbf{\Omega}$, where $\mathbf{g} = \mathbf{M}\mathbf{f} + \mathbf{e}$, $\mathbf{f} = [\mathbf{X}_1, \ldots, \mathbf{X}_n]\left[\mathbf{b}_1^T, \ldots, \mathbf{b}_n^T\right]^T$ such that $\sum_i^n |\mathbf{\Omega}\mathbf{b}_i|$ is sparse, and $\mathbf{e}$ is an additive noise.

Output: $\hat{\mathbf{f}} = [\mathbf{X}_1, \ldots, \mathbf{X}_n]\left[\hat{\mathbf{b}}_1^T, \ldots, \hat{\mathbf{b}}_n^T\right]^T$: an estimate for $\mathbf{f}$ such that $\sum_i^n \left|\mathbf{\Omega}\hat{\mathbf{b}}_i\right|$ is sparse.

Initialize cosupport $\Lambda = \{1, \ldots, p\}$ and set $t = 0$.

while halting criterion is not satisfied **do**

$t = t + 1$.

Calculate a new estimate:

$$\left[\hat{\mathbf{b}}_1^T, \ldots, \hat{\mathbf{b}}_n^T\right]^T = \underset{\tilde{\mathbf{b}}_1, \ldots, \tilde{\mathbf{b}}_n}{\operatorname{argmin}} \sum_{i=1}^{n} \left\|\mathbf{\Omega}_\Lambda \tilde{\mathbf{b}}_i\right\|_2^2 \quad \text{s.t.} \quad \left\|\mathbf{g} - \mathbf{M}\left[\mathbf{X}_1, \ldots \mathbf{X}_n\right] \begin{bmatrix} \tilde{\mathbf{b}}_1 \\ \vdots \\ \tilde{\mathbf{b}}_n \end{bmatrix}\right\|_2 \leq \|\mathbf{e}\|_2 . \tag{26}$$

Update cosupport: $\Lambda = \Lambda \setminus \left\{\operatorname{argmax}_j \sum_{i=1}^n \left\|\mathbf{\Omega}_j \hat{\mathbf{b}}_i\right\|_2^2\right\}$.

end while

Form an estimate for the original signal: $\hat{\mathbf{f}} = [\mathbf{X}_1, \ldots, \mathbf{X}_n]\left[\hat{\mathbf{b}}_1^T, \ldots, \hat{\mathbf{b}}_n^T\right]^T$.

Ideally, we would expect that after several iterations of updating the cosupport in BGAPN we would have $\mathbf{\Omega}_\Lambda \hat{\mathbf{b}}_i = 0$. However, many signals are only nearly cosparse, i.e., have k significantly large values in $\mathbf{\Omega}\mathbf{b}_i$ while the rest are smaller than a small constant ϵ. Therefore, a natural stopping criterion in this case would be to stop when the maximal value in $\left|\mathbf{\Omega}\hat{\mathbf{b}}_i\right|$ is smaller than ϵ. This is the stopping criterion we use throughout this paper for BGAPN. Of course, this is not the only option for a stopping criterion, e.g., one may look at the relative solution change in each iteration or use a constant number of iterations if k is foreknown.

We present also a modified version of BGAPN in Algorithm 3 that imposes a continuity constraint on the change points. This is done by creating a binary diagonal matrix $\mathbf{W} = \operatorname{diag}(w_1, \ldots, w_p)$ such that in each iteration of the program the ith element w_i is 1 if it corresponds to a change point and zero otherwise. This matrix serves as a weights matrix to penalize discontinuity in the change point. This is done by adding the regularizing term L

$$\gamma \left\|\mathbf{W}\mathbf{\Omega}\left[\mathbf{X}_1, \ldots \mathbf{X}_n\right] \begin{bmatrix} \tilde{\mathbf{b}}_1 \\ \vdots \\ \tilde{\mathbf{b}}_n \end{bmatrix}\right\|_2^2$$

to the minimization problem in (26) (27) in Algorithm 3), which leads to the additional step (28) in the modified program.

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 01/08/18 to 132.68.36.165. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

Algorithm 3 The Block GAPN Algorithm with Continuity Constraint.

Input: $\mathbf{M}, \mathbf{g}, \mathbf{\Omega}$, γ, where $\mathbf{g} = \mathbf{M}\mathbf{f} + \mathbf{e}$, $\mathbf{f} = [\mathbf{X}_1, \dots, \mathbf{X}_n] \left[\mathbf{b}_1^T, \dots, \mathbf{b}_n^T\right]^T$ such that $\sum_i^n |\mathbf{\Omega}\mathbf{b}_i|$ is sparse, $\mathbf{e}$ is an additive noise, and γ is a weight for the continuity constraint.

Output: $\hat{\mathbf{f}} = [\mathbf{X}_1, \dots, \mathbf{X}_n] \left[\hat{\mathbf{b}}_1^T, \dots, \hat{\mathbf{b}}_n^T\right]^T$: an estimate for $\mathbf{f}$ such that $\sum_i^n \left|\mathbf{\Omega}\hat{\mathbf{b}}_i\right|$ is sparse.

Initialize cosupport $\Lambda = \{1, \dots, p\}$ and set $t = 0$.

while halting criterion is not satisfied **do**

$t = t + 1$.

Calculate a new estimate:

$$\left[\hat{\mathbf{b}}_1^T, \dots, \hat{\mathbf{b}}_n^T\right]^T = \underset{\tilde{\mathbf{b}}_1, \dots, \tilde{\mathbf{b}}_n}{\operatorname{argmin}} \sum_{i=1}^{n} \left\|\mathbf{\Omega}_\Lambda \tilde{\mathbf{b}}_i\right\|_2^2 \quad \text{s.t.} \quad \left\|\mathbf{g} - \mathbf{M}\left[\mathbf{X}_1, \dots \mathbf{X}_n\right] \begin{bmatrix} \tilde{\mathbf{b}}_1 \\ \vdots \\ \tilde{\mathbf{b}}_n \end{bmatrix}\right\|_2 \leq \|\mathbf{e}\|_2 . \tag{27}$$

Update cosupport: $\Lambda = \Lambda \setminus \left\{\operatorname{argmax}_j \sum_{i=1}^n \left\|\mathbf{\Omega}_j \hat{\mathbf{b}}_i\right\|_2^2\right\}$.

Create weight matrix: $\mathbf{W} = \operatorname{diag}(w_1, \dots, w_p)$, where $w_i = 0$ if $i \in \Lambda$ or $w_i = 1$ otherwise.

Recalculate the estimate:

$$\left[\hat{\mathbf{b}}_1^T, \dots, \hat{\mathbf{b}}_n^T\right]^T = \underset{\tilde{\mathbf{b}}_1, \dots, \tilde{\mathbf{b}}_n}{\operatorname{argmin}} \sum_{i=1}^{n} \left\|\mathbf{\Omega}_\Lambda \tilde{\mathbf{b}}_i\right\|_2^2 + \gamma \left\|\mathbf{W}\mathbf{\Omega}\left[\mathbf{X}_1, \dots \mathbf{X}_n\right] \begin{bmatrix} \tilde{\mathbf{b}}_1 \\ \vdots \\ \tilde{\mathbf{b}}_n \end{bmatrix}\right\|_2^2 \quad \text{s.t.} \quad \left\|\mathbf{g} - \mathbf{M}\left[\mathbf{X}_1, \dots \mathbf{X}_n\right] \begin{bmatrix} \tilde{\mathbf{b}}_1 \\ \vdots \\ \tilde{\mathbf{b}}_n \end{bmatrix}\right\|_2 \leq \|\mathbf{e}\|_2 . \tag{28}$$

end while

Form an estimate for the original signal: $\hat{\mathbf{f}} = [\mathbf{X}_1, \dots, \mathbf{X}_n] \left[\hat{\mathbf{b}}_1^T, \dots, \hat{\mathbf{b}}_n^T\right]^T$.

Acknowledgments. The authors would like to thank Tal Nir and Guy Roseman for providing their code for the experiments. The authors would like to thank the anonymous reviewers for their helpful and constructive comments that greatly contributed to improving this paper.

REFERENCES

[1] L. Ambrosio and V. M. Tortorelli, *Approximation of functional depending on jumps by elliptic functional via t-convergence*, Comm. Pure Appl. Math., 43 (1990), pp. 999–1036.

Downloaded 01/08/18 to 132.68.36.165. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

[2] G. Aubert and P. Kornprobst, *Mathematical Problems in Image Processing: Partial Differential Equations and the Calculus of Variations*, Appl. Math. Sci. 147, Springer, 2006.
[3] R. G. Baraniuk, V. Cevher, M. F. Duarte, and C. Hegde, *Model-based compressive sensing*, in IEEE Trans. Inform. Theory, 56 (2010), pp. 1982–2001.
[4] T. Blumensath and M. E. Davies, *Sampling theorems for signals from the union of finite-dimensional linear subspaces*, IEEE Trans. Inform. Theory., 55 (2009), pp. 1872 –1882.
[5] T. Blumensath and M. E. Davies, *Iterative hard thresholding for compressed sensing*, Appl. Comput. Harmon. Anal., 27 (2009), pp. 265–274.
[6] A. M. Bruckstein, D. L. Donoho, and M. Elad, *From sparse solutions of systems of equations to sparse modeling of signals and images*, SIAM Rev., 51 (2009), pp. 34–81.
[7] E. J. Candès, *Modern statistical estimation via oracle inequalities*, Acta Numer., 15 (2006), pp. 257–325.
[8] E. J. Candès and D. L. Donoho, *Curvelets? A surprisingly effective nonadaptive representation for objects with edges*, in Curves and Surface Fitting: Saint-Malo 99, C. Rabut A. Cohen, and L. Schumaker, eds., Vanderbilt University Press, Nashville, 2000, p. 105–120.
[9] E. J. Candès, Y. C. Eldar, D. Needell, and P. Randall, *Compressed sensing with coherent and redundant dictionaries*, Appl. Comput. Harmon. Anal., 31 (2011), pp. 59–73.
[10] V. Caselles, R. Kimmel, and G. Sapiro, *Geodesic active contours*, Int. J. Comput. Vis., 22 (1997), pp. 61–79.
[11] T. F. Chan and J. Shen, *Image Processing and Analysis*, SIAM, Philadelphia, 2005.
[12] T. F. Chan and L.A. Vese, *Active contours without edges*, IEEE Trans. Image Process., 10 (2001), pp. 266–277.
[13] S. Chen, S. A. Billings, and W. Luo, *Orthogonal least squares methods and their application to non-linear system identification*, Int. J. Control, 50 (1989), pp. 1873–1896.
[14] S. S. Chen, D. L. Donoho, and M. A. Saunders, *Atomic decomposition by basis pursuit*, SIAM J. Sci. Comput., 20 (1998), pp. 33–61.
[15] S. F. Cotter, B. D. Rao, K. Engan, and K. Kreutz-Delgado, *Sparse solutions to linear inverse problems with multiple measurement vectors*, IEEE Trans. Signal Process., 53 (2005), pp. 2477–2488.
[16] W. Dai and O. Milenkovic, *Subspace pursuit for compressive sensing signal reconstruction*, IEEE Trans. Inform. Theory., 55 (2009), pp. 2230 –2249.
[17] M. A. Davenport, D. Needell, and M. B. Wakin, *Signal space CoSaMP for sparse recovery with redundant dictionaries*, IEEE Trans. Inform. Theory, 59 (2013), pp. 6820–6829.
[18] G. Davis, S. Mallat, and M. Avellaneda, *Adaptive time-frequency decompositions*, Opt. Eng., 33 (1994), pp. 2183–2191.
[19] G. Davis, S. Mallat, and M. Avellaneda, *Adaptive greedy approximations*, Constr. Approx., 50 (1997), pp. 57–98.
[20] D. L. Donoho, *Sparse components of images and optimal atomic decompositions*, Constr. Approx., 17 (2001), pp. 353–382.
[21] D. L. Donoho and M. Elad, *On the stability of the basis pursuit in the presence of noise*, Signal Process., 86 (2006), pp. 511–532.
[22] M. Elad, P. Milanfar, and R. Rubinstein, *Analysis versus synthesis in signal priors*, Inverse Problems, 23 (2007), pp. 947–968.
[23] Y. C. Eldar, P. Kuppinger, and H. Bolcskei, *Block-sparse signals: Uncertainty relations and efficient recovery*, in IEEE Trans. Signal Process., 58 (2010), pp. 3042–3054.
[24] Y. C. Eldar and M. Mishali, *Robust recovery of signals from a structured union of subspaces*, in IEEE Trans. Inform. Theory, 55 (2009), pp. 5302–5316.
[25] P. F. Felzenszwalb and D. P. Huttenlocher, *Efficient graph-based image segmentation*, Int. J. Comput. Vis., 59 (2004), pp. 167–181.
[26] M. Fornasier and H. Rauhut, *Recovery algorithms for vector-valued data with joint sparsity constraints*, SIAM J. Numer. Anal., 46 (2008), pp. 577–613.
[27] S. Foucart, *Hard thresholding pursuit: An algorithm for compressive sensing*, SIAM J. Numer. Anal., 49 (2011), pp. 2543–2563.
[28] R. Giryes and M. Elad, *CoSaMP and SP for the cosparse analysis model*, in Proceedings of the 20th European Signal Processing Conference (EUSIPCO-2012), Bucharest, Romania, 2012, IEEE, Piscataway, NJ, 2012, pp. 964–968.

Downloaded 01/08/18 to 132.68.36.165. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

[29] R. Giryes and M. Elad, *RIP-based near-oracle performance guarantees for SP, CoSaMP, and IHT*, IEEE Trans. Signal Process., 60 (2012), pp. 1465–1468.
[30] R. Giryes and M. Elad, *Can we allow linear dependencies in the dictionary in the synthesis framework?*, in IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2013, IEEE, Piscataway, NJ, 2013, pp. 5459–5463.
[31] R. Giryes and M. Elad, *Iterative hard thresholding for signal recovery using near optimal projections*, in Proceedings of the 10th International Conference on Sampling Theory and Applications (SAMPTA), 2013, pp. 212–215.
[32] R. Giryes and M. Elad, *OMP with highly coherent dictionaries*, in Proceedings of the 10th International Conference on Sampling Theory and Applications (SAMPTA), 2013, pp. 9–12.
[33] R. Giryes, S. Nam, M. Elad, R. Gribonval, and M. E. Davies, *Greedy-like algorithms for the cosparse analysis model*, Linear Algebra Appl., 441 (2014), pp. 22–60.
[34] R. Giryes, S. Nam, R. Gribonval, and M. E. Davies, *Iterative cosparse projection algorithms for the recovery of cosparse vectors*, in Proceedings of the 19th European Signal Processing Conference (EUSIPCO-2011), Barcelona, Spain, 2011, IEEE, Piscataway, NJ, 2011, pp. 1460–1464.
[35] R. Giryes and D. Needell, *Greedy signal space methods for incoherence and beyond*, Appl. Comput. Harmon. Anal., 39 (2015), pp. 1–20.
[36] R. Giryes and D. Needell, *Near oracle performance and block analysis of signal space greedy methods*, J. Approx. Theory, 194 (2015), pp. 157–174.
[37] R. Gribonval and M. Nielsen, *Sparse representations in unions of bases*, IEEE Trans. Inform Theory., 49 (2003), pp. 3320–3325.
[38] T. X. Han, S. Kay, and T. S. Huang, *Optimal segmentation of signals and its application to image denoising and boundary feature extraction*, in Proceedings of the IEEE International Conference on Image Processing (ICIP), Vol. 4, 2004, IEEE, Piscataway, NJ, 2004, pp. 2693–2696.
[39] D. L. B. Jupp, *Approximation to data by splines with free knots*, SIAM J. Numer. Anal., 15 (1978), pp. 328–343.
[40] Y. M. Lu and M. N. Do, *A theory for sampling signals from a union of subspaces*, IEEE Trans. Signal Process., 56 (2008), pp. 2334–2345.
[41] S. Mallat and Z. Zhang, *Matching pursuits with time-frequency dictionaries*, IEEE Trans. Signal Process., 41 (1993), pp. 3397–3415.
[42] M. Mishali and Y. C. Eldar, *Reduce and boost: Recovering arbitrary sets of jointly sparse vectors*, IEEE Trans. Signal Process., 56 (2008), pp. 4692–4702.
[43] J. M. Morel and S. Solimini, *Variational Methods in Image Segmentation*, Birkhauser Boston, Cambridge, MA, 1995.
[44] D. Mumford and J. Shah, *Optimal approximations by piecewise smooth functions and associated variational problems*, Comm. Pure Appl. Math., 42 (1989), pp. 577–685.
[45] S. Nam, M. E. Davies, M. Elad, and R. Gribonval, *Recovery of cosparse signals with greedy analysis pursuit in the presence of noise*, in Proceedings of the 4th IEEE International Workshop on Computational Advances in Multi-Sensor Adaptive Processing (CAMSAP), 2011, IEEE, Piscataway, NJ, 2011, pp. 361–364.
[46] S. Nam, M. E. Davies, M. Elad, and R. Gribonval, *The cosparse analysis model and algorithms*, Appl. Comput. Harmon. Anal., 34 (2013), pp. 30–56.
[47] D. Needell and J. A. Tropp, *CoSaMP: Iterative signal recovery from incomplete and inaccurate samples*, Appl. Comput. Harmon. Anal., 26 (2009), pp. 301–321.
[48] D. Needell and R. Ward, *Stable image reconstruction using total variation minimization*, SIAM J. Imaging Sci., 6 (2013), pp. 1035–1058.
[49] T. Nir and A. M. Bruckstein, *On over-parameterized model based TV-denoising*, in Proceedings of the International Symposium on Signals, Circuits, and Systems, ISSCS, 2007, vol. 1, 2007, IEEE, Piscataway, NJ, pp. 1–4.
[50] T. Nir, A. M. Bruckstein, and R. Kimmel, *Over-parameterized variational optical flow*, Int. J. Comput. Vis., 76 (2008), pp. 205–216.
[51] T. Peleg and M. Elad, *Performance guarantees of the thresholding algorithm for the CoSparse analysis model*, IEEE Trans. Inform. Theory, 59 (2013), pp. 1832–1845.
[52] P. Perona and J. Malik, *Scale-space and edge detection using anisotropic diffusion*, IEEE Trans. Pattern Anal. Mach. Intell., 12 (1990), pp. 629–639.

Downloaded 01/08/18 to 132.68.36.165. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

[53] G. Rosman, S. Shem-Tov, D. Bitton, T. Nir, G. Adiv, R. Kimmel, A. Feuer, and A. M. Bruckstein, *Over-parameterized optical flow using a stereoscopic constraint*, in Scale Space and Variational Methods in Computer Vision, A. M. Bruckstein, B. M. Haar Romeny, A. M. Bronstein, and M. M. Bronstein, eds., Lecture Notes in Comput. Sci. 6667, Springer, Berlin, 2012, pp. 761–772.

[54] L. I. Rudin, S. Osher, and E. Fatemi, *Nonlinear total variation based noise removal algorithms*, Phys. D, 60 (1992), pp. 259–268.

[55] J. Savage and K. Chen, *On multigrids for solving a class of improved total variation based staircasing reduction models*, in image processing based on partial differential equations, X.-C. Tai, K.-A. Lie, T. F. Chan, and S. Osher, eds., Math. Vis., Springer, Berlin, 2007, pp. 69–94.

[56] S. Shem-Tov, G. Rosman, G. Adiv, R. Kimmel, and A. M. Bruckstein, *On globally optimal local modeling: from moving least squares to over-parametrization*, in Innovations for Shape Analysis, Springer, Heidelberg, 2013, pp. 379–405.

[57] M. Stojnic, F. Parvaresh, and B. Hassibi, *On the reconstruction of block-sparse signals with an optimal number of measurements*, IEEE Trans. Signal Process., 57 (2009), pp. 3075–3085.

[58] R. Tibshirani, *Regression shrinkage and selection via the lasso*, J. Roy. Statist. Soc. Ser. B, 58 (1996), pp. 267–288.

[59] J. A. Tropp, *Algorithms for simultaneous sparse approximation. Part* II*: Convex relaxation*, Signal Process., 86 (2006), pp. 589–602.

[60] J. A. Tropp, A. C. Gilbert, and M. J. Strauss, *Algorithms for simultaneous sparse approximation. Part* I*: Greedy pursuit*, Signal Process., 86 (2006), pp. 572–588.

[61] J. Weickert, *Anisotropic Diffusion in Image Processing*, Teubner, Stuttgart, Germany, 1998.

[62] J. Weickert, A. Bruhn, T. Brox, and N. Papenberg, *A survey on variational optic flow methods for small displacements*, in Mathematical Models for Registration and Applications to Medical Imaging, Math. Ind. 10, Springer, Berlin, 2006, pp. 103–136.

[63] D. P. Wipf and B. D. Rao, *An empirical Bayesian strategy for solving the simultaneous sparse approximation problem*, IEEE Trans. Signal Process., 55 (2007), pp. 3704–3716.

[64] M. Yuan and Y. Lin, *Model selection and estimation in regression with grouped variables*, J. Roy. Statist. Soc. Ser. B, 68 (2006), pp. 49–67.

Downloaded 01/08/18 to 132.68.36.165. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.