TRACKING SOLUTIONS OF TIME VARYING LINEAR INVERSE PROBLEMS

Martin Kleinsteuber and Simon Hawe
Department of Electrical Engineering and Information Technology, Technische Universität München
Arcistraße 21, Munich, Germany

Keywords: Linear inverse problems, Signal reconstruction, Time varying signal reconstruction, Discretized newton flow.

Abstract: The reconstruction of a signal from only a few measurements, deconvolving, or denoising are only a few interesting signal processing applications that can be formulated as linear inverse problems. Commonly, one overcomes the ill-posedness of such problems by finding solutions which best match some prior assumptions. These are often sparsity assumptions as in the theory of Compressive Sensing. In this paper, we propose a method to track solutions of linear inverse problems. We assume that the corresponding solutions vary smoothly over time. A discretized Newton flow allows to incorporate the time varying information for tracking and predicting the subsequent solution. This prediction requires to solve a linear system of equation, which is in general computationally cheaper than solving a new inverse problem. It may also serve as an additional prior that takes the smooth variation of the solutions into account, or, as an initial guess for the preceding reconstruction. We exemplify our approach with the reconstruction of a compressively sampled synthetic video sequence.

1 INTRODUCTION

Linear inverse problems arise in various signal processing applications like in signal deconvolution, denoising, interpolation, or signal reconstruction from few measurements as in Compressive Sensing. Basically, they aim at computing or reconstructing a signal $\mathbf{s} \in \mathbb{R}^n$ from a set of measurements $\mathbf{y} \in \mathbb{R}^m$. Formally, this measurement process can be written as

$$\mathbf{y} = \mathcal{A}\mathbf{s} + \mathbf{e}, \tag{1}$$

where the vector $\mathbf{e} \in \mathbb{R}^m$ models sampling errors and noise, and $\mathcal{A} \in \mathbb{R}^{m \times n}$ is the measurement matrix. In most interesting cases, this problem is ill-posed because either the exact measurement process and hence $\mathcal{A}$ is unknown as in image deblurring, or the number of observations is much smaller than the dimension of the signal, which is the case in Compressive Sensing. In this paper, we restrict to the latter case where the measurement matrix $\mathcal{A}$ is known.

Prior assumptions on the signal help to overcome the ill-posedness of this problem and to stabilize the solution. They are incorporated in the signal recovery process by the minimization of a suitable function $g\colon \mathbb{R}^n \to \mathbb{R}$, leading to the optimization problem

$$\begin{aligned} &\underset{\mathbf{s}^\star \in \mathbb{R}^n}{\text{minimize}} \quad g(\mathbf{s}^\star) \\ &\text{subject to} \quad \|\mathcal{A}\mathbf{s}^\star - \mathbf{y}\|_2^2 \leq \varepsilon, \end{aligned} \tag{2}$$

where ε is an estimated upper bound on the noise power $\|\mathbf{e}\|_2^2$. We refer to (Elad et al., 2007) for the investigation of two conceptual different approaches, the so called *synthesis* and the *analysis* approach.

One of the most commonly used priors which we will use in our experiments, is based on the fact that many interesting signals have a sparse or compressible representation with respect to some (possibly overcomplete) basis. This means that the entire information about the signal is contained in only a few transform coefficients. A common sparsity measure is the ℓ_p-(pseudo-) norm

$$\|\mathbf{v}\|_p^p := \sum_i |\mathbf{v}(i)|^p, \quad 0 \leq p \leq 1, \tag{3}$$

where here and throughout the paper, $\mathbf{v}(i)$ denotes the i^{th} entry of the vector $\mathbf{v}$. Furthermore, let $\mathcal{D}$ be a linear transformation such that $\mathcal{D}\mathbf{s}$ is sparse. In that case $g(\mathbf{s}^\star) = \|\mathcal{D}\mathbf{s}^\star\|_p^p$ is a frequently used regularization term for problem (2). For example, $\mathcal{D}$ may serve as a basis transformation as in classical Compressive Sensing, (Candès and Romberg, 2007; Donoho,

Kleinsteuber M. and Hawe S. (2012).
TRACKING SOLUTIONS OF TIME VARYING LINEAR INVERSE PROBLEMS.
In *Proceedings of the 1st International Conference on Pattern Recognition Applications and Methods*, pages 253-257
DOI: 10.5220/0003864102530257

2006) or as a discretized form of the total variation (Combettes and Pesquet, 2004; Rudin et al., 1992).

Many approaches to tackle problem (2) rely on the convexity of g, like NESTA (Becker et al., 2009), conjugate subgradient methods (Hawe et al., 2011) or TwIST (Bioucas-Dias and Figueiredo, 2007), just to mention a few. That is the reason why the ℓ_1-norm is most commonly employed. Although this convex relaxation leads to perfect signal recovery under certain assumptions, cf. (Donoho and Elad, 2003), it has been shown in (Chartrand and Staneva, 2008) that in some cases, the ℓ_p-norm for $0 \leq p < 1$ severely outperforms its convex counterpart.

In this work, we do not assume convexity of g but we require differentiability. For the ℓ_p-norm, this can easily be achieved by a suitable smooth approximation. Here, we propose an approach based on minimizing the unconstrained Lagrangian form of (2) that is given by

$$\underset{\mathbf{s}^\star \in \mathbb{R}^n}{\text{minimize}} \quad f(\mathbf{s}^\star) = \tfrac{1}{2}\|\mathcal{A}\mathbf{s}^\star - \mathbf{y}\|_2^2 + \lambda g(\mathbf{s}^\star). \tag{4}$$

The Lagrange multiplier $\lambda \in \mathbb{R}_0^+$ weighs between the sparsity of the solution and its fidelity to the acquired samples according to $\lambda \sim \varepsilon$.

Consider now a sequence of linear inverse problems whose solutions vary smoothly over time. As an example, one may think of the denoising of short video sequences or the reconstruction of compressively sensed magnetic resonance image sequences, cf. (Lustig et al., 2007). In this work, we propose an approach to track the solutions of time varying linear inverse problems. We employ preceding solutions to predict the current signal's estimate. To the best of the authors' knowledge, this idea has not been pursued so far in the literature. The crucial idea is to use a discretized Newton flow to track solutions of a time varying version of (4). We provide three practical update formulas for the tracking problem and conclude with a proof of concept by applying our approach to a short synthetic video sequence, where each video frame is recovered from compressively sampled measurements.

2 TRACKING THE SOLUTIONS

2.1 Problem Statement

Let $t \mapsto \mathbf{s}(t) \in \mathbb{R}^n$ be a C^1-curve i.e. with continuous first derivative that represents a time varying signal $\mathbf{s}$. Moreover, let $\mathbf{y}(t) = \mathcal{A}\mathbf{s}(t)$ be the measurements of $\mathbf{s}$ at time t. In this paper, we consider the problem of reconstructing a sequence of signals $\big(\mathbf{s}(t_k)\big)_{k\in\mathbb{N}}$ at consecutive instances of time. Instead of estimating $\mathbf{s}(t_k)$ by solving the inverse problem based on the measurements $\mathbf{y}(t_k)$, we investigate in how far the previously recovered estimates $\mathbf{s}_i^\star$ of $\mathbf{s}(t_i)$, $i = 1,\ldots,k$ can be employed to *predict* $\mathbf{s}(t_{k+1})$ without acquiring new measurements $\mathbf{y}(t_{k+1})$. This prediction step may serve as an intermediate replacement for this reconstruction step or it may be employed as an initialization for reconstruction at time t_k. Note that in our approach, we assume a fixed measurement matrix $\mathcal{A}$.

Consider the time variant version of the unconstrained Lagrangian function

$$f(\mathbf{s}^\star, t) = \tfrac{1}{2}\|\mathcal{A}\mathbf{s}^\star - \mathbf{y}(t)\|_2^2 + \lambda g(\mathbf{s}^\star). \tag{5}$$

The minimum of (5) at time t necessarily yields the gradient

$$F(\mathbf{s}^\star, t) := \frac{\partial}{\partial \mathbf{s}^\star} f(\mathbf{s}^\star, t) \tag{6}$$

to be zero. Consequently, we want to find the smooth curve $\mathbf{s}^\star(t)$ such that

$$F(\mathbf{s}^\star(t), t) = 0. \tag{7}$$

In other words, we want to track the minima of (5) over time. A discretized Newton flow, which is explained in the following subsection, will be used for that purpose.

2.2 Discretized Newton Flow

Homotopy methods are a well-known approach for solving problem (7). These methods are based on an associated differential equation whose solutions track the roots of F. To make the paper self contained, we shortly rederive the discretized Newton flow for our situation at hand based on (Baumann et al., 2005). Specifically, we consider the implicit differential equation

$$\mathcal{J}_F(\mathbf{s}^\star, t)\dot{\mathbf{s}}^\star + \frac{\partial}{\partial t} F(\mathbf{s}^\star, t) = -\alpha F(\mathbf{s}^\star, t), \tag{8}$$

where $\alpha > 0$ is a free parameter that stabilizes the dynamics around the desired solution. Here,

$$\mathcal{J}_F(\mathbf{s}^\star, t) := \frac{\partial}{\partial \mathbf{s}^\star} F(\mathbf{s}^\star, t) \tag{9}$$

is the $(n \times n)$-matrix of partial derivatives of F with respect to $\mathbf{s}^\star$. Under suitable invertibility conditions on $\mathcal{J}_F$, we rewrite (8) in explicit form as

$$\dot{\mathbf{s}}^\star = -\mathcal{J}_F(\mathbf{s}^\star, t)^{-1} \left(\alpha F(\mathbf{s}^\star, t) + \frac{\partial}{\partial t} F(\mathbf{s}^\star, t) \right). \tag{10}$$

We discretize (10) at time instances t_k, for $k \in \mathbb{N}$ and assume without loss of generality a fixed stepsize $h >$

0. Depending on the stepsize we choose $\alpha := \frac{1}{h}$. With the shorthand notation for $\mathbf{s}_k^\star := \mathbf{s}^\star(t_k)$, the single-step Euler discretization of the time-varying Newton flow is therefore given as

$$\mathbf{s}_{k+1}^\star = \mathbf{s}_k^\star - \mathcal{J}_F(\mathbf{s}_k^\star, t_k)^{-1}\left(F(\mathbf{s}_k^\star, t_k) + h\frac{\partial F}{\partial t}(\mathbf{s}_k^\star, t_k)\right). \quad (11)$$

We approximate the partial derivative $\frac{\partial F}{\partial t}(\mathbf{s}_k^\star, t_k)$ by an m^{th}-order Taylor approximation $H_m(\mathbf{s}^\star, t)$. For the practically interesting case these are, cf. (Baumann et al., 2005)

$$H_1(\mathbf{s}^\star, t) = \tfrac{1}{h}\Big(F(\mathbf{s}^\star, t) - F(\mathbf{s}^\star, t-h)\Big) \quad (12)$$

$$H_2(\mathbf{s}^\star, t) = \tfrac{1}{2h}\Big(3F(\mathbf{s}^\star, t) - 4F(\mathbf{s}^\star, t-h) + F(\mathbf{s}^\star, t-2h)\Big) \quad (13)$$

$$H_3(\mathbf{s}^\star, t) = \tfrac{1}{30h}\Big(37F(\mathbf{s}^\star, t) - 45F(\mathbf{s}^\star, t-h) + 9F(\mathbf{s}^\star, t-2h) - F(\mathbf{s}^\star, t-3h)\Big) \quad (14)$$

These approximations turn (11) into the update formula

$$\mathbf{s}_{k+1}^\star = \mathbf{s}_k^\star - \mathcal{J}_F(\mathbf{s}_k^\star, t_k)^{-1}\Big(F(\mathbf{s}_k^\star, t_k) + hH_m(\mathbf{s}_k^\star, t_k)\Big). \quad (15)$$

2.3 Explicit Update Formulas

In this subsection we derive concrete update formulas for tracking the solution of inverse problems as defined in (5).

Often the inverse $\mathcal{J}_F(\mathbf{s}_k^\star, t_k)^{-1}$ is not accessible or infeasible to calculate, in particular when dealing with high dimensional data. Hence for computing the estimate $\mathbf{s}_{k+1}^*$ as in equation (15), we solve

$$\underset{\mathbf{s}\in\mathbb{R}^n}{\text{minimize}} \quad \|\mathcal{J}_F(\mathbf{s}_k^\star, t_k)\mathbf{s} - \mathbf{b}_m(\mathbf{s}_k^\star, t_k)\|_2^2, \quad (16)$$

with

$$\mathbf{b}_m(\mathbf{s}_k^\star, t_k) := \mathcal{J}_F(\mathbf{s}_k^\star, t_k)\mathbf{s}_k^\star - \Big(F(\mathbf{s}_k^\star, t_k) + hH_m(\mathbf{s}_k^\star, t_k)\Big). \quad (17)$$

Typically, linear Conjugate Gradient methods efficiently solve this linear equation, (Nocedal and Wright, 2006). Note, that this is less computational expensive compared to solving an individual reconstruction by minimizing (4).

We now derive three explicit update schemes for the concrete problem of tracking solutions to inverse problems based on the approximations (12)-(14).

Let ∇g denote the gradient of g. Then from (5) we obtain

$$F(\mathbf{s}^\star, t) = \mathcal{A}^\top(\mathcal{A}\mathbf{s}^\star - \mathbf{y}(t)) + \lambda\nabla g(\mathbf{s}^\star). \quad (18)$$

The derivative of F with respect to $\mathbf{s}^\star$ is thus

$$\mathcal{J}_F(\mathbf{s}^\star, t) = \mathcal{A}^\top\mathcal{A} + \lambda\mathcal{H}_g(\mathbf{s}^\star), \quad (19)$$

where $\mathcal{H}_g(\mathbf{s}^\star)$ denotes the Hessian of g at $\mathbf{s}^\star$. Combing equation (18) with (12)-(14) yields

$$hH_1(\mathbf{s}^\star, t) = \mathcal{A}^\top\Big(\mathbf{y}(t-h) - \mathbf{y}(t)\Big) \quad (20)$$

$$hH_2(\mathbf{s}^\star, t) = \tfrac{1}{2}\mathcal{A}^\top\Big(4\mathbf{y}(t-h) - 3\mathbf{y}(t) - \mathbf{y}(t-2h)\Big) \quad (21)$$

$$hH_3(\mathbf{s}^\star, t) = \tfrac{1}{30}\mathcal{A}^\top\Big(45\mathbf{y}(t-h) - 37\mathbf{y}(t) - 9\mathbf{y}(t-2h) + \mathbf{y}(t-3h)\Big). \quad (22)$$

This results in the explicit formulas for $\mathbf{b}_1, \mathbf{b}_2, \mathbf{b}_3$

$$\mathbf{b}_1(\mathbf{s}_k^\star, t_k) = \lambda\Big(\mathcal{H}_g(\mathbf{s}_k^\star)\mathbf{s}_k^\star - \nabla g(\mathbf{s}_k^\star)\Big) + \mathcal{A}^\top(2\mathbf{y}(t_k) + \mathbf{y}(t_{k-1})\Big) \quad (23)$$

$$\mathbf{b}_2(\mathbf{s}_k^\star, t_k) = \lambda\Big(\mathcal{H}_g(\mathbf{s}_k^\star)\mathbf{s}_k^\star - \nabla g(\mathbf{s}_k^\star)\Big) + \tfrac{1}{2}\mathcal{A}^\top\Big(5\mathbf{y}(t_k) - 4\mathbf{y}(t_{k-1}) + \mathbf{y}(t_{k-2})\Big) \quad (24)$$

$$\mathbf{b}_3(\mathbf{s}_k^\star, t_k) = \lambda\Big(\mathcal{H}_g(\mathbf{s}_k^\star)\mathbf{s}_k^\star - \nabla g(\mathbf{s}_k^\star)\Big) + \tfrac{1}{30}\mathcal{A}^\top\Big(67\mathbf{y}(t_k) - 45\mathbf{y}(t_{k-1}) + 9\mathbf{y}(t_{k-2}) - \mathbf{y}(t_{k-3})\Big). \quad (25)$$

The three different explicit update formulas follow straightforwardly

$$\mathbf{s}_{k+1}^\star = \underset{\mathbf{s}\in\mathbb{R}^n}{\arg\min}\, \|\mathcal{J}_F(\mathbf{s}_k^\star, t_k)\mathbf{s} - \mathbf{b}_m(\mathbf{s}_k^\star, t_k)\|_2^2. \quad (26)$$

3 EXPERIMENTS

In this section we provide an example as a proof of concept of our proposed algorithm. It consists of tracking the reconstruction result of a series of compressively sampled time varying images $\mathbf{s}(t_k) \in \mathbb{R}^n$. The images are created synthetically and show a ball moving with constant velocity, see Figure 1. To enhance legibility, all formulas are expressed in terms of matrix vector products. However, regarding the implementation, we want to emphasize that filtering techniques are used to deal with the large image data.

Considering the measurement matrix $\mathcal{A}$, we take $m \ll n$ randomly selected coefficients of the Rudin-Shapiro transformation (RST) (Benke, 1994). The

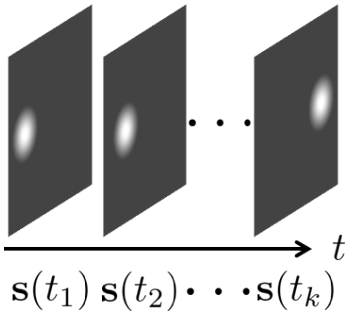


Figure 1: Time sequence of synthetic test image.

RST, also know as the real valued Dragon-Noiselet-transformation, is used because of its efficient implementation and due to its desirable properties for image reconstruction (Romberg, 2008). We empirically set the number of measurements to $m = 0.2n$. In our experiments we found that the number of measurements does not severely affect the accuracy of the tracking algorithm, but the speed of convergence. The larger we chose m the faster the algorithm converges.

Regarding the regularization term g, we exploit the fact that most images have a sparse gradient. The simplest way of approximating the image gradient is in terms of finite differences between neighboring pixels in horizontal, and vertical direction respectively. The computation of the finite differences can be formulated as $\mathcal{D}\mathbf{s}$ with a suitable matrix $\mathcal{D} \in \mathbb{R}^{2n\times n}$.

To measure the sparsity of the gradient, we employ a smooth approximation of the ℓ_p-pseudo-norm of $\mathcal{D}\mathbf{s}$ defined as

$$g(\mathbf{s}) = \sum_{i=1}^{2n} \left((\mathbf{e}_i \mathcal{D}\mathbf{s})^2 - \mu\right)^{\frac{p}{2}}, \qquad (27)$$

with $0 < p \leq 1$ and a smoothing parameter $\mu \in \mathbb{R}^+$. The vector $\mathbf{e}_i \in \mathbb{R}^{2n}$ is a unit vector where the i^{th} component is equal to one and the others are zero. For $p = 1$, equation (27) is a smooth approximation of the well known anisotropic total-variation pseudo-norm.

We start our tracking algorithm by measuring RST coefficients at consecutive instances of time $y(t_k) = \mathcal{A}\mathbf{s}(t_k)$. From these consecutive measurements we find $\mathbf{s}^\star_k$ by individually solving (5) using a Conjugate Gradient (CG) method with backtracking line-search and Hestenes-Stiefel update rule (Nocedal and Wright, 2006). Explicitly, the gradient of the proposed regularizer (27) is

$$\nabla g(\mathbf{s}) = \mathcal{D}^\top \sum_{i=1}^{2n} p\, \mathcal{E}_i \left((\mathbf{e}_i^\top \mathcal{D}\mathbf{s})^2 + \mu\right)^{\frac{p}{2}-1} \mathcal{D}\mathbf{s}, \quad (28)$$

where $\mathcal{E}_i := \mathbf{e}_i \mathbf{e}_i^\top$. The Hessian of g is given by

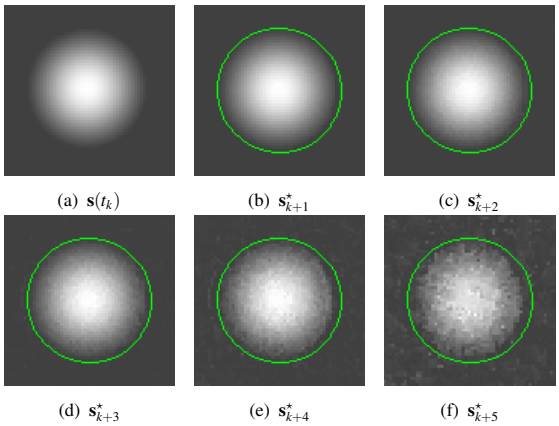
Table 1: PSNR in dB and MSE between estimated signal $\mathbf{s}^\star_{k+j}$ for $j = 1,\ldots,5$ and original signals $\mathbf{s}(t_{k+j})$ $j = 1,\ldots,5$.

	$\mathbf{s}^\star_{k+1}$	$\mathbf{s}^\star_{k+2}$	$\mathbf{s}^\star_{k+3}$	$\mathbf{s}^\star_{k+4}$	$\mathbf{s}^\star_{k+5}$
PSNR	57.2	51.5	34.9	33.3	29.0
MSE	0.12	0.45	20.8	30.3	80.2

$$\begin{aligned}\mathcal{H}_g(\mathbf{s}) = \mathcal{D}^\top \sum_{i=1}^{2n} p\mathcal{E}_i \Big(\left((\mathbf{e}_i^\top \mathcal{D}\mathbf{s})^2 + \mu\right)^{\frac{p}{2}-1} \\ + (p-2)\left((\mathbf{e}_i^\top \mathcal{D}\mathbf{s})^2 + \mu\right)^{\frac{p}{2}-2} (\mathbf{e}_i^\top \mathcal{D}\mathbf{s})^2 \Big) \mathcal{D} \\ =: \mathcal{D}^\top Q \mathcal{D}. \qquad (29)\end{aligned}$$

From this, we obtain $\mathbf{s}^\star_{k+1}$ by (26), using a linear CG-method. Regarding the update formula for $\mathbf{b}_m$, we found in our experiments that (24) yields a good trade-off between prediction results and computational burden.

The tracking results for our example are presented in Figure 2(b)-(f) for $p = 0.7$. We use the knowledge of $\mathbf{s}(t_k)$, $\mathbf{s}(t_{k-1})$ and $\mathbf{s}(t_{k-2})$ to iteratively estimate $\mathbf{s}^\star_{k+j}$ for $j = 1,\ldots,5$ only based on the update formula (26). Clearly, the smaller j is, the better the estimation. Note that the results shown in Figure 2(e)-(f) are solely based on previously *predicted* images. The green circle indicates the position of the ball in the original images $\mathbf{s}(t_{k+j})$, $j = 1,\ldots,5$. It can be seen that although the quality of the images decreases, the position of the circle is still captured adequately. As a quantitative measure of the reconstruction quality, Table 1 contains the peak signal to noise ratio (PSNR) and the mean squared error (MSE) of the estimated signals to the original signals.



(a) $\mathbf{s}(t_k)$ (b) $\mathbf{s}^\star_{k+1}$ (c) $\mathbf{s}^\star_{k+2}$

(d) $\mathbf{s}^\star_{k+3}$ (e) $\mathbf{s}^\star_{k+4}$ (f) $\mathbf{s}^\star_{k+5}$

Figure 2: Excerpt of original image (a) and estimated images (b)-(f). The green circle indicates the position of the ball in the original images.

A final word on the computational cost of the algorithm. Considering the cost for applying the Hessian operator as defined in (29), it mainly depends on

the cost of applying $\mathcal{D}$ and its transposed, as the matrix Q is just a diagonal operator that can be applied in $O(n)$ flops. Furthermore, we want to mention that *any* sparsifying transformation $\mathcal{D}$ that admits a fast implementation e.g. the Wavelet or Curvelet transformation for images, can be easily used within this framework.

4 CONCLUSIONS

In this paper we present a concept for tracking the solutions of inverse problems that vary smoothly over time. The tracking is achieved by employing a discretized Newton flow on the gradient of the cost function. The approach allows us to predict the signal at the next time instance from previous reconstruction results without explicitly taking new measurements. One advantage is that this prediction step is computationally less expensive than an individual reconstruction.

REFERENCES

Baumann, M., Helmke, U., and Manton, J. (2005). Reliable tracking algorithms for principal and minor eigenvector computations. In *44th IEEE Conference on Decision and Control and European Control Conference*, pages 7258–7263.

Becker, S., Bobin, J., and Candès, E. J. (2009). Nesta: a fast and accurate first-order method for sparse recovery. *SIAM Journal on Imaging Sciences*, 4(1):1–39.

Benke, G. (1994). Generalized rudin-shapiro systems. *Journal of Fourier Analysis and Applications*, 1(1):87–101.

Bioucas-Dias, J. and Figueiredo, M. (2007). A new twist: Two-step iterative shrinkage/thresholding algorithms for image restoration. *Image Processing, IEEE Transactions on*, 16(12):2992 –3004.

Candès, E. J. and Romberg, J. (2007). Sparsity and incoherence in compressive sampling. *Inverse Problems*, 23(3):969–985.

Chartrand, R. and Staneva, V. (2008). Restricted isometry properties and nonconvex compressive sensing. *Inverse Problems*, 24(3):1–14.

Combettes, P. L. and Pesquet, J. C. (2004). Image restoration subject to a total variation constraint. *IEEE Transactions on Image Processing*, 13(9):1213–1222.

Donoho, D. L. (2006). Compressed sensing. *IEEE Transactions on Information Theory*, 52(4):1289–1306.

Donoho, D. L. and Elad, M. (2003). Optimally sparse representation in general (nonorthogonal) dictionaries via ℓ_1 minimization. *Proceedings of the National Academy of Sciences of the United States of America*, 100(5):2197–2202.

Elad, M., Milanfar, P., and Rubinstein, R. (2007). Analysis versus synthesis in signal priors. *Inverse Problems*, 3(3):947–968.

Hawe, S., Kleinsteuber, M., and Diepold, K. (2011). Dense disparity maps from sparse disparity measurements. In *IEEE 13th International Conference on Computer Vision*.

Lustig, M., Donoho, D., and Pauly, J. M. (2007). Sparse MRI: The application of compressed sensing for rapid MR imaging. *Magnetic Resonance in Medicine*, 58(6):1182–1195.

Nocedal, J. and Wright, S. J. (2006). *Numerical Optimization, 2nd Ed.* Springer, New York.

Romberg, J. (2008). Imaging via compressive sampling. *IEEE Signal Processing Magazine*, 25(2):14–20.

Rudin, L. I., Osher, S., and Fatemi, E. (1992). Nonlinear total variation based noise removal algorithms. *Phys. D*, 60(1-4):259–268.