Electronic Transactions on Numerical Analysis.
Volume 58, pp. 486–516, 2023.

ISSN 1068–9613.
DOI: 10.1553/etna_vol58s486

ETNA
Kent State University and
Johann Radon Institute (RICAM)

A COMPUTATIONAL FRAMEWORK FOR EDGE-PRESERVING REGULARIZATION IN DYNAMIC INVERSE PROBLEMS*

MIRJETA PASHA†, ARVIND K. SAIBABA‡, SILVIA GAZZOLA§, MALENA I. ESPAÑOL¶, AND ERIC DE STURLER‖

Abstract. We devise efficient methods for dynamic inverse problems, where both the quantities of interest and the forward operator (measurement process) may change in time. Our goal is to solve for all the quantities of interest simultaneously. We consider large-scale ill-posed problems made more challenging by their dynamic nature and, possibly, by the limited amount of available data per measurement step. To alleviate these difficulties, we apply a unified class of regularization methods that enforce simultaneous regularization in space and time (such as edge enhancement at each time instant and proximity at consecutive time instants) and achieve this with low computational cost and enhanced accuracy. More precisely, we develop iterative methods based on a majorization-minimization (MM) strategy with quadratic tangent majorant, which allows the resulting least-squares problem with a total variation regularization term to be solved with a generalized Krylov subspace (GKS) method; the regularization parameter can be determined automatically and efficiently at each iteration. Numerical examples from a wide range of applications, such as limited-angle computerized tomography (CT), space-time image deblurring, and photoacoustic tomography (PAT), illustrate the effectiveness of the described approaches.

Key words. dynamic inversion, time-dependence, edge-preservation, majorization-minimization, regularization, generalized Krylov subspaces, image deblurring, photoacoustic tomography, computerized tomography

AMS subject classifications. 65F10, 65F22, 65F50

1. Introduction.

In the classical setting, inverse problems are commonly formulated as static, where the underlying parameters that define the problem do not change during the measurement process. There exists a very rich literature and many numerical methods for this setting; see [27, 37, 43, 55, 64] and the references therein. Dynamic inverse problems, where time-dependent information needs to be recovered from time-dependent data, have recently gained considerable attention because of new developments in science and engineering applications. Important examples include dynamical impedance tomography [61, 62], process tomography [68], undersampled dynamic X-ray tomography [15], and passive seismic tomography [67, 73], to mention a few. A common objective is to improve the reconstruction of non-stationary objects using time-dependent projection measurements. For instance, the movement of objects during a CT scan leads to artifacts in the stationary reconstruction, even if the change in time is small. More specifically, in the imaging of organs like the heart and lungs, small changes in shape due to the heartbeat or breathing can significantly affect the quality of the reconstructed solution. In [1, 8, 50], approaches for reconstructing a static image from dynamic data are discussed. In [15], the authors discuss the reconstruction of dynamic data in space and time. Computationally feasible methods in the Bayesian framework for dynamic inverse problems are presented in [23], and the quantification of the uncertainties is discussed in [60]. In this work, we are interested in similar scenarios where the target of interest changes in space and time; our approach is not limited to any specific motion of the objects during the measurement process. Furthermore, we seek to preserve the edges of the desired solution. Edge-preserving reconstruction is a technique to smooth images while

*Received May 25, 2023. Accepted June 6, 2023. Published online on July 17, 2023. Recommended by Lothar Reichel.
†Department of Mathematics, Tufts University, USA (mirjeta.pasha@tufts.edu).
‡Department of Mathematics, North Carolina State University, USA (asaibab@ncsu.edu).
§Department of Mathematical Sciences, University of Bath, UK (s.gazzola@bath.ac.uk).
¶School of Mathematical and Statistical Sciences, Arizona State University, USA (malena.espanol@asu.edu).
‖Department of Mathematics, Virginia Tech, USA (sturler@vt.edu).

ETNA
Kent State University and
Johann Radon Institute (RICAM)

preserving edges, which has been employed in many fundamental applications in image processing such as artifact removal [71], denoising [36, 59, 65], image segmentation [24, 39], and feature selection [72]. The proposed methods rely on total variation (TV)-type regularization. While there has been considerable work on edge-preserving methods, only a few contributions address edge-preserving methods for dynamic inverse problems. These have been developed mostly in recent years, highlighting the need for better methods to handle advances in science and technology. See the *Related work* paragraph of Section 1.2 for comparisons with other work.

1.1. Background on dynamic inverse problems. First, we define some notation. Let $\mathbf{U}^{(t)} \in \mathbb{R}^{n_v \times n_h}$ be the 2D (matrix) representation of an image with n_v rows and n_h columns obtained at time instance $t = 1, 2, \ldots, n_t$. Let $\mathbf{u}^{(t)}$ be the column vector obtained by a lexicographical ordering of the two-dimensional $\mathbf{U}^{(t)}$, that is, $\mathbf{u}^{(t)} = \mathrm{vec}(\mathbf{U}^{(t)}) \in \mathbb{R}^{n_s}$, with vec being the operation that vectorizes a matrix by stacking its columns and $n_s = n_v n_h$. Then, let $\mathbf{U} = \left[\mathbf{u}^{(1)}, \ldots, \mathbf{u}^{(n_t)}\right] \in \mathbb{R}^{n_s \times n_t}$ be such that $\mathbf{u} = \mathrm{vec}(\mathbf{U}) \in \mathbb{R}^n$ and $n = n_s n_t$. A pictorial representation of these quantities is displayed in Figure 1.1.

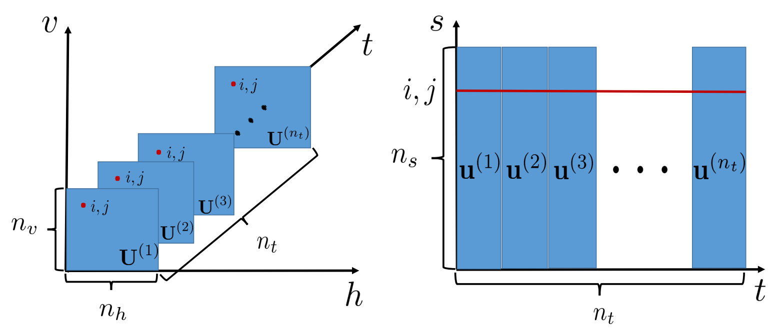


FIGURE 1.1. *Images $\mathbf{U}^{(t)}$ to be reconstructed with pixels i, j in red (left), and their corresponding vectorization $\mathbf{u}^{(t)}$, which are the columns of the matrix $\mathbf{U}$ where the pixels i, j are now in the same row (right).*

We are interested in solving inverse problems in space and time with an unknown target of interest. The goal is to recover from the available measurements $\mathbf{d}^{(t)} \in \mathbb{R}^{m_t}$, for $t = 1, 2, \ldots, n_t$, the images $\mathbf{u}^{(t)} \in \mathbb{R}^{n_s}$, whose entries represent pixels in the image. Since we focus on imaging applications, we use the term 'pixels' (rather than 'parameters') throughout the paper. Given the number of time points n_t, $m = \sum_{t=1}^{n_t} m_t$ is the total number of available measurements. We consider the number of pixels, n_s, to be fixed for all time points. Dynamic problems may also involve reconstructing a sequence of images with varying numbers of pixels (e.g., in image registration), but we do not consider that setting in this paper. For completeness, we define static and dynamic inverse problems in the context of this paper.

Dynamic inverse problems. In a dynamic inverse problem, both the images of interest and the measurement process are known to change in time. Therefore, combining prior information at different time instances enhances the reconstruction and recovery of dynamic information about the objects of interest. More specifically, we have the measurement equation

$$\mathbf{d} = \mathbf{F}\mathbf{u} + \mathbf{e}, \tag{1.1}$$

where we consider two cases for the forward operator $\mathbf{F} \in \mathbb{R}^{m \times n}$:

ETNA
Kent State University and
Johann Radon Institute (RICAM)

(a) **Time-dependent**: Here $\mathbf{F}$ is a block-diagonal matrix of the form

$$\mathbf{F} = \begin{bmatrix} \mathbf{A}^{(1)} & & \\ & \ddots & \\ & & \mathbf{A}^{(n_t)} \end{bmatrix}, \tag{1.2}$$

where the blocks $\mathbf{A}^{(t)}$ may change in time $t = 1, \ldots, n_t$.

(b) **Time-independent**: Here $\mathbf{A}^{(t)} = \mathbf{A}$, for $t = 1, \ldots, n_t$, (that is, the blocks $\mathbf{A}^{(t)}$ are the same in time) so that $\mathbf{F}$ simplifies to $\mathbf{F} = \mathbf{I}_{n_t} \otimes \mathbf{A}$, with $\otimes$ being the Kronecker product.

The vector $\mathbf{d} = \mathrm{vec}([\mathbf{d}^{(1)}, \ldots, \mathbf{d}^{(n_t)}]) \in \mathbb{R}^m$ represents measured data that are contaminated by an unknown error (or noise) $\mathbf{e} \in \mathbb{R}^m$ that may stem from measurement errors. We assume that the noise vector follows a multivariate normal (or Gaussian) distribution with mean zero and covariance $\mathbf{\Gamma}$, i.e., $\mathbf{e} \sim \mathcal{N}(\mathbf{0}, \mathbf{\Gamma})$. The inverse problem involves recovering the pixels $\mathbf{u}$ from the data $\mathbf{d}$. That is, we seek to solve the general regularized problem

$$\mathbf{u}_{\text{dynamic}} = \arg\min_{\mathbf{u} \in \mathbb{R}^n} \mathcal{J}(\mathbf{u}) := \mathcal{F}(\mathbf{u}) + \lambda \mathcal{R}(\mathbf{u}), \tag{1.3}$$

where the functional $\mathcal{F}(\mathbf{u})$ is a data-misfit term that takes the form $\frac{1}{2}\|\mathbf{F}\mathbf{u} - \mathbf{d}\|^2_{\mathbf{\Gamma}^{-1}}$ and $\mathcal{R}(\mathbf{u})$ is a regularization term that can take different forms; several instances of $\mathcal{R}(\mathbf{u})$ will be discussed in Section 3. Throughout this paper, $\lambda > 0$ is an appropriate regularization parameter that determines a balance between the data-misfit and the regularization term $\mathcal{R}(\mathbf{u})$.

Static inverse problems. By contrast, in a static inverse problem, the information from each time step t is used to reconstruct the unknown images $\mathbf{u}^{(t)}$, $t = 1, 2, \ldots, n_t$. We assume that the measurement noise at each time step is independent of other time steps so that the overall noise covariance matrix $\mathbf{\Gamma} = \mathrm{BlockDiag}(\mathbf{\Gamma}_1, \ldots, \mathbf{\Gamma}_{n_t})$ is a block-diagonal matrix, where $\mathbf{\Gamma}_t$ is the noise covariance matrix at step t. We then solve the sequence of optimization problems

$$\mathbf{u}^{(t)}_{\text{static}} = \arg\min_{\mathbf{u} \in \mathbb{R}^{n_s}} \frac{1}{2}\|\mathbf{A}^{(t)}\mathbf{u} - \mathbf{d}^{(t)}\|^2_{\mathbf{\Gamma}_t^{-1}} + \lambda \mathcal{R}(\mathbf{u}), \qquad t = 1, 2, \ldots, n_t, \tag{1.4}$$

independently to obtain the solution to the static inverse problem.

Challenges. The considered inverse problems are typically ill-conditioned. Moreover, when solving dynamic inverse problems, the unknown has $n = n_s n_t$ pixels, which can be orders of magnitude higher than those for large-scale static inverse problems. Therefore, a clear challenge is the large scale of the considered problems. Furthermore, another challenge in dynamic inverse problems may stem from the limited information available per time instance during the measurement process.

This paper focuses on developing efficient regularization approaches for dynamic inverse problems that promote edge-preservation in the reconstructed images by incorporating specific representations of the prior information. Namely, we propose a combination of spatial and temporal prior information representations that allow for recovering piecewise constant solutions. We adopt efficient numerical methods that can enforce these representations.

1.2. Overview of the main contributions. This paper presents a unified computational framework for edge-preserving regularization in dynamic inverse problems. For each regularization term, we write down the corresponding optimization problem for reconstructing the desired solution, whose objective functions are convex but non-differentiable. To remedy the non-differentiability, we consider a smoothed functional instead, and we derive an iterative reweighted least-squares (IRLS) approach [7] for each optimization problem using

ETNA
Kent State University and
Johann Radon Institute (RICAM)

the majorization-minimization (MM) technique [41]. To efficiently solve the sequence of least-squares problems and define the regularization parameter, we use a generalized Krylov subspace (GKS) method [48], resulting in a so-called MM-GKS method. This unified approach has the following noteworthy features:

1. **flexibility**: the ability to choose between many different edge-preserving regularization techniques, each with its different strengths and weaknesses, but using the same MM-GKS solver;
2. **efficiency**: in contrast to inner-outer iteration schemes typical of IRLS methods applied to large-scale problems, the approach in this paper solves the optimization problem using a single generalized Krylov subspace, thus making judicious use of the forward/adjoint operator which can be expensive in many applications;
3. **automated**: the approach uses heuristics to automatically select regularization parameters in the projected space associated with the generalized Krylov subspace while solving the inverse problem;
4. **practicality**: our approach is capable of reconstructing over 1.9 million pixels in fewer than 100 MM-GKS iterations and is demonstrated to be effective on a variety of test problems with simulated and real data arising from space-time image deblurring, photoacoustic tomography (PAT), and limited angle computerized tomography (CT).

In this paper, we illustrate our framework with six different regularization terms, based on TV, for combining spatio-temporal information. For each regularization technique, we provide a motivation and an interpretation using tensor notation, which is useful for further generalization and extensions. Our framework is applicable beyond dynamic inverse problems and extends to other problem settings requiring solution techniques that combine limited information from different sources to improve the quality of the resulting reconstruction and recover dynamic information from different channels such as multichannel imaging [44] and electroencephalographic current density reconstruction [34].

Related work. A review of dynamic inverse problems with temporal information is given in [38]. We limit our discussion to a few related references. First, we discuss the use of TV regularization for solving dynamic inverse problems. An approach similar to our anisotropic space-time TV (Section 3.1) was discussed in [20] for image restoration. The reference [63], while it did not consider dynamic problems, used a TV technique similar to 3D joint anisotropic space-time TV (Section 3.3). An important point here is that, while in related works specific regularization methods are used for dynamic inverse problems, our approach treats these regularization techniques in a unified framework, using the same solver and the same technique to estimate the regularization parameter, which can be applied to ill-posed inverse problems in general. Beyond TV, some works consider edge-preserving reconstructions in dynamic inverse problems. The approach in [15, 52] is to use optical flow for jointly reconstructing the image and estimating object motion. In [10], a 3D shearlet-based approach is used for dynamic inverse problems in two spatial dimensions with time as the third dimension.

Overview of the paper. This paper is organized as follows. In Section 2, we present some background material, including additional notation, a survey of well-established regularization terms, and an iterative method used to solve the inverse problem with an MM strategy. In Section 3, we discuss six different methods for edge-preserving regularization in dynamic inverse problems, write a unifying framework, and derive, by using an MM approach, an IRLS method for solving the resulting optimization problem. Some alternative approaches and extensions that fit within our framework are presented in Section 4. In Section 5, we describe iterative methods based on generalized Krylov subspaces to efficiently solve the resulting optimization problem and define the regularization parameter at each iteration. In

ETNA
Kent State University and
Johann Radon Institute (RICAM)

Section 6, we present numerical examples that demonstrate the performance of the proposed regularization terms and the MM solvers. Finally, some conclusions, remarks, and future directions are presented in Section 7.

2. Background. In this section, we review known facts about tensors, regularization terms such as (discrete) isotropic and anisotropic TV, and the MM approach for solving optimization problems.

2.1. Tensor notation. The use of tensor notation is very convenient for describing dynamic images. A *tensor* $\boldsymbol{\mathcal{X}}$ is a multi-dimensional array (also called *n-way* or *n-mode* array) whose entries are scalars. A tensor's *order* refers to the number of ways or modes. For instance, vectors are tensors of order one, and matrices are tensors of order two. More details on tensors can be found in [47].

In this work, we primarily focus on 3rd-order tensors $\boldsymbol{\mathcal{X}} \in \mathbb{R}^{n_1 \times n_2 \times n_3}$ with entries $x_{i,j,k}$. Fibers are higher-order analogs of matrix rows and columns. A (tubal) *fiber* of a third-order tensor is a vector that is obtained by fixing two of the indices of the tensor $\boldsymbol{\mathcal{X}}$. We define $\boldsymbol{\mathcal{X}}_{:,j,k}$, $\boldsymbol{\mathcal{X}}_{i,:,k}$, and $\boldsymbol{\mathcal{X}}_{i,j,:}$ to be mode-1, mode-2, and mode-3 fibers, respectively. We implicitly assume that once a mode fiber has been extracted, it is reshaped as a column vector. *Slices* are two-dimensional sections of a tensor that are obtained by fixing one of the indices. We define $\boldsymbol{\mathcal{X}}_{i,:,:}$, $\boldsymbol{\mathcal{X}}_{:,j,:}$, and $\boldsymbol{\mathcal{X}}_{:,:,k}$ to be horizontal, lateral, and frontal slices, respectively. As before, when a slice is extracted, we implicitly assume it is a matrix. The *mode-j unfolding* or *matricization* of a tensor $\boldsymbol{\mathcal{X}}$ is obtained by arranging the mode-j fibers to be the columns of a resulting matrix. We denote these by $\mathbf{X}_{(1)} \in \mathbb{R}^{n_1 \times (n_2 n_3)}$, $\mathbf{X}_{(2)} \in \mathbb{R}^{n_2 \times (n_1 n_3)}$, and $\mathbf{X}_{(3)} \in \mathbb{R}^{n_3 \times (n_1 n_2)}$.

Another important concept here is the *mode-j product* that defines the operation of multiplying a tensor $\boldsymbol{\mathcal{X}} \in \mathbb{R}^{n_1 \times n_2 \times n_3}$ by a matrix $\mathbf{L}_j \in \mathbb{R}^{r \times n_j}$, for $j = 1, 2, 3$, given in the following definition. We write $\boldsymbol{\mathcal{Y}} = \boldsymbol{\mathcal{X}} \times_j \mathbf{L}_j$ in terms of the mode unfoldings as $\mathbf{Y}_{(j)} = \mathbf{L}_j \mathbf{X}_{(j)}$. For distinct modes in a series of multiplications, the order of the multiplication is irrelevant.

We will also need to use *norms* for tensors, which we define entrywise. That is, for $q \in [1, \infty)$, we define

$$\|\boldsymbol{\mathcal{X}}\|_q = \left(\sum_{i=1}^{n_1} \sum_{j=1}^{n_2} \sum_{k=1}^{n_3} |x_{i,j,k}|^q \right)^{1/q}.$$

A tensor representation of the dynamic inverse problem solution described in Section 1.1 is obtained by defining the multi-dimensional array $\boldsymbol{\mathcal{U}} \in \mathbb{R}^{n_v \times n_h \times n_t}$, with its frontal slices taken to be 2D representations of the image $\mathbf{u}^{(t)}$. That is, we let

$$\boldsymbol{\mathcal{U}}_{:,:,t} = \text{mat}(\mathbf{u}^{(t)}) \in \mathbb{R}^{n_v \times n_h}, \qquad t = 1, \ldots, n_t\,. \tag{2.1}$$

Furthermore, $\mathbf{u}^{(t)}$ are the mode-3 fibers and $\mathbf{U} = \mathbf{U}_{(3)}^T$ is the transposed mode-3 unfolding.

2.2. Regularization terms based on the first derivative operator. When the desired solution is known to be piecewise constant, TV regularization is a popular choice. It allows the solution to have discontinuities by preserving edges and discouraging oscillations [18, 19, 35, 51, 59]. TV regularization enforces sparse gradient representations for the solution.

Let

$$\mathbf{L}_d = \alpha_d \begin{bmatrix} 1 & -1 & & & \\ & 1 & -1 & & \\ & & \ddots & \ddots & \\ & & & 1 & -1 \end{bmatrix} \in \mathbb{R}^{(n_d-1) \times n_d} \qquad \text{and} \qquad \mathbf{I}_{n_d} \in \mathbb{R}^{n_d \times n_d} \tag{2.2}$$

ETNA
Kent State University and
Johann Radon Institute (RICAM)

be a rescaled finite difference discretization of the first-derivative operator with $\alpha_d > 0$ and the identity matrix of order n_d, respectively. Operators of this kind are known to damp fast oscillatory components of a vector $\mathbf{u}^{(t)}$; see, for instance, a discussion in [26]. In defining some of the operators below, we will augment the matrix $\mathbf{L}_d$ with one zero row (at the bottom) and denote it by $\bar{\mathbf{L}}_d$. The matrices $\mathbf{L}_d$ and $\bar{\mathbf{L}}_d$ are used to obtain discretizations of the first derivatives in the d-direction, with $d = v$ (vertical direction), $d = h$ (horizontal direction), and $d = t$ (time direction). For simplicity, in the following, we let $\alpha_d = 1$, but different values can be used in practice: a value $\alpha_d \neq 1$ can be treated as a regularization parameter that must be estimated as part of the inversion process.

Considering only the spatial derivatives for now, these have the form

$$\begin{aligned} \mathrm{vec}(\mathbf{L}_v \mathbf{U}^{(t)}) &= (\mathbf{I}_{n_h} \otimes \mathbf{L}_v)\mathbf{u}^{(t)} \ \in \mathbb{R}^{(n_v-1)n_h}, \\ \mathrm{vec}(\mathbf{U}^{(t)} \mathbf{L}_h^T) &= (\mathbf{L}_h \otimes \mathbf{I}_{n_v})\mathbf{u}^{(t)} \ \in \mathbb{R}^{(n_h-1)n_v}, \end{aligned} \qquad t = 1, \dots, n_t\,.$$

When time is considered, we have

$$\mathrm{vec}(\mathbf{U}\mathbf{L}_t^T) = (\mathbf{L}_t \otimes \mathbf{I}_{n_s})\mathbf{u} \ \in \mathbb{R}^{(n_t-1)n_s}.$$

By letting $n_t = 1$ (i.e., $n = n_s$) for now, so that $\mathbf{u} = \mathbf{u}^{(1)} = \mathrm{vec}(\mathbf{U}^{(1)})$, we define the anisotropic TV ($\mathrm{TV}_{\mathrm{aniso}}$) as

$$\begin{aligned} \mathrm{TV}_{\mathrm{aniso}}(\mathbf{u}) &= \sum_{k=1}^{(n_v-1)} \sum_{\ell=1}^{n_h} \left| \left(\mathbf{L}_v \mathbf{U}^{(1)} \right)_{k,\ell} \right| + \sum_{k=1}^{(n_h-1)} \sum_{\ell=1}^{n_v} \left| \left(\mathbf{U}^{(1)} \mathbf{L}_h^T \right)_{k,\ell} \right| \\ &= \|(\mathbf{I}_{n_h} \otimes \mathbf{L}_v)\mathbf{u}\|_1 + \|(\mathbf{L}_h \otimes \mathbf{I}_{n_v})\mathbf{u}\|_1 = \|\mathbf{L}_s \mathbf{u}\|_1\,, \end{aligned} \tag{2.3}$$

where

$$\mathbf{L}_s = \begin{bmatrix} \mathbf{I}_{n_h} \otimes \mathbf{L}_v \\ \mathbf{L}_h \otimes \mathbf{I}_{n_v} \end{bmatrix}.$$

Assuming for simplicity that $n_h = n_v$, we define the isotropic TV ($\mathrm{TV}_{\mathrm{iso}}$) as

$$\begin{aligned} \mathrm{TV}_{\mathrm{iso}}(\mathbf{u}) &= \sum_{k=1}^{n_v} \sum_{\ell=1}^{n_h} \sqrt{(\bar{\mathbf{L}}_v \mathbf{U}^{(1)})_{k,\ell}^2 + (\mathbf{U}^{(1)} (\bar{\mathbf{L}}_h)^T)_{k,\ell}^2} \\ &= \sum_{\ell=1}^{n_v n_h} \sqrt{((\mathbf{I}_{n_h} \otimes \bar{\mathbf{L}}_v)\mathbf{u})_\ell^2 + ((\bar{\mathbf{L}}_h \otimes \mathbf{I}_{n_v})\mathbf{u})_\ell^2} \\ &= \left\| \left[(\mathbf{I}_{n_h} \otimes \bar{\mathbf{L}}_v)\mathbf{u},\, (\bar{\mathbf{L}}_h \otimes \mathbf{I}_{n_v})\mathbf{u} \right] \right\|_{2,1}\,, \end{aligned}$$

where $\|\cdot\|_{2,1}$ denotes the norm defined as $\|\mathbf{Y}\|_{2,1} = \sum_{i=1}^{m_y} \|\mathbf{Y}_{i,:}\|_2$ for a matrix $\mathbf{Y} \in \mathbb{R}^{m_y \times n_y}$.

2.3. A majorization-minimization method. In this section, we provide an overview of the majorization-minimization technique for approximating the solution of (1.3) by solving a sequence of optimization problems; see [42, 49] for more details on the MM methods used. Suppose we want to minimize an objective function $\mathcal{J}(\mathbf{u})$. We shall need the following definition of a quadratic tangent majorant.

DEFINITION 2.1 ([41]). *Let $\mathbf{y} \in \mathbb{R}^n$ be fixed. The functional $\mathcal{Q}(\cdot;\mathbf{y})\colon \mathbb{R}^n \to \mathbb{R}$ is said to be a quadratic tangent majorant for $\mathcal{J}(\mathbf{x})$ at $\mathbf{x} = \mathbf{y} \in \mathbb{R}^n$ if it satisfies the following conditions:*

1. *$\mathcal{Q}(\mathbf{x};\mathbf{y})$ is quadratic in $\mathbf{x}$,*

ETNA
Kent State University and
Johann Radon Institute (RICAM)

2. $\mathcal{Q}(\mathbf{y};\mathbf{y}) = \mathcal{J}(\mathbf{y})$,
3. $\nabla_{\mathbf{x}}\mathcal{Q}(\mathbf{y};\mathbf{y}) = \nabla_{\mathbf{x}}\mathcal{J}(\mathbf{y})$,
4. $\mathcal{Q}(\mathbf{x};\mathbf{y}) \geq \mathcal{J}(\mathbf{x}) \quad \forall \mathbf{x} \in \mathbb{R}^n$.

The MM methods considered in this paper establish an iterative scheme whereby, starting from a given approximation of $\mathbf{u}_{\mathrm{true}}$, a quadratic tangent majorant functional for $\mathcal{J}(\mathbf{u})$ at the approximation of $\mathbf{u}_{\mathrm{true}}$ computed at the previous iteration is defined and approximately minimized to get the next approximation of $\mathbf{u}_{\mathrm{true}}$. In other words, after the approximation $\mathbf{u}_{(k)}$ has been computed at the kth iteration of the MM scheme, the $(k+1)$st approximate solution is computed as

$$\mathbf{u}_{(k+1)} = \arg\min_{\mathbf{u}\in\mathbb{R}^n} \mathcal{Q}(\mathbf{u};\mathbf{u}_{(k)}), \qquad k = 0, 1, \ldots \tag{2.4}$$

At the first iteration, one may take $\mathbf{u}_{(0)} = \mathbf{0}$. The convergence of the MM approach with quadratic tangent majorants was established in [41], which we also use in this paper.

3. Dynamic edge-preserving regularization.

We propose a unified framework with six main methods for edge-preserving reconstruction applied to dynamic inverse problems with a spatial and time component. For each technique, we motivate the kind of regularization, and using an MM approach, we derive an IRLS method for solving the resulting optimization problem. To save on space, we provide a detailed derivation for one of the terms (AnisoTV) and leave the other derivations in Appendix A. We also provide an interpretation for the regularization term using tensor notation.

3.1. Anisotropic space-time total variation (AnisoTV).

In this first technique, we use the summation of the anisotropic TV of the images at each time step as a regularizer as well as regularization for temporal information. Let $\mathbf{L}_s$ be as in (2.3). The anisotropic TV terms $\|\mathbf{L}_s\mathbf{u}^{(t)}\|_1$, $t = 1, \ldots, n_t$, ensure that the discrete spatial gradients of the images are sparse at each time instant. In addition, to incorporate temporal information, assuming that the images do not change considerably from one time instant to the next, we also want to penalize the difference between any two consecutive images; we do so by considering the 1-norm differences $\|\mathbf{u}^{(t+1)} - \mathbf{u}^{(t)}\|_1$, for $t = 1, \ldots, n_t - 1$. These two requirements can be imposed using the following regularization term

$$\begin{aligned}
\mathcal{R}_1(\mathbf{u}) &= \sum_{t=1}^{n_t} \|\mathbf{L}_s\mathbf{u}^{(t)}\|_1 + \sum_{t=1}^{n_t-1} \|\mathbf{u}^{(t+1)} - \mathbf{u}^{(t)}\|_1 \\
&= \|(\mathbf{I}_{n_t} \otimes \mathbf{L}_s)\mathbf{u}\|_1 + \|(\mathbf{L}_t \otimes \mathbf{I}_{n_s})\mathbf{u}\|_1 \\
&= \|\mathbf{D}_1\mathbf{u}\|_1, \qquad \text{where} \qquad \mathbf{D}_1 = \begin{bmatrix} \mathbf{I}_{n_t} \otimes \mathbf{L}_s \\ \mathbf{L}_t \otimes \mathbf{I}_{n_s} \end{bmatrix} = \begin{bmatrix} \mathbf{I}_{n_t} \otimes \mathbf{I}_{n_h} \otimes \mathbf{L}_v \\ \mathbf{I}_{n_t} \otimes \mathbf{L}_h \otimes \mathbf{I}_{n_v} \\ \mathbf{L}_t \otimes \mathbf{I}_{n_h} \otimes \mathbf{I}_{n_v} \end{bmatrix}.
\end{aligned} \tag{3.1}$$

Alternatively, recalling the tensor representation $\boldsymbol{\mathcal{U}}$ of $\mathbf{u}$ in (2.1), we can write

$$\mathcal{R}_1(\mathbf{u}) = \|\boldsymbol{\mathcal{U}} \times_1 \mathbf{L}_v\|_1 + \|\boldsymbol{\mathcal{U}} \times_2 \mathbf{L}_h\|_1 + \|\boldsymbol{\mathcal{U}} \times_3 \mathbf{L}_t\|_1.$$

The optimization problem and the MM approach. With the regularization term defined as in (3.1), the optimization problem that we seek to solve takes the form

$$\min_{\mathbf{u}\in\mathbb{R}^n} \mathcal{J}_1(\mathbf{u}) := \mathcal{F}(\mathbf{u}) + \lambda\mathcal{R}_1(\mathbf{u}), \qquad \text{where } \lambda > 0\,. \tag{3.2}$$

We now derive an MM approach for solving this optimization problem by solving a sequence of simpler optimization problems whose closed-form solutions exist. We do this in detail here since the other regularization terms we propose have similar derivations. At the kth iteration of the MM method, let $\mathbf{u}_{(k)}$ be the current iterate. Since the regularization term is nondifferentiable, we first majorize it as

$$\mathcal{R}_1(\mathbf{u}) \leq \sum_{\ell} \sqrt{(\mathbf{D}_1\mathbf{u})_\ell^2 + \epsilon^2} =: \mathcal{R}_{1\epsilon}(\mathbf{u}), \tag{3.3}$$

where $\mathcal{R}_{1\epsilon}$ is the smoothed regularization term. Similarly, we define the smoothed objective function $\mathcal{J}_{1\epsilon}$, by replacing $\mathcal{R}_1(\mathbf{u})$ with $\mathcal{R}_{1\epsilon}(\mathbf{u})$ in (3.2).

To obtain a quadratic tangent majorant, we use the elementary inequality [49, Equation (1.5)]

$$\sqrt{u} \leq \sqrt{v} + \frac{1}{2\sqrt{v}}(u - v), \tag{3.4}$$

for $u, v > 0$, which is an equality if $u = v$. By applying (3.4) to each term in the sum (3.3), with $u = (\mathbf{D}_1\mathbf{u})_\ell^2 + \epsilon^2$ and $v = (\mathbf{D}_1\mathbf{u}_{(k)})_\ell^2 + \epsilon^2$, we obtain that

$$\mathcal{R}_1(\mathbf{u}) \leq \sum_{\ell} \frac{1}{2\sqrt{(\mathbf{D}_1\mathbf{u}_{(k)})_\ell^2 + \epsilon^2}}(\mathbf{D}_1\mathbf{u})_\ell^2 + \tilde{c}_1 = \frac{1}{2}\|\mathbf{M}_1^{(k)}\mathbf{u}\|_2^2 + \tilde{c}_1,$$

where $\tilde{c}_1$ is a constant independent of $\mathbf{u}$ (but dependent on $\mathbf{u}_{(k)}, \mathbf{D}_1$, and ϵ) and $\mathbf{M}_1^{(k)}$ is the weighting matrix

$$\mathbf{M}_1^{(k)} := \mathbf{W}_1^{(k)}\mathbf{D}_1, \quad \text{with} \quad \mathbf{W}_1^{(k)} = \text{diag}((\mathbf{D}_1\mathbf{u}_{(k)})^2 + \epsilon^2)^{-1/4}). \tag{3.5}$$

Note that all operations in the expressions on the right-hand sides, including squaring, are performed entry-wise.

We can now define the quadratic tangent majorant $\mathcal{Q}_1(\mathbf{u}; \mathbf{u}_{(k)})$ for the objective function $\mathcal{J}_{1\epsilon}(\mathbf{u})$ as

$$\mathcal{Q}_1(\mathbf{u}; \mathbf{u}_{(k)}) := \mathcal{F}(\mathbf{u}) + \frac{\lambda}{2}\|\mathbf{M}_1^{(k)}\mathbf{u}\|_2^2 + c_1,$$

where $c_1 = \lambda\tilde{c}_1$.

Thus, as described in Section 2.3, we state the IRLS approach for solving the optimization problem (3.2): given an initial guess $\mathbf{u}_{(0)}$, we solve the sequence of optimization problems

$$\mathbf{u}_{(k+1)} = \arg\min_{\mathbf{u}\in\mathbb{R}^n} \mathcal{Q}_1(\mathbf{u}; \mathbf{u}_{(k)}), \qquad k = 0, 1, 2, \ldots, \tag{3.6}$$

to obtain the next iterate $\mathbf{u}_{(k+1)}$. Namely, this can be interpreted as an IRLS approach for the smooth approximation $\mathcal{J}_{1\epsilon}$ since, at each iteration, it replaces the regularization term $\mathcal{R}_{1\epsilon}(\mathbf{u})$ by an iteratively reweighted ℓ_2-regularization term.

3.2. Total variation in space and Tikhonov in time (TVplusTikhonov). In this technique, we consider anisotropic TV in space and assume that the target of interest has small changes in time. Then, we define a new regularization term as

$$\begin{aligned} \mathcal{R}_2(\mathbf{u}) &:= \sum_{t=1}^{n_t} \|\mathbf{L}_s\mathbf{u}^{(t)}\|_1 + \sum_{t=1}^{n_t-1} \|\mathbf{u}^{(t+1)} - \mathbf{u}^{(t)}\|_2^2 \\ &= \|(\mathbf{I}_{n_t} \otimes \mathbf{L}_s)\mathbf{u}\|_1 + \|(\mathbf{L}_t \otimes \mathbf{I}_{n_s})\mathbf{u}\|_2^2. \end{aligned} \tag{3.7}$$

In tensor notation, similar to $\mathcal{R}_1(\mathbf{u})$, we can succinctly write

$$\mathcal{R}_2(\mathbf{u}) = \|\boldsymbol{\mathcal{U}} \times_1 \mathbf{L}_v\|_1 + \|\boldsymbol{\mathcal{U}} \times_2 \mathbf{L}_h\|_1 + \|\boldsymbol{\mathcal{U}} \times_3 \mathbf{L}_t\|_2^2.$$

Note that, when compared with $\mathcal{R}_1(\mathbf{u})$, $\mathcal{R}_2(\mathbf{u})$ requires the difference between the images at consecutive time steps to be small. In contrast, $\mathcal{R}_1(\mathbf{u})$ additionally promotes the sparsity of the difference. Details about how to apply the MM method to minimize the functional $\mathcal{J}_2(\mathbf{u}) := \mathcal{F}(\mathbf{u}) + \lambda\mathcal{R}_2(\mathbf{u})$ are provided in Appendix A.1.

3.3. 3D joint anisotropic space-time total variation (Aniso3DTV). To explain this approach, it is easier to consider the tensor notation directly. We define the tensor $\boldsymbol{\mathcal{Y}}$ in which the finite difference tensor is applied simultaneously across all three modes

$$\boldsymbol{\mathcal{Y}} = \boldsymbol{\mathcal{U}} \times_1 \mathbf{L}_v \times_2 \mathbf{L}_h \times_3 \mathbf{L}_t.$$

We can write the 3D anisotropic TV norm as the vectorized 1-norm of this tensor. That is

$$\mathcal{R}_3(\mathbf{u}) = \|\boldsymbol{\mathcal{Y}}\|_1 = \sum_{v=1}^{n_v}\sum_{h=1}^{n_h}\sum_{t=1}^{n_t} |y_{v,h,t}|.$$

This is in contrast to $\mathcal{R}_1(\mathbf{u})$ in (3.1), which computes the sum of the tensor 1-norms in which only one derivative is applied per summand.

To derive an equivalent representation using matrix notation, consider the mode-1 unfolding of the tensor $\boldsymbol{\mathcal{Y}}$, $\mathbf{Y}_{(1)} = \mathbf{L}_v\mathbf{U}_{(1)}(\mathbf{L}_t^T \otimes \mathbf{L}_h^T)$. Let $\mathbf{y} = \mathrm{vec}(\mathbf{Y}_{(1)})$ and $\mathbf{u} = \mathrm{vec}(\mathbf{U}_{(1)})$ denote the vectorizations of the mode-1 unfoldings of $\boldsymbol{\mathcal{Y}}$ and $\boldsymbol{\mathcal{U}}$, respectively, which are related through the formula

$$\mathbf{y} = \mathbf{D}_3\mathbf{u}, \quad \text{with} \quad \mathbf{D}_3 = (\mathbf{L}_t \otimes \mathbf{L}_h \otimes \mathbf{L}_v), \quad \text{so that} \quad \mathcal{R}_3(\mathbf{u}) := \|\mathbf{D}_3\mathbf{u}\|_1.$$

Details about how to apply the MM method to minimize the functional $\mathcal{J}_3(\mathbf{u}) := \mathcal{F}(\mathbf{u}) + \lambda\mathcal{R}_3(\mathbf{u})$ are provided in Appendix A.2.

3.4. 3D joint isotropic space-time total variation (Iso3DTV). In this next approach, we apply isotropic TV in all three directions, i.e., two spatial and one temporal direction. We first introduce the variables

$$\begin{aligned}
\bar{\mathbf{z}}_v(\mathbf{u}) &:= (\mathbf{I}_{n_t} \otimes \mathbf{I}_{n_h} \otimes \bar{\mathbf{L}}_v)\mathbf{u}\,,\\
\bar{\mathbf{z}}_h(\mathbf{u}) &:= (\mathbf{I}_{n_t} \otimes \bar{\mathbf{L}}_h \otimes \mathbf{I}_{n_v})\mathbf{u}\,,\\
\bar{\mathbf{z}}_t(\mathbf{u}) &:= (\bar{\mathbf{L}}_t \otimes \mathbf{I}_{n_h} \otimes \mathbf{I}_{n_v})\mathbf{u}\,.
\end{aligned} \tag{3.8}$$

Recall that $\bar{\mathbf{L}}_d$, $d = v, h, t$ is obtained by augmenting $\mathbf{L}_d$ with a row of zeros. Then, we can compactly write the following regularization term

$$\begin{aligned}
\mathcal{R}_4(\mathbf{u}) :=& \sum_{\ell=1}^{n_v n_h n_t} \sqrt{(\bar{\mathbf{z}}_v(\mathbf{u}))_\ell^2 + (\bar{\mathbf{z}}_h(\mathbf{u}))_\ell^2 + (\bar{\mathbf{z}}_t(\mathbf{u}))_\ell^2}\\
=& \;\|\,[\bar{\mathbf{z}}_v(\mathbf{u}), \bar{\mathbf{z}}_h(\mathbf{u}), \bar{\mathbf{z}}_t(\mathbf{u})]\,\|_{2,1}.
\end{aligned}$$

To devise a tensor formulation for $\mathcal{R}_4(\mathbf{u})$, first consider the following tensors

$$\boldsymbol{\mathcal{Z}}_v = \boldsymbol{\mathcal{U}} \times_1 \bar{\mathbf{L}}_v, \qquad \boldsymbol{\mathcal{Z}}_h = \boldsymbol{\mathcal{U}} \times_2 \bar{\mathbf{L}}_h, \qquad \boldsymbol{\mathcal{Z}}_t = \boldsymbol{\mathcal{U}} \times_3 \bar{\mathbf{L}}_t,$$

and their mode-3 unfoldings $(\mathbf{Z}_v)_{(3)}$, $(\mathbf{Z}_h)_{(3)}$, $(\mathbf{Z}_t)_{(3)}$, respectively. Define a new tensor $\boldsymbol{\mathcal{Y}} \in \mathbb{R}^{n_s \times n_t \times 3}$ such that

$$\boldsymbol{\mathcal{Y}}_{:,:,1} = (\mathbf{Z}_v)_{(3)}^T, \qquad \boldsymbol{\mathcal{Y}}_{:,:,2} = (\mathbf{Z}_h)_{(3)}^T, \qquad \boldsymbol{\mathcal{Y}}_{:,:,3} = (\mathbf{Z}_t)_{(3)}^T.$$

Then, $\mathcal{R}_4(\mathbf{u})$ is the sum of the 2-norms of the mode-3 fibers of $\boldsymbol{\mathcal{Y}}$, that is,

$$\mathcal{R}_4(\mathbf{u}) = \sum_{i=1}^{n_s}\sum_{j=1}^{n_t} \|\boldsymbol{\mathcal{Y}}_{i,j,:}\|_2 \,.$$

To interpret this representation, the frontal slices of the tensor $\boldsymbol{\mathcal{Y}}$ are the collection of gradient images at all time instances, and the derivatives are taken one direction at a time. The regularization operator $\mathcal{R}_4(\mathbf{u})$ is the sum of two norms of its tubal fibers. Details about how to apply the MM method to minimize the functional $\mathcal{J}_4(\mathbf{u}) := \mathcal{F}(\mathbf{u}) + \lambda\mathcal{R}_4(\mathbf{u})$ are provided in Appendix A.3.

3.5. Isotropic in space, anisotropic in time total variation (IsoTV). This method can be considered a variation of the AnisoTV method presented in Section 3.1, where only the spatial anisotropic TV is replaced by the spatial isotropic TV. Namely, using the notation in (3.8), we consider the regularization term

$$\begin{aligned}\mathcal{R}_5(\mathbf{u}) &= \sum_{\ell=1}^{n_v n_h n_t} \sqrt{(\bar{\mathbf{z}}_v(\mathbf{u}))_\ell^2 + (\bar{\mathbf{z}}_h(\mathbf{u}))_\ell^2} + \sum_{t=1}^{n_t-1} \|\mathbf{u}^{(t+1)} - \mathbf{u}^{(t)}\|_1 \\ &= \|\,[\bar{\mathbf{z}}_v(\mathbf{u}), \bar{\mathbf{z}}_h(\mathbf{u})]\,\|_{2,1} + \|(\mathbf{L}_t \otimes \mathbf{I}_{n_s})\mathbf{u}\|_1.\end{aligned}$$

The associated tensor formulation reads similar to the ones presented in Sections 3.1 and 3.4, namely,

$$\mathcal{R}_5(\mathbf{u}) = \sum_{i=1}^{n_s}\sum_{j=1}^{n_t} \|\boldsymbol{\mathcal{Y}}_{i,j,:}\|_2 + \|\boldsymbol{\mathcal{U}} \times_3 \mathbf{L}_t\|_1,$$

where $\boldsymbol{\mathcal{Y}} \in \mathbb{R}^{n_s \times n_t \times 2}$ is such that $\boldsymbol{\mathcal{Y}}_{:,:,1} = (\mathbf{Z}_v)_{(3)}^T$, and $\boldsymbol{\mathcal{Y}}_{:,:,2} = (\mathbf{Z}_h)_{(3)}^T$. Details about how to apply the MM method to minimize the functional $\mathcal{J}_5(\mathbf{u}) := \mathcal{F}(\mathbf{u}) + \lambda\mathcal{R}_5(\mathbf{u})$ are provided in Appendix A.4.

3.6. Group sparsity (GS). Group sparsity allows to promote sparsity when reconstructing a vector of unknown pixels that are naturally partitioned in subsets; see [4]. In our applications, there are several possible ways to define groups. For example, we can naturally group the variables corresponding to pixels at each time instant, i.e., $\{\mathbf{u}^{(t)}\}_{t=1}^{n_t}$. To enforce piecewise constant structures in space and time, we adopt the following approach. Let $n_s' = (n_v - 1)n_h + (n_h - 1)n_v$ be the total number of pixels in the gradient images. Consider the groups defined by the vectors

$$\mathbf{z}_\ell = \left[(\mathbf{L}_s\mathbf{u}^{(1)})_\ell, \ldots, (\mathbf{L}_s\mathbf{u}^{(n_t)})_\ell\right] = \left(\mathbf{I}_{n_t} \otimes \mathbf{e}_\ell^T\mathbf{L}_s\right)\mathbf{u} \in \mathbb{R}^{n_t}, \qquad \ell = 1, \ldots, n_s'.$$

Alternatively, define the matrix $\mathbf{Z}$ whose columns represent the vectorized gradient images at different time t as

$$\begin{aligned}\mathbf{Z} &= [\mathbf{L}_s\mathbf{u}^{(1)}, \ldots, \mathbf{L}_s\mathbf{u}^{(n_t)}] = \mathbf{L}_s\mathbf{U} \in \mathbb{R}^{n_s' \times v_t}, \\ \mathbf{z} &= \mathrm{vec}(\mathbf{Z}) = (\mathbf{I}_{n_t} \otimes \mathbf{L}_s)\mathbf{u}\,.\end{aligned} \tag{3.9}$$

Note that $\mathbf{z}_\ell$ are the rows of $\mathbf{Z}$. These are also illustrated in Figure 3.1. The regularization term corresponding to group sparsity can then be expressed as a mixture of norms

$$\mathcal{R}_6(\mathbf{u}) := \sum_{\ell=1}^{n_s'} \|\mathbf{z}_\ell\|_2 = \sum_{\ell=1}^{n_s'} \left(\sum_{t=1}^{n_t} (\mathbf{L}_s\mathbf{u}^{(t)})_\ell^2\right)^{1/2} = \|\mathbf{L}_s\mathbf{U}\|_{2,1}.$$

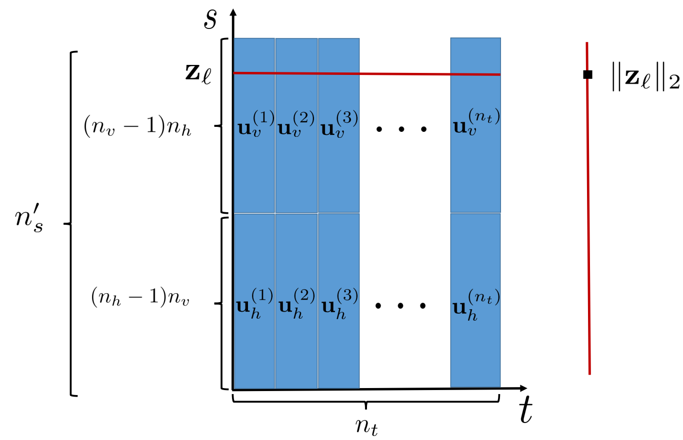


FIGURE 3.1. *The vector of spatial derivatives $\mathbf{L}_s\mathbf{u}^{(t)}$ contains the partial derivatives with respect to the vertical ($\mathbf{u}_v^{(t)}$) and horizontal ($\mathbf{u}_h^{(t)}$) directions for each image. These vectors are the columns of the matrix $\mathbf{Z}$ depicted here. We compute the 2-norm of each row $\mathbf{z}_\ell$ of $\mathbf{Z}$ and add them.*

In other words, the regularization term behaves like a 1-norm for the vector $[\|\mathbf{z}_1\|_2 \cdots \|\mathbf{z}_{n_s'}\|_2]$. This regularization term induces sparsity on the vector of 2-norms of $\mathbf{z}_\ell$, $\ell = 1,\dots,n_s'$, encouraging $\|\mathbf{z}_\ell\|_2$ (and, in turn, each vector $\mathbf{z}_\ell$) to be zero. On the one hand, by using this regularization, we are ensuring that the sparsity in the gradient images is being shared across time instances. On the other hand, this regularization formulation does not enforce sparsity across the groups, i.e., across the vectors $\mathbf{z}_\ell$.

To devise a tensor formulation, let $\boldsymbol{\mathcal{U}}$ be the tensor of images, and let $\boldsymbol{\mathcal{X}} = \boldsymbol{\mathcal{U}} \times_1 \mathbf{L}_v$ and $\boldsymbol{\mathcal{Y}} = \boldsymbol{\mathcal{U}} \times_2 \mathbf{L}_h$ be the tensors obtained by taking the gradient in the vertical and horizontal directions. Then, $\mathcal{R}_6(\mathbf{u})$ is the sum of 2-norms of the mode-3 fibers of $\boldsymbol{\mathcal{X}}$ and $\boldsymbol{\mathcal{Y}}$. That is,

$$\mathcal{R}_6(\mathbf{u}) = \sum_{i=1}^{(n_v-1)} \sum_{j=1}^{n_h} \|\boldsymbol{\mathcal{X}}_{i,j,:}\|_2 + \sum_{i=1}^{(n_h-1)} \sum_{j=1}^{n_v} \|\boldsymbol{\mathcal{Y}}_{i,j,:}\|_2.$$

Note also that, following (3.9), $\mathbf{Z} = [\mathbf{X}_{(3)}, \mathbf{Y}_{(3)}]^T$. Details about how to apply the MM method to minimize the functional $\mathcal{J}_6(\mathbf{u}) := \mathcal{F}(\mathbf{u}) + \lambda\mathcal{R}_6(\mathbf{u})$ are provided in Appendix A.5.

3.7. Summary of the proposed approaches. In this section, we have presented six different regularization terms for promoting edge-preserving reconstructions in dynamic inverse problems. Here we show that they can be treated in a unified fashion, providing a succinct summary of all the proposed methods. For each regularization term, we solve an optimization problem of the form

$$\min_{\mathbf{u}\in\mathbb{R}^n} \mathcal{J}_{j\epsilon}(\mathbf{u}) := \mathcal{F}(\mathbf{u}) + \lambda\mathcal{R}_{j\epsilon}(\mathbf{u}), \qquad \lambda > 0, \quad j = 1,\dots,6, \tag{3.10}$$

where $\mathcal{R}_{j\epsilon}(\mathbf{u})$ is a smoothed regularization term depending on the method used and $\mathcal{F}(\mathbf{u})$ is a term that measures the data-misfit. For each optimization problem, we have derived an MM approach that (partially) solves a sequence of IRLS problems. That is, given an initial guess $\mathbf{u}_{(0)}$, at step k we (partially) solve the optimization problem

$$\mathbf{u}_{(k+1)} = \arg\min_{\mathbf{u}\in\mathbb{R}^n} \frac{1}{2}\|\mathbf{F}\mathbf{u} - \mathbf{d}\|^2_{\mathbf{\Gamma}^{-1}} + \frac{\lambda}{2}\|\mathbf{M}_j^{(k)}\mathbf{u}\|_2^2, \qquad k = 0, 1\dots \tag{3.11}$$

The matrix $\mathbf{M}_j^{(k)}$ takes different forms depending on the regularization technique used.

Table 3.1 summarizes some details about the proposed regularization terms and points to the formulas defining the reweighting matrices appearing within $\mathbf{M}_j^{(k)}$ in the MM step. In Section 5, we discuss iterative methods to efficiently solve the sequence of least-squares problems (3.11) and select the regularization parameter λ.

TABLE 3.1
The six different methods introduced in Section 3, the associated regularization terms, and the weighting matrices for the MM step. The index j runs from 1 to 6. The vectors $\bar{\mathbf{z}}_d(\mathbf{u})$, $d = v, h, t$, are defined in (3.8).

Method	j	$\mathcal{R}_j(\mathbf{u})$	**MM weights**
AnisoTV	1	$\\|(\mathbf{I}_{n_t} \otimes \mathbf{L}_s)\mathbf{u}\\|_1 + \\|(\mathbf{L}_t \otimes \mathbf{I}_{n_s})\mathbf{u}\\|_1$	(3.5)
TVplusTikhonov	2	$\\|(\mathbf{I}_{n_t} \otimes \mathbf{L}_s)\mathbf{u}\\|_1 + \\|(\mathbf{L}_t \otimes \mathbf{I}_{n_s})\mathbf{u}\\|_2^2$	(A.3)
Aniso3DTV	3	$\\|(\mathbf{L}_t \otimes \mathbf{L}_h \otimes \mathbf{L}_v)\mathbf{u}\\|_1$	(A.4)
Iso3DTV	4	$\\| [\bar{\mathbf{z}}_v(\mathbf{u}), \bar{\mathbf{z}}_h(\mathbf{u}), \bar{\mathbf{z}}_t(\mathbf{u})] \\|_{2,1}$	(A.5)
IsoTV	5	$\\| [\bar{\mathbf{z}}_v(\mathbf{u}), \bar{\mathbf{z}}_h(\mathbf{u})] \\|_{2,1} + \\|(\mathbf{L}_t \otimes \mathbf{I}_{n_s})\mathbf{u}\\|_1$	(A.6)
GS	6	$\\|\mathbf{L}_s\mathbf{U}\\|_{2,1}$	(A.7)

4. Extensions and alternative approaches. In Section 3, we presented a variety of regularization methods that use different forms of TV and sparsity-enforcing regularization to obtain solutions methods that enhance edge representation. In this section, we summarize some alternative approaches that can be used, still within the MM framework, to enforce edge-preserving reconstructions.

Beyond the ℓ_1- and ℓ_2-norms. One way to interpret the anisotropic TV is that it enforces sparsity in the gradient images. A natural measure of the sparsity of a vector is the ℓ_0-"norm", which counts the number of nonzero entries. However, solving minimization problems that involve the ℓ_0-term is known to be NP-hard; hence to remedy this difficulty, one approximates the ℓ_0-"norm" by ℓ_1-convex relaxation. Several nonconvex penalties with $0 < q < 1$ have been used alternatively to ℓ_1; see [21, 70]. The methods we discuss in Section 3 can be generalized using ℓ_q-regularization. For example, the regularization term (3.1) in Section 3.1 can be generalized by choosing $\mathcal{R}_1^q(\mathbf{u}) = \frac{1}{q}\|\mathbf{D}_1\mathbf{u}\|_q^q$, for $0 < q \leq 2$. Similarly, the GS method (Section 3.6) can be expressed using general mixed ℓ_p-ℓ_q "norms" instead of ℓ_2-ℓ_1.

Beyond the gradient operator. One can build appropriate sparsity transforms using and combining operators other than the first-order finite difference operator $\mathbf{L}_d$ defined in (2.2), where $d = v, h, t$. A first simple extension replaces $\mathbf{L}_d$ by a discretization of the *second-order derivative operator*, which can still assist in preserving edges [2, 57]. Moreover, one can replace the operator $\mathbf{L}_d$ implicitly appearing in any of the regularizers defined in Section 3 by a *wavelet* transform; see [25, 54] and references therein for more details and properties of different classes of wavelets. Similar to wavelets, *framelet* representations of images are orthogonal basis transformations that form a dictionary of minimum size that initially decomposes the images into transformed coefficients; see [16]. Finally, several variations are also possible when specifically considering the GS regularizer proposed in Section 3.6. For instance, one can consider 'overlapping groups' and also replace $\mathbf{L}_s$ with other operators, such as the ones mentioned above. It is well-known that, beyond dynamic inverse problems, sparse representations can improve pattern recognition, feature extraction, compression, multi-task regression, and noise reduction; see, for example, [3, 46].

Beyond one single regularization parameter. Specifically for dynamic inverse problems, it may be meaningful to adapt the regularization parameters based on the dynamics. For instance, one can define dedicated regularization parameters for different domains (spatial or temporal). Within the framework presented in Section 3, this can be achieved by setting, in addition or as an alternative to λ, appropriate values for the parameters α_d in (2.2). For instance, [40] considers a scenario where the regularization parameters are different for the spatial and temporal domains. Although there is a rich literature on methods to estimate a single regularization parameter, finding multiple regularization parameters is challenging and an active area of research; see, e.g., [6, 30, 31].

5. Iterative methods for IRLS problems and parameter choice. In this section, we describe a numerical method to solve the optimization problems arising from the approaches described in Section 3.

Towards the end of this section, we describe how a suitable value for the regularization parameter $\lambda^{(k)}$ can be determined. To compute the iterate $\mathbf{u}_{(k+1)}$ as in (3.11), we set the gradient of $\mathcal{Q}_j(\mathbf{u};\mathbf{u}_{(k)})$ to zero, which leads to the regularized normal equations (or general Tikhonov problem)

$$\left(\mathbf{F}^T\mathbf{\Gamma}^{-1}\mathbf{F} + \lambda^{(k)}(\mathbf{M}_j^{(k)})^T\mathbf{M}_j^{(k)}\right)\mathbf{u}_{(k+1)} = \mathbf{F}^T\mathbf{\Gamma}^{-1}\mathbf{d}. \tag{5.1}$$

The system (5.1) has a unique solution if the null spaces of $\mathbf{F}^T\mathbf{\Gamma}^{-1}\mathbf{F}$ and $(\mathbf{M}_j^{(k)})^T\mathbf{M}_j^{(k)}$ or, equivalently, the null spaces of $\mathbf{F}^T\mathbf{F}$ and $\mathbf{D}_j^T\mathbf{D}_j$, only intersect at $\mathbf{0}$, $j = 1,\dots,6$ (for convenience, we have defined $\mathbf{D}_2 = \mathbf{D}_1$).

Therefore, for the methods 1–3 (AnisoTV, TVplusTikhonov, and Aniso3DTV), this matches the assumptions of [41, Theorem 5], and as a consequence, the sequence $\{\mathbf{u}_{(k)}\}$ converges to a global minimizer of $\mathcal{J}_{j\epsilon}(\mathbf{u})$ for each method (see [41, Corollary 6]). For the methods 4–6 (Iso3DTV, IsoTV, and GS), it may be possible to extend the analysis from that paper; however, we do not pursue it here.

Since solving (5.1) for large-scale matrices $\mathbf{F}$ and $\mathbf{M}^{(k)}$ may be computationally demanding or even prohibitive, we search for a solution to (5.1) in a low-dimensional subspace (namely, a generalized Krylov space) and solve a much smaller projected problem. If the approximate solution is unsatisfactory, then we extend the search space with the (normalized) residual and consider the next problem in the sequence (3.11) so that, for each k, only one projected problem is solved, as detailed below. This leads to the Generalized Krylov subspace (GKS) process [41, 48]. A summary of the resulting algorithm adapted to the problems described in this paper is sketched in Algorithm 1 below, together with a few explanations.

At the kth iteration of Algorithm 1, given a c-dimensional ($c \ll n$) search space $\mathcal{V}_c = \text{range}(\mathbf{V}_c)$, where $\mathbf{V}_c \in \mathbb{R}^{n\times c}$ has orthonormal columns, we compute an approximate solution to (5.1) by first computing the thin QR-decompositions $\mathbf{\Gamma}^{-1/2}\mathbf{F}\mathbf{V}_c = \mathbf{Q}_{\mathbf{F}}^{(k)}\mathbf{R}_{\mathbf{F}}^{(k)}$ and $\mathbf{M}_j^{(k)}\mathbf{V}_c = \mathbf{Q}_{\mathbf{M}}^{(k)}\mathbf{R}_{\mathbf{M}}^{(k)}$ (line 7) and by then substituting $\mathbf{u} = \mathbf{V}_c\mathbf{y}$ in (5.1), leading to the small minimization problem in line 9: its solution can be computed at a low cost, giving the approximate solution $\mathbf{u}_{(k+1)} = \mathbf{V}_c\mathbf{y}_{(k+1)}$ (line 10). The residual associated with (5.1) can be computed as

$$\mathbf{r}_{(k+1)} = \mathbf{F}^T\mathbf{\Gamma}^{-1}(\mathbf{F}\mathbf{V}_c\mathbf{y}_{(k+1)} - \mathbf{d}) + \lambda^{(k)}(\mathbf{M}_j^{(k)})^T\mathbf{M}_j^{(k)}\mathbf{V}_c\mathbf{y}_{(k+1)}. \tag{5.2}$$

If the stopping criteria (discussed in Section 6) are not satisfied, then we use the normalized residual to expand the search space, i.e., $\text{range}(\mathbf{V}_{c+1}) = \text{range}([\mathbf{V}_c, \quad \mathbf{r}_{(k+1)}/\|\mathbf{r}_{(k+1)}\|_2])$, as prescribed in lines 11–12 (note that while in exact arithmetic $\mathbf{r}_{(k+1)} \perp \mathbf{V}_c$, in practice, for numerical stability, we first explicitly orthogonalize the new residual against $\mathbf{V}_c$). We then compute $\mathbf{W}_j^{(k+1)}$ and $\mathbf{M}_j^{(k+1)}$ and continue the iterations, solving for $\mathbf{u}_{(k+2)}$. As $\mathbf{\Gamma}$ is fixed, the thin QR-decomposition $\mathbf{\Gamma}^{-1/2}\mathbf{F}\mathbf{V}_c = \mathbf{Q}_{\mathbf{F}}^{(k)}\mathbf{R}_{\mathbf{F}}^{(k)}$ can be updated efficiently when an additional column is appended to $\mathbf{V}_c$ (line 7). To compute a small initial search space

Algorithm 1 MM-GKS for dynamic inverse problems.

1: **Input:** Matrix $\mathbf{F} \in \mathbb{R}^{m \times n}$, noise-corrupted data $\mathbf{d} \in \mathbb{R}^m$, $\mathbf{\Gamma} \in \mathbb{R}^{m \times m}$, $\mathbf{D}_s \in \mathbb{R}^{r \times n}$ with $s = 1, 2, \ldots, 6$.
Dimension ℓ of the initial approximation subspace, parameters $\epsilon > 0$, stopping criterion tolerance.
2: Generate the initial subspace basis: $\mathbf{V}_\ell \in \mathbb{R}^{n \times \ell}$ such that $\mathbf{V}_\ell^T \mathbf{V}_\ell = \mathbf{I}$.
3: Compute $\mathbf{\Gamma}^{-1/2}\mathbf{F}\mathbf{V}_\ell$ and $\mathbf{M}_s\mathbf{V}_\ell$, and the QR factorization $\mathbf{\Gamma}^{-1/2}\mathbf{F}\mathbf{V}_\ell = \mathbf{Q}_\mathbf{F}\mathbf{R}_\mathbf{F}$;
Compute $\mathbf{u}_{(\mathbf{0})} = \mathbf{V}_\ell \arg\min_{\mathbf{y} \in \mathbb{R}^\ell} \left\| \mathbf{R}_\mathbf{F}\mathbf{y} - (\mathbf{Q}_\mathbf{F})^\mathbf{T}\mathbf{\Gamma}^{-\mathbf{1/2}}\mathbf{d} \right\|_\mathbf{2}^\mathbf{2}$
4: **for** $k = 0, 1, 2, \ldots$ until a stopping criterion is satisfied
5: Let $c = \ell + k$.
6: Compute $\mathbf{W}_s^{(k)}$ as in Table 3.1, using $\mathbf{u} = \mathbf{u}_{(k)}$; compute the corresponding $\mathbf{M}_s^{(k)}$.
7: Update $\mathbf{\Gamma}^{-1/2}\mathbf{F}\mathbf{V}_c = \mathbf{Q}_\mathbf{F}^{(k)}\mathbf{R}_\mathbf{F}^{(k)}$ and compute $\mathbf{M}_s^{(k)}\mathbf{V}_c = \mathbf{Q}_\mathbf{M}^{(k)}\mathbf{R}_\mathbf{M}^{(k)}$.
8: Determine the $\lambda^{(k)}$ by GCV; see [13, Section 3.2] for details.
9: Compute $\mathbf{y}_{(k+1)} = \arg\min_{\mathbf{y} \in \mathbb{R}^c} \left\| \begin{bmatrix} \mathbf{R}_\mathbf{F}^{(k)} \\ (\lambda^{(k)})^{1/2}\mathbf{R}_\mathbf{M}^{(k)} \end{bmatrix} \mathbf{y} - \begin{bmatrix} (\mathbf{Q}_\mathbf{F}^{(k)})^T\mathbf{\Gamma}^{-1/2}\mathbf{d} \\ \mathbf{0} \end{bmatrix} \right\|_2^2$.
10: Compute $\mathbf{u}_{(k+1)} = \mathbf{V}_c\mathbf{y}_{(k+1)}$.
11: Compute the residual

$$\mathbf{r}_{(k+1)} = \mathbf{F}^T\mathbf{\Gamma}^{-1}(\mathbf{F}\mathbf{V}_c\mathbf{y}_{(k+1)} - \mathbf{d}) + \lambda^{(k)}(\mathbf{M}_s^{(k)})^T\mathbf{M}_s^{(k)}\mathbf{V}_c\mathbf{y}_{(k+1)}.$$

12: Reorthogonalize: $\mathbf{r}_{(k+1)} = \mathbf{r}_{(k+1)} - \mathbf{V}_c\mathbf{V}_c^T\mathbf{r}_{(k+1)}$.
13: Enlarge the solution subspace with $\mathbf{v}_{\text{new}} = \frac{\mathbf{r}_{(k+1)}}{\|\mathbf{r}_{(k+1)}\|_2}$, $\mathbf{V}_{c+1} = [\mathbf{V}_c, \mathbf{v}_{\text{new}}]$.
14: **end for**

and the initial approximation $\mathbf{u}_{(0)}$, the GKS algorithm is generally started by running a few, say ℓ, steps of a Golub-Kahan bidiagonalization applied to $\mathbf{\Gamma}^{-1/2}\mathbf{F}$ and $\mathbf{\Gamma}^{-1/2}\mathbf{d}$ (line 2). We emphasize again that, at each iteration index k, an approximation of $\mathbf{u}_{(k+1)}$ in (3.10) is obtained by solving a single projected problem of dimension $k + \ell$.

To select the regularization parameter at the kth iteration (line 8), we work on the projected problem appearing in line 9 (solely involving small quantities). In particular, to efficiently apply generalized cross validation (GCV), we compute the generalized singular value decomposition of the $c \times c$ matrix pair $(\mathbf{R}_\mathbf{F}^{(k)}, \mathbf{R}_\mathbf{M}^{(k)})$.

Alternative well-established approaches based on the L-curve or the discrepancy principle (DP) [17] or the unbiased predictive risk estimator (UPRE) [58], can be applied.

Computational cost of Algorithm 1. Let $T_\mathbf{F}$ and $T_{\mathbf{F}^T}$ denote the cost of evaluating a matrix-vector product with $\mathbf{F}$ and its transpose $\mathbf{F}^T$, respectively; this cost depends on the forward operator used in the application. Similarly, let $T_\mathbf{M}$ and $T_{\mathbf{M}^T}$ denote the cost of computing matrix-vector products with $\mathbf{M}_s^{(k)}$ and its transpose $(\mathbf{M}_s^{(k)})^T$, respectively; these costs depends on the specific regularization approach that is used, but they are generally small compared to $T_\mathbf{F}$ and $T_{\mathbf{F}^T}$, since $\mathbf{M}_s^{(k)}$ is typically very sparse. At the kth iteration of Algorithm 1, two QR factorizations need to be computed: one for $\mathbf{\Gamma}^{-1/2}\mathbf{F}\mathbf{V}_c$ and one for $\mathbf{M}_s^{(k)}\mathbf{V}_c$. The cost of this is $\mathcal{O}((m+n)c^2)$; however, this can be mitigated for the term $\mathbf{\Gamma}^{-1/2}\mathbf{F}\mathbf{V}_c$ by updating the QR factorization rather than recomputing it from scratch. We cannot do that for the second term $\mathbf{M}_s^{(k)}\mathbf{V}_c$ since the entire matrix changes at each iteration. There is an additional cost of $\mathcal{O}(c^3)$ at each iteration to estimate the regularization parameter and a cost of $\mathcal{O}(nc)$ to reorthogonalize and produce an estimate of the solution. The total cost per iteration is therefore

$$\text{Cost} = T_{\mathbf{F}} + T_{\mathbf{F}^T} + T_{\mathbf{M}} + T_{\mathbf{M}^T} + \mathcal{O}((m+n)c^2 + c^3) \ \text{ flops.}$$

This analysis assumes that the initial basis $\mathbf{V}_\ell$ is available; when ℓ is small, as is the case in our experiments, this cost is negligible compared to the cost of the GKS approach. Algorithm 1 is computationally efficient when $T_{\mathbf{F}}$ and $T_{\mathbf{F}^T}$ are large compared to the cost of orthogonalization. The computational cost due to orthogonalization may be large when the number of iterations is high. Nevertheless, as we show in the numerical experiments (Table 6.3 for the dynamic PAT in Section 6), our solver is much faster than other approaches we consider. Alternatively, for large-scale problems with high memory requirements, a recently proposed restarted MMGKS can be used [14].

Developing even more efficient methods for large-scale dynamic inverse problems is an important topic for future study. Several possibilities can be explored, including using fixed quadratic majorant [41] and randomized sketching-based techniques [5].

6. Numerical experiments. In this section, we provide numerical examples from three different dynamic inverse problems. Our goal is two-fold: to show that using dynamic information can be advantageous in reconstructions and to compare the different spatio-temporal regularization methods proposed in this paper. In addition, we provide comparisons with several solvers such as ADMM and variations of MM, demonstrating the computational efficiency of our approach.

Discussion on the choice of numerical examples. Our first example considers a synthetic space-time image deblurring where images change in time, but the blurring operator is fixed for all the time instances. In this example, the true solution is available, which allows a comparison between the proposed methods.

The second example is a problem from dynamic photoacoustic tomography (PAT), in which there are few measurements per time step (since information is collected from limited angles) but many time steps yielding many measurements overall. This is the largest test problem we consider, with over 1.9 million unknowns, in which the forward operator $\mathbf{A}^{(t)}$ changes at each time step. In this example, we compare a few of the regularization methods for dynamic inverse problems against the static inverse problem. Furthermore, we also compare the solvers adopted in this paper with other MM solvers and a state-of-the-art method, i.e., ADMM. The final example concerns real data arising from limited angle CT where the target of interest is a sequence of “emoji images”. For this example, the true solution is unavailable, and we only provide a qualitative assessment. Still, this example clearly illustrates the impact of incorporating temporal information in the reconstruction process.

Quality measures and stopping criteria. To assess the quality of the reconstructed solution, we compute the Relative Reconstruction Errors (RREs) obtained using the ℓ_2-error norms. That is, for some recovered $\mathbf{u}_{(k)}$ at the kth iteration, the RRE is defined as follows:

$$\text{RRE} := \text{RRE}(\mathbf{u}_{(k)}, \mathbf{u}_{\text{true}}) = \frac{||\mathbf{u}_{(k)} - \mathbf{u}_{\text{true}}||_2}{||\mathbf{u}_{\text{true}}||_2}.$$

In addition to the RRE, in some examples, we report the Peak Signal to Noise Ratio (PSNR) (from MATLAB) and the Structural SIMilarity index (SSIM) between $\mathbf{u}_{(k)}$ and $\mathbf{u}_{\text{true}}$ to measure the quality of the computed approximate solutions. For the definition of the SSIM, we refer to [66] for details. Briefly, the SSIM measures how well the overall structure of the image is recovered; the higher the index, the better the reconstruction. The highest achievable value is 1.

The iterations are terminated as soon as the maximum number of iterations is reached or one of the following criteria is satisfied

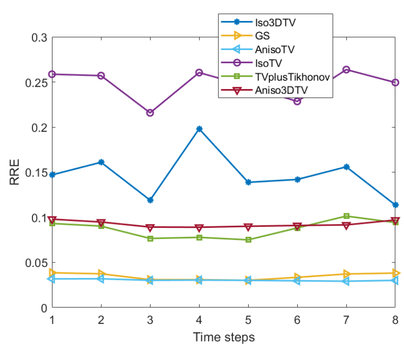
$$\text{(6.1)} \qquad \text{(i)} \quad \frac{\|\mathbf{u}_{(k)} - \mathbf{u}_{(k-1)}\|_2}{\|\mathbf{u}_{(k-1)}\|_2} \leq \mathrm{tol}_1\,, \qquad \text{(ii)} \quad \frac{\|\mathbf{r}_{(k+1)}\|_2}{\|\mathbf{r}_{(1)}\|_2} \leq \mathrm{tol}_2\,,$$

with $\mathrm{tol}_1 = 9 \times 10^{-4}$ and $\mathrm{tol}_2 = 10^{-5}$. Criteria (i) and (ii) monitor the relative change of two consecutive iterations and the relative reduction in the residual (5.2), respectively. We also experimented with two other stopping criteria: the discrepancy principle and the relative change in the regularization parameter, which are not reported in our numerical results. For consistency, in all the numerical examples, we set $\ell = 5$, that is, we run five iterations of the Golub-Kahan bidiagonalization algorithm to generate an initial subspace. We choose the smoothing parameter $\epsilon = 10^{-3}$. In the synthetic data examples (Examples 1 and 2), we perturb the measurements with white Gaussian noise, i.e., the noise vector $\mathbf{e}$ appearing in (1.1) has mean zero and a rescaled identity covariance matrix; we refer to the ratio $\sigma = \|\mathbf{e}\|_2/\|\mathbf{Fu}\|_2$ as the noise level.

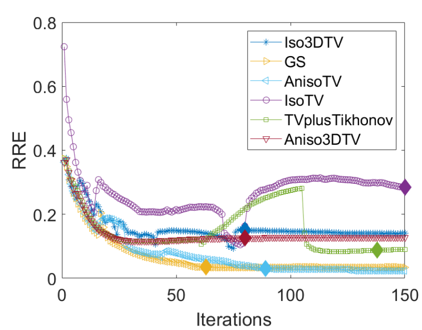
All the timing results were run on a Mac Mini (M1, 2020) with 16 GB RAM running MacOS Big Sur and MATLAB 2021a.

Example 1: space-time image deblurring. The goal here is to reconstruct a sequence of approximations of desired images from a sequence of blurry and noisy images. A sample of the true images is shown in the first row of Figure 6.2. The simulated available data are obtained by blurring eight images of size 128×128 with a Gaussian point spread function with a medium blur using [28]. We consider all the operators $\mathbf{A} = \mathbf{A}^{(t)} \in \mathbb{R}^{16{,}384\times 16{,}384}$, $t = 1, 2, \ldots, 8$ to be the same, so that $\mathbf{F} = \mathbf{I}_8 \otimes \mathbf{A} \in \mathbb{R}^{131{,}072\times 131{,}072}$.

The blurred images are perturbed with white Gaussian noise of level $\sigma = 0.01$ and are shown in the second row of Figure 6.2. We solve (3.10), where the index $j = 1, 2, \ldots, 6$ corresponds to (all the) methods listed in Table 3.1. Some quantitative results are displayed in Figure 6.1.



(a)



(b)

FIGURE 6.1. *Space-time image deblurring test problem: a) RRE computed for the dynamic problem at the iteration when the stopping criteria are first satisfied for each time step. b) History of RRE (all time steps together) for 150 iterations. The methods considered here are AnisoTV, TVplusTikhonov, IsoTV, Aniso3DTV, Iso3DTV, and GS. The solid diamond markers highlight the iteration satisfying the stopping criteria.*

Figure 6.1(a) displays the RRE at each of the eight time points for each method. The RRE is computed at the iteration k when the stopping criteria (6.1) are first satisfied; the number of iterations k and the regularization parameter $\lambda^{(k)}$ that was chosen are displayed in Table 6.1; note that we estimate the corresponding regularization parameter at each MM-GKS iteration.

ETNA
Kent State University and
Johann Radon Institute (RICAM)

In Figure 6.1(b), we display the RRE versus the number of iterations for all the methods when each method is allowed to run for 150 iterations without considering any other stopping criteria. Solid diamond markers over the lines in Figure 6.1(a) indicate the iteration and the value of the RRE (for each time slice) when the stopping criteria are satisfied. In contrast, each line in Figure 6.1(b) gives the RRE for each method for all images together, that is, the error in $\mathbf{u}_{(k)}$.

We observe that AnisoTV and GS outperform the other methods for this example in terms of RRE. Moreover, as illustrated in Figure 6.1, for the methods IsoTV[1], Iso3DTV, and TVplusTikhonov, we observe an increase of the RRE in the early iterations, but if the method is let to run enough iterations, then the RRE values start to stabilize (for all methods except for IsoTV). Reconstructions with AnisoTV at time steps $t = 1, 4, 5, 6, 7$ are displayed in the third row of Figure 6.2.

TABLE 6.1
Space-time image deblurring example: the number of iterations when a stopping criterion is satisfied for the first time and the corresponding regularization parameters for the considered methods.

Method	AnisoTV	TVplusTikhonov	IsoTV	Aniso3DTV	Iso3DTV	GS
Iterations k	89	138	150*	69	80	63
$\lambda^{(k)}$	0.163	0.126	0.295	0.007	0.168	0.114

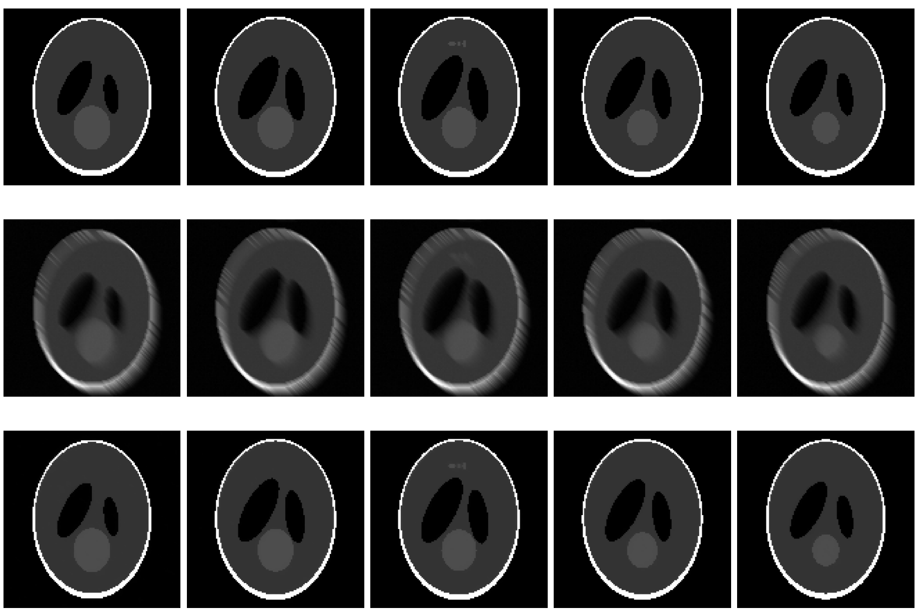


FIGURE 6.2. *Space-time image deblurring test problem: the first row represents a sample of true images at time steps $t = 1, 4, 5, 6, 7$. The second row shows the respective blurred any noisy images with medium blur and Gaussian noise of level $\sigma = 0.01$. The third row shows the reconstructed images $\mathbf{u}^{(t)}$, $t = 1, 4, 5, 6, 7$ obtained by AnisoTV when the stopping criteria are satisfied.*

[1] IsoTV stopped by the maximum number of iterations (150) for this example, but we highlight that when we slightly increase the tolerance of the stopping criteria, we observed convergence within 150 iterations.

ETNA
Kent State University and
Johann Radon Institute (RICAM)

Example 2: dynamic photoacoustic tomography (PAT). As a second example, we consider a dynamic instance of PAT, which is a hybrid imaging modality that combines the rich contrast of optical imaging with the high resolution of ultrasound imaging; dynamical PAT models were already considered in [22, 23, 52]. Specifically, the forward operator $\mathbf{F}$ is time-dependent and has the block-diagonal structure (1.2). The operators $\mathbf{A}^{(t)}$ are computed by using the `PRspherical` function from [28], for the projection angles t, $t+30,\ldots,$ $t+241$, $t=1,2,\ldots,n_t$. We add white Gaussian noise of level $\sigma = 0.01$ to the available measurements.

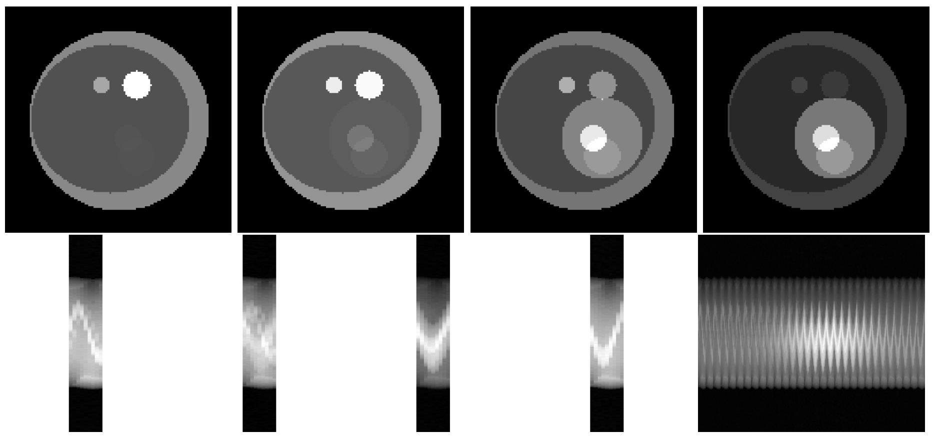
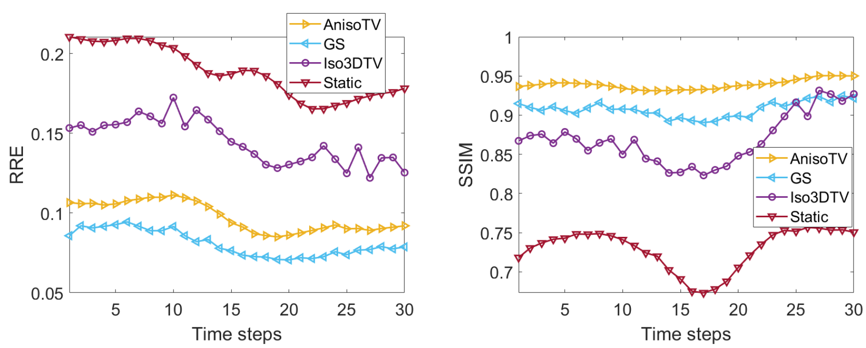


FIGURE 6.3. *Dynamic PAT test problem: first row, from left to right: true images at time steps* $t = 1, 10, 20, 30$. *Second row, from left to right: sample of sinograms at time steps* $t = 1, 10, 20, 30$ *and the full sinogram.*



(a) (b)

FIGURE 6.4. *Dynamic PAT test problem: a) RRE computed at the iteration (reported in Table 6.2) when the stopping criteria are first satisfied, for each time step. b) SSIM computed at the iteration when the stopping criteria are first satisfied, for each time step. The methods considered here are AnisoTV (right-pointing triangle line), Iso3DTV (dotted line), and GS (left-pointing triangle line).*

In a first instance, we consider images $\mathbf{U}^{(t)}$, $t = 1, 2, \ldots, 30$ of size 256×256 where each image represents a superposition of six circular objects that are in motion. This implies that the total number of unknowns is $n = 256 \times 256 \times 30 = 1,966,080$, leading to a severely underdetermined inverse problem. For each angle there are 362 measurements, resulting in a total of $m = 97,740$ observations. A sample of true images at time instances $t = 1, 10, 20, 30$

is shown in the first row of Figure 6.3, and the corresponding noisy sinograms $\mathbf{d}^{(t)} \in \mathbb{R}^{3,258}$ along with the full sinogram (obtained by concatenating all 30 available sinograms together) are shown in the second row of Figure 6.3. We carry out the following numerical experiments:

(a) Solve the large-scale dynamic inverse problem (3.10) with $j = 1, 4, 6$. More specifically, we choose AnisoTV from the anisotropic-type methods, Iso3DTV from the isotropic-type methods, and GS.

(b) Solve the static inverse problem (1.4) with the regularization term

$$\mathcal{R}(\mathbf{u}) = \|\mathbf{L}_s \mathbf{u}^{(t)}\|_1 \quad \text{at } \; t = 1, 2, \ldots, 30,$$

accounting for spatial regularization only. Throughout this paper, we solve the static inverse problem (1.4) by the MM-GKS algorithm where the regularization parameter is adapted at each iteration and all the stopping criteria are set the same as for solving the dynamic inverse problem (3.10).

We compute the $\text{RRE}(\mathbf{u}^{(k)}, \mathbf{u}_{\text{true}})$ as well as the SSIM for both experimental setups as described in (a) and (b) above, and we report the results in Figure 6.4 when the stopping criteria (6.1) are satisfied for the first time. The number of iterations k and the corresponding regularization parameter $\lambda^{(k)}$ are reported in Table 6.2. GS outperforms all the methods in this experimental setup, followed by AnisoTV, as illustrated in Figure 6.5 by both the RRE and SSIM quality measures. Notice here that Iso3DTV is the least accurate method among the ones we propose; however, it still outperforms the static approach.

In Figure 6.5, we report the reconstructions at times steps $t = 1, 10, 20, 30$ from left to right, respectively (exact images are reported in Figure 6.3). Different rows correspond to reconstructions with different methods. The first row shows the reconstructions obtained by solving the static inverse problem (1.4), where we observe that even though the method is able to provide the locations of the inclusions, the detailed information of the inclusions is missing. The second row shows the reconstructions with Iso3DTV, where certainly the artifacts around the circular inclusions are present and the background appears perturbed as well. Improved reconstructions are observed in the third and the fourth rows of Figure 6.5, obtained by AnisoTV and GS, respectively.

TABLE 6.2
Dynamic PAT test problem: the number of iterations k when the stopping criteria is satisfied for the first time and the corresponding regularization parameter $\lambda^{(k)}$ at those iterations for AnisoTV, Iso3DTV, and GS.

Method	AnisoTV	Iso3DTV	GS
Iterations k	84	81	91
$\lambda^{(k)}$	0.0073	0.0073	0.0073

Comparing different solvers. We consider 30 images of size 128×128 to compare the new solvers with other solvers based on the IRLS (or, equivalently, MM) strategies (2.4) and primal-dual-type methods such as ADMM. For simplicity, during the comparison, we only display results for the AnisoTV regularization term (3.1); the other regularizers listed in Table 3.1 provide very similar results in terms of accuracy and computational time.

Specifically, we consider the so-called

(c) MM-LSQR method: In this approach, we use LSQR [56] to solve the sequence of least-squares problems (3.6), written in the augmented form. We allow for 30 MM iterations and limit the number of inner iterations to a maximum of 100 or stop if the tolerance of 10^{-5} is achieved for the solution obtained by LSQR. We select the best (i.e., the one that produces the smallest RRE) regularization parameter out of 15 candidate values.

ETNA
Kent State University and
Johann Radon Institute (RICAM)

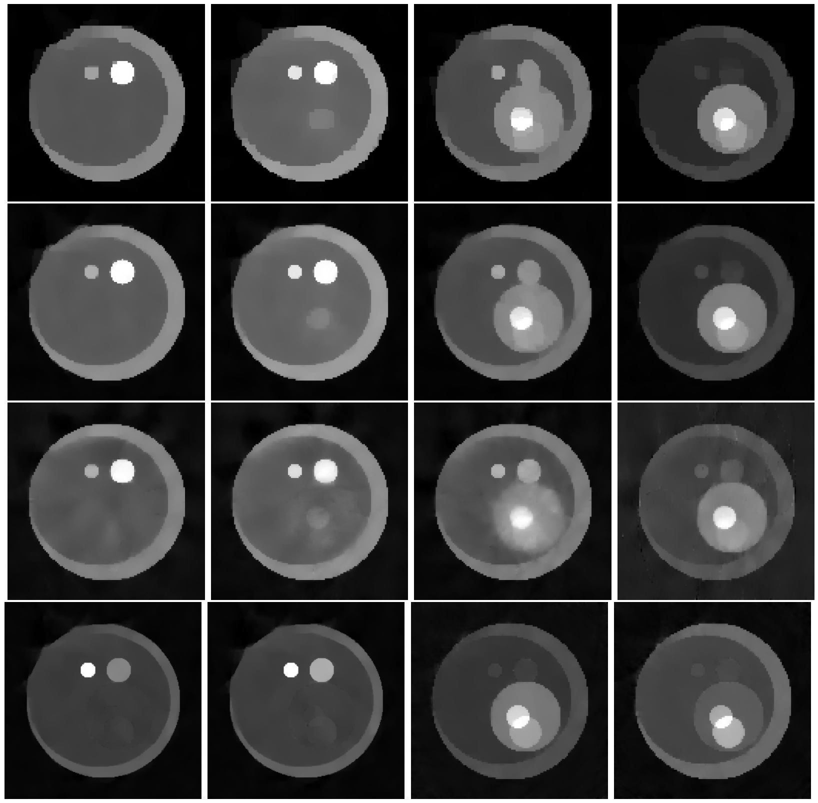


FIGURE 6.5. *Dynamic PAT test problem: panels in the first row show the reconstructions by solving the static inverse problem, panels in the second, third, and fourth rows show reconstructions with AnisoTV, Iso3DTV, and GS at time steps* $t = 1, 10, 20, 30$ *from left to right, respectively.*

(d) Inner-outer reweighting scheme: We follow the inner-outer approach introduced in [29], where the authors present an IRLS approach that uses an adaptive diagonal weighting matrix that shares some common features with spatial anisotropic TV involving the discrete spatial gradient operator $\mathbf{L}_s$ (2.3) and a projection-based iterative method developed in [45] to solve the corresponding sequence of general-form Tikhonov problems. We extended this approach to spatio-temporal TV by considering the spatio-temporal first-derivative operator $\mathbf{D}_1$ (3.1) rather than $\mathbf{L}_s$. We set a maximum number of outer iterations to 30 and limit the number of inner iterations to 60 and consider two different methods for estimating the regularization parameter at each inner iteration: the discrepancy principle and the L-curve. We call this approach IRN-aTV.

(e) ADMM: We consider here a primal-dual solver for (1.3) such as the Alternating Direction Method of Multipliers (ADMM) [9].
When ADMM is employed to solve the minimization problem (1.3) with a regularization term (for instance $\mathcal{R}_1(\mathbf{u})$) and a fixed regularization parameter, the main computational cost at its kth iteration is sourced from solving a linear system of equations of the form

$$(\mathbf{F}^T\mathbf{F} + \omega\mathbf{D}_1^T\mathbf{D}_1)\mathbf{u} = \mathbf{F}^T\mathbf{d} + \mathbf{D}_1^T\boldsymbol{\mu}_k + \omega\mathbf{D}_1^T\mathbf{c}_k \,, \tag{6.2}$$

where $\boldsymbol{\mu}_k$ is the current Lagrange multiplier, $\mathbf{c}_k$ is a current auxiliary variable, and

ETNA
Kent State University and
Johann Radon Institute (RICAM)

$\omega > 0$ is a penalty parameter for the Lagrangian. This is followed by the application of a proximal operator. We follow the approach in [33], but it is adapted to our setup. We let the maximum number of iterations of ADMM be 150 (the same maximum number of iterations as in AnisoTV). We solve the linear system (6.2) using LSQR for which we stop the iterations when the tolerance 10^{-5} or the maximum number of iterations 100 is reached. The regularization parameter is selected after searching for a regularization parameter that minimizes the RRE over 15 candidate runs.

We are somewhat limited in the solvers we can compare the methods in (a) against; this is because, while many methods are applicable to standard-form Tikhonov regularization, far fewer methods are applicable to general-form Tikhonov regularization (which is needed to solve (2.4)), which are further limited by the requirement that the regularization parameter λ should be ideally estimated during the reconstruction process.

In Table 6.3 we list the RRE, PSNR, the number of iterations, and the CPU time (in hours) for the anisotropic-TV-like methods described in (a), (c), (d), and (e).

TABLE 6.3
Dynamic PAT test problem: comparison of different solvers in terms of RRE, PSNR, Iterations (number of outer iterations), either MM or ADMM) and CPU time. CPU time includes the effort to find the best regularization parameter (over 15 candidate runs) for MM-LSQR and ADMM.

	MM-GKS	IRN-aTV (DP)	IRN-aTV (L-curve)	MM-LSQR	ADMM
RRE	0.096	0.0712	0.082	0.1879	0.087
PSNR	36.1	38.7	37.5	30.3	37.1
Iterations	85	30	30	30	99
CPU time (hours)	0.17	4.42	1.99	2.49	3.71

We make the following observations:

1. We clearly see that MM-LSQR is not competitive either in run time or the reconstruction error. Incrementing the number of inner and outer iterations will likely reduce the RRE, but it will increase the computational cost further.
2. The IRN-aTV methods have slightly lower RRE but considerably higher run times than MM-GKS. We did not investigate how to effectively stop the inner and outer iterations and used the implementation in IR Tools.
3. The ADMM yields a relatively low reconstruction error. For each regularization parameter value, the algorithm is fairly efficient and takes ~ 0.25 hours, but computed over 15 candidate runs it takes 3.71 hours. Note that the time to run ADMM for one (known) regularization parameter is still $\sim 50\%$ more expensive than MM-GKS.

To explain these observations, we note that the MM-GKS approach is more efficient for a comparable accuracy because, unlike the other three methods, 1) it is not an inner-outer method and 2) the regularization parameter is determined automatically at a negligible computational cost. More precisely, at each iteration, MM-GKS only increments the current basis for the solution by one vector to approximately solve each reweighted least-squares problem in the sequence (2.4) rather than computing a new basis from scratch. In contrast, the IRN-aTV method technically involves three levels of iterations: the outermost iterations update the weights needed for edge-preserving regularization, while the inner iterations used to solve the resulting general-form Tikhonov problem involve themselves an inner set of LSQR iterations [45].

Furthermore, MM-GKS is able to estimate the regularization parameter during the reconstruction (unlike MM-LSQR and ADMM), avoiding the need for a repeated inner-outer loop over all candidate regularization parameters. This numerical experiment highlights why the

ETNA
Kent State University and
Johann Radon Institute (RICAM)

MM-GKS approach is efficient in this context despite the potentially large orthogonalization cost.

We further remark that although MM-GKS is competitive with other methods we consider, the number of basis vectors that need to be stored in MM-GKS grows with the number of iterations. In large-scale problems (as in the current example), memory capacity can be easily reached, and we may not be able to run enough MM-GKS iterations to converge. A remedy to memory limitations was restarting, introduced in [14]. Other efficient strategies include recycling the subspace, which is a subject of future research.

Example 3: dynamic X-Ray Tomography-3D Emoji Data. In this example, we test our methods for real data of an "emoji" phantom measured at the University of Helsinki [53]. The forward operator and the data can be obtained from the file `DataDynamic_128x30.mat`. The available data represents $n_t = 33$ time steps of a series of the X-ray sinogram of emojis made of small ceramic stones obtained by shining 217 projections from $n_a = 30$ angles. The inverse problem involves reconstructing a sequence of images $\mathbf{U}^{(t)}$, $t = 1, 2, \ldots, 33$, of size $n_h \times n_v$, where $n_h = n_v = 128$, from low-dose observations measured from a limited number of angles n_a. These images represent the dynamic sequence of emojis varying from an expressionless face with closed eyes and a straight mouth to a face with smiling eyes and mouth, where the outermost circular shape does not change. As a result, the unknown images are collected in $\mathbf{u} = \left[(\mathbf{u}^{(1)})^T, (\mathbf{u}^{(2)})^T, \ldots, (\mathbf{u}^{(33)})^T\right]^T \in \mathbb{R}^{540,672}$. See the first row of Figure 6.6 for a sample of 4 images at time steps $t = 6, 14, 20, 26$. The low-dose available observations can be modeled by the measurement matrix $\mathbf{F}$ which describes the forward model of the Radon transform that represents line integrals. In this case, we have a block matrix $\mathbf{F}$ as in (1.2) with 33 blocks. Although the ground truth is not available, we can qualitatively compare the visual results.

We visualize the reconstructions from different numbers of angles $n_a = 10$ and $n_a = 30$, highlighting the effect of the number of the projection angles and also the visual differences in the reconstruction when static sub-problems (1.4) are solved independently and when the dynamic inverse problem (3.10) is solved. For each case, in Table 6.4, we report the number of iterations that the method took to converge and the regularization parameter at that iteration.

Case 1: considering $n_a = 10$ projection angles. First, we limit the number of angles n_a to 10 from the dataset `DataDynamic_128x30.mat`. In this way we generate underdetermined problems $\mathbf{A}^{(t)}\mathbf{u}^{(t)} + \mathbf{e}^{(t)} = \mathbf{d}^{(t)}$, $t = 1, 2, \ldots, 33$ where $\mathbf{A}^{(t)} \in \mathbb{R}^{2,170\times 16,384}$. Therefore $\mathbf{F} \in \mathbb{R}^{71,610\times 540,672}$ and the measurement vector $\mathbf{d} \in \mathbb{R}^{71,610}$ contains the measured sinograms $\mathbf{d}^{(t)} \in \mathbb{R}^{2,170}$ obtained from 217 projections around 10 equidistant angles.

Figure 6.6 displays some reconstructions (see also the supplementary materials[2] for an animation). Looking at the second row of images, it is evident that a limited number of projection angles per time step results in poor reconstructions when solving the static inverse problem, where the important details (features of the face) are missing. Solving the dynamic inverse problem results in an enhanced quality of the reconstruction. In particular, by considering the new regularization terms in AnisoTV (third row) and IsoTV3D (fourth row), we are able to reconstruct the edges clearly and have fewer artifacts overall.

Case 2: considering $n_a = 30$ projection angles. In this second case, we consider the full number of angles in the dataset `DataDynamic_128x30.mat`, i.e., $n_a = 30$ to highlight the importance of the number of projection angles. Here $\mathbf{A}^{(t)} \in \mathbb{R}^{6,510\times 16,384}$ and the measured sinograms are obtained from 217 projections with 30 angles each, that is, $\mathbf{d}^{(t)} \in \mathbb{R}^{6,510}$. Hence $\mathbf{F} \in \mathbb{R}^{214,830\times 540,672}$ and $\mathbf{d} \in \mathbb{R}^{214,830}$. The reconstructions of the static problems (1.4) are displayed in the second row of Figure 6.7. The third and

[2] https://etna.ricam.oeaw.ac.at/volumes/2021-2030/vol58/addition/p486.php

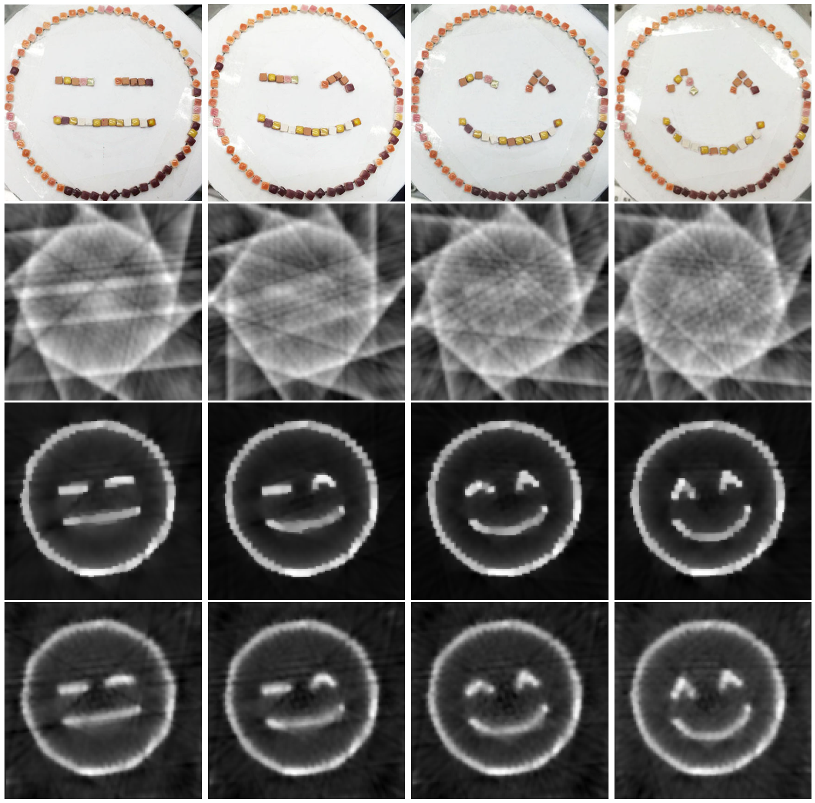


FIGURE 6.6. *Reconstruction results for the emoji test problem with* $n_a = 10$*. The rows represent (from top to bottom): the original images, the reconstructions when images are considered independently, the reconstructions by AnisoTV, the reconstructions by Iso3DTV, at time steps* $t = 2, 10, 18, 31$ *(from left to right).*

the fourth rows display the reconstruction by AnisoTV and Iso3DTV at the time instances $t = 6, 14, 20, 26$ from left to right, respectively. The first remark is that similar to the case $n_a = 10$, the reconstructions obtained using the dynamic inverse problem are qualitatively better than that of the static inverse problem. In addition, we observe that increasing the number of projection angles from 10 to 30 helps in removing the background artifacts and better preserving the edges.

We remark that other methods such as TVplusTikhonov, IsoTV, and Aniso3DTV produce reconstructions of similar quality to AnisoTV and Iso3DTV, and, therefore, we do not report them here. In contrast to the other test problems that we presented above, where GS was one of the most accurate methods, it (qualitatively) appears to be the least accurate one in this example. This observation allows us to highlight one of the goals of this paper, which is to present a variety of regularization methods without advocating for one over the other, since the performance of the methods we describe is application-dependent.

Nonnegativity constraints. In many applications, such as medical imaging and astronomical imaging, the pixels of the desired solution are nonnegative [12, 11, 32], that is, the exact solution of (1.3) is known to live in the closed and convex set

$$\Omega_0 = \{\mathbf{u} \in \mathbb{R}^n : (\mathbf{u})_\ell \geq 0, \ \ \ell = 1, 2, \ldots, n\}.$$

In general, imposing nonnegativity helps mitigate the artifacts from limited angles. Here we consider the optimization problems (3.10) subject to the constraint $\mathbf{u} \in \Omega_0$. This is

ETNA
Kent State University and
Johann Radon Institute (RICAM)

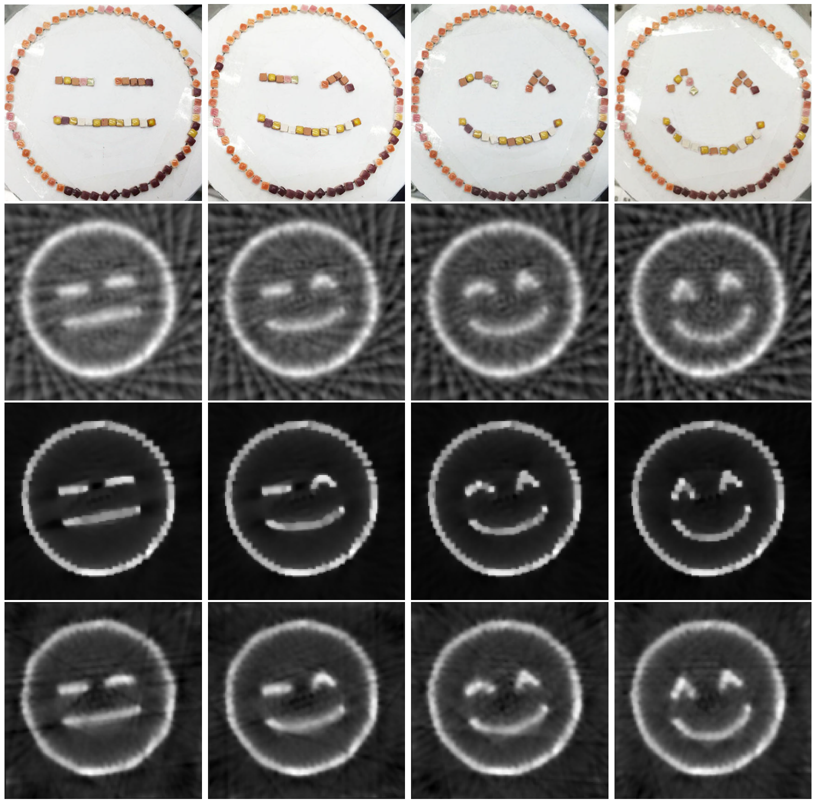


FIGURE 6.7. *Reconstruction results for the emoji test problem with $n_a = 30$. The rows represent (from top to bottom): the original images, the reconstructions when images are considered independently, the reconstructions by AnisoTV, and the reconstructions by Iso3DTV, at time steps $t = 2, 10, 18, 31$ (from left to right).*

TABLE 6.4
Dynamic X-Ray Tomography example: the number of iterations when the stopping criteria are satisfied for the first time and the regularization parameters at those iterations for AnisoTV and Iso3DTV.

	Method	AnisoTV	Iso3DTV
$n_a = 10$	**Iterations** k	115	63
	$\lambda^{(k)}$	0.515	0.7935
$n_a = 30$	**Iterations** k	86	94
	$\lambda^{(k)}$	0.796	1.109

heuristically implemented by projecting the solution $\mathbf{u}_{(k)}$ onto Ω_0 at each iteration. We illustrate the effect of the nonnegativity constraint in Example 3 for *Case 1*, with the number of projection angles taken to be $n_a = 10$ and with observations $\mathbf{d}^{(t)}$, $t = 1, 2, \ldots, 33$. The reconstructed images at time steps $t = 6, 14, 20, 26$ are displayed in Figure 6.8. From visual inspection, there are fewer artifacts around the edges when the nonnegativity constraint is applied.

7. Conclusions and future directions. In this paper, we proposed a unified approach for solving large-scale dynamic inverse problems and providing solutions with edge-preserving and sparsity-promoting properties. The approaches we discussed here are grouped into isotropic TV methods (which include IsoTV and Iso3DTV), anisotropic TV (which include AnisoTV and an Aniso3DTV), and another set of methods based on the concept of group

ETNA
Kent State University and
Johann Radon Institute (RICAM)

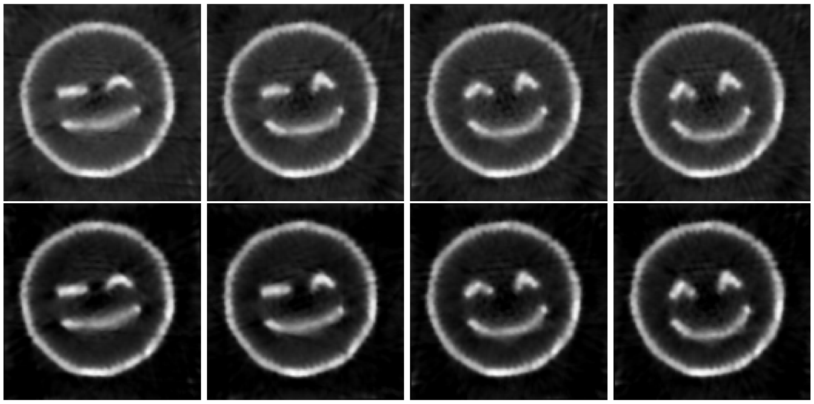


FIGURE 6.8. *Reconstruction results for the emoji test problem with $n_a = 10$. The first row shows the reconstructions with Iso3DTV for the unconstrained problem, and the second row shows the reconstructions with nonnegativity Iso3DTV at time steps $n_t = 6, 14, 20, 26$ respectively from left to right.*

sparsity, GS. All the methods can be expressed in a unified framework using the MM technique, where the resulting least-squares problem can be solved in a generalized Krylov subspace of a relatively small dimension, and the regularization parameter can be estimated efficiently. Several numerical examples, performed on both synthetic and real data, illustrate the performances of the proposed methods in terms of the quality of the reconstructed solutions. Although we propose a unified and generic framework that can be used to solve a wide range of dynamic inverse problems, there are quite a few potential directions to investigate for future work (see also Section 4). One direction of interest worth emphasizing is to further investigate the use of multiple regularization parameters, for instance, regularization parameters for the temporal and spatial domain or adapted regularization parameters for different channels. Another direction includes alternative formulations along with their Bayesian interpretation and uncertainty quantification. Moreover, it is known that tensor formulations preserve the structure of the data. Hence we are interested in investigating efficient tensor-based regularization methods [63]. Some other applications of interest include video reconstruction, multi-channel X-ray spectral tomography, and moving object detection.

Supplementary Material. The supplementary material accompanying this article is an animation (MP4 format) illustrating the regularization of the dynamic problem in Example 3 and can be found at
https://etna.ricam.oeaw.ac.at/volumes/2021-2030/vol58/addition/p486.php.

Acknowledgments. This work was initiated as a part of the Statistical and Applied Mathematical Sciences Institute (SAMSI) Program on Numerical Analysis in Data Science in 2020. Any opinions, findings, conclusions, or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the National Science Foundation (NSF). The authors thank Profs. Misha Kilmer, Andrew Brown, and Lothar Reichel for the constructive discussions and suggestions. MP gratefully acknowledges support from the NSF under award No. 2202846. AKS would like to acknowledge partial support from NSF through the awards DMS-1845406 and DMS-1720398. The work of SG is partially supported by EPSRC under grant EP/T001593/1. The work by EdS was supported in part by NSF grant DMS 1720305.

Appendix A. Optimization problems and MM approaches. In this appendix, we provide the details of the MM approaches for the optimization problems corresponding to the

ETNA
Kent State University and
Johann Radon Institute (RICAM)

regularization terms: TVplusTikhonov, Aniso3DTV, Iso3DTV, IsoTV, GS.

A.1. TVplusTikhonov. We solve the inverse problem (1.1) by solving the optimization problem:

$$\min_{\mathbf{u}\in\mathbb{R}^n} \mathcal{J}_2(\mathbf{u}) := \mathcal{F}(\mathbf{u}) + \lambda\mathcal{R}_2(\mathbf{u}), \tag{A.1}$$

where $\lambda > 0$. To achieve this, we can apply the MM approach similar to Section 3.1. In particular, we consider the smoothed version $\mathcal{R}_{2\epsilon}(\mathbf{u})$ of $\mathcal{R}_2(\mathbf{u})$, where the smoothing is applied only to the first term in (3.7); the corresponding smoothed objective function is denoted by $\mathcal{J}_{2\epsilon}(\mathbf{u})$. To derive a quadratic tangent majorant for $\mathcal{J}_{2\epsilon}(\mathbf{u})$, we only need to majorize its first term, so that we obtain

$$\mathcal{Q}_2(\mathbf{u};\mathbf{u}_{(k)}) := \mathcal{F}(\mathbf{u}) + \frac{\lambda}{2}\|\mathbf{M}_2^{(k)}\mathbf{u}\|_2^2 + c_2, \tag{A.2}$$

where c_2 is a constant independent of $\mathbf{u}$, and the matrix $\mathbf{M}_2^{(k)}$ is defined as

$$\mathbf{M}_2^{(k)} := \begin{bmatrix} \mathbf{W}_2^{(k)} & \\ & \mathbf{I} \end{bmatrix} \mathbf{D}_1, \tag{A.3}$$

with $\mathbf{D}_1$ as in (3.1). The weighting matrix $\mathbf{W}_2^{(k)}$ is defined as

$$\mathbf{W}_2^{(k)} = \operatorname{diag}\left((\mathbf{D}_1\mathbf{u}_{(k)})^2 + \epsilon^2\right)^{-1/4}.$$

As in (3.6), to solve the optimization problem (A.1), we solve a sequence of reweighted least-squares problems with the objective function $\mathcal{Q}_2$ defined in (A.2).

A.2. Aniso3DTV. The problem that we want to solve can be formulated as

$$\min_{\mathbf{u}\in\mathbb{R}^n} \mathcal{J}_3(\mathbf{u}) := \mathcal{F}(\mathbf{u}) + \lambda\mathcal{R}_3(\mathbf{u}),$$

which can be tackled with the MM approach similar to the one described in Section 3.1. Again, we consider the smoothed version $\mathcal{R}_{3\epsilon}(\mathbf{u})$ of $\mathcal{R}_3(\mathbf{u})$; the corresponding smoothed objective function is denoted by $\mathcal{J}_{3\epsilon}(\mathbf{u})$. We majorize $\mathcal{J}_{3\epsilon}(\mathbf{u})$ by the quadratic tangent majorant

$$\mathcal{Q}_3(\mathbf{u};\mathbf{u}_{(k)}) := \mathcal{F}(\mathbf{u}) + \frac{\lambda}{2}\|\mathbf{M}_3^{(k)}\mathbf{u}\|_2^2 + c_3,$$

where c_3 is a constant independent of $\mathbf{u}$ and

$$\mathbf{M}_3^{(k)} = \mathbf{W}_3^{(k)}\mathbf{D}_3, \qquad \text{where} \qquad \mathbf{W}_3^{(k)} = \operatorname{diag}(((\mathbf{D}_3\mathbf{u}_{(k)})^2 + \epsilon^2)^{-1/4}). \tag{A.4}$$

A.3. Iso3DTV. We have the following problem

$$\min_{\mathbf{u}\in\mathbb{R}^n} \mathcal{J}_4(\mathbf{u}) := \mathcal{F}(\mathbf{u}) + \lambda\mathcal{R}_4(\mathbf{u}).$$

We first consider, instead of $\mathcal{R}_4(\mathbf{u})$, the smoothed regularization term

$$\mathcal{R}_{4\epsilon}(\mathbf{u}) := \sum_{\ell=1}^{n_v n_h n_t} \sqrt{(\bar{\mathbf{z}}_v(\mathbf{u}))_\ell^2 + (\bar{\mathbf{z}}_h(\mathbf{u}))_\ell^2 + (\bar{\mathbf{z}}_t(\mathbf{y}))_\ell^2 + \epsilon^2}$$

and the corresponding objective function $\mathcal{J}_{4\epsilon}(\mathbf{u})$. Following the derivation in [69], we devise weights to be used in an MM approach to Iso3DTV. We can define the quadratic tangent majorant $\mathcal{Q}_4(\mathbf{u};\mathbf{u}_{(k)})$ for the objective function $\mathcal{J}_{4\epsilon}(\mathbf{u})$ as

$$\mathcal{Q}_4(\mathbf{u};\mathbf{u}_{(k)}) := \mathcal{F}(\mathbf{u}) + \frac{\lambda}{2}\|\mathbf{M}_4^{(k)}\mathbf{u}\|_2^2 + c_4,$$

where c_4 is a constant independent of $\mathbf{u}$, and $\mathbf{M}_4^{(k)}$ is the weighted matrix

$$\mathbf{M}_4^{(k)} := \mathbf{W}_4^{(k)}\mathbf{D}_4, \qquad \text{with} \qquad \mathbf{D}_4 := \begin{bmatrix} \mathbf{I}_{n_t}\otimes\mathbf{I}_{n_h}\otimes\bar{\mathbf{L}}_v \\ \mathbf{I}_{n_t}\otimes\bar{\mathbf{L}}_h\otimes\mathbf{I}_{n_v} \\ \bar{\mathbf{L}}_t\otimes\mathbf{I}_{n_h}\otimes\mathbf{I}_{n_v} \end{bmatrix}, \tag{A.5}$$

and

$$\mathbf{W}_4^{(k)} = \mathbf{I}_3 \otimes \operatorname{diag}\left(\left((\bar{\mathbf{z}}_v(\mathbf{u}_{(k)}))^2 + (\bar{\mathbf{z}}_h(\mathbf{u}_{(k)}))^2 + (\bar{\mathbf{z}}_t(\mathbf{u}_{(k)}))^2 + \epsilon^2\right)^{-1/4}\right),$$

where $(\bar{\mathbf{z}}_d(\mathbf{u}_{(k)}))$ are the vectors $\bar{\mathbf{z}}_d(\mathbf{u})$ in (3.8), $d = v, h, t$, evaluated at $\mathbf{u} = \mathbf{u}_{(k)}$, i.e., at the kth iteration. Finally, the matrix $\mathbf{D}_4$ is similar to $\mathbf{D}_1$ defined in (3.1), with the augmented derivative matrices $\bar{\mathbf{L}}_d$ instead of $\mathbf{L}_d$.

A.4. IsoTV. We have the following problem

$$\min_{\mathbf{u}\in\mathbb{R}^n} \mathcal{J}_5(\mathbf{u}) = \min_{\mathbf{u}\in\mathbb{R}^n} \mathcal{F}(\mathbf{u}) + \lambda\mathcal{R}_5(\mathbf{u}).$$

We define a smoothed version of $\mathcal{R}_5(\mathbf{u})$, denoted by $\mathcal{R}_{5\epsilon}(\mathbf{u})$, where the smoothing is applied separately to the first and second terms; the corresponding smoothed objective function is denoted by $\mathcal{J}_{5\epsilon}(\mathbf{u})$. We can then define the quadratic tangent majorant $\mathcal{Q}_5(\mathbf{u};\mathbf{u}_{(k)})$ for the objective function $\mathcal{J}_{5\epsilon}(\mathbf{u})$ as

$$\mathcal{Q}_5(\mathbf{u};\mathbf{u}_{(k)}) := \mathcal{F}(\mathbf{u}) + \frac{\lambda}{2}\|\mathbf{M}_5^{(k)}\mathbf{u}\|_2^2 + c_5.$$

The constant c_5 independent of $\mathbf{u}$, and $\mathbf{M}_5^{(k)}$ is the weighted matrix

$$\mathbf{M}_5^{(k)} := \mathbf{W}_5^{(k)}\mathbf{D}_5, \tag{A.6}$$

with

$$\mathbf{D}_5 := \begin{bmatrix} \mathbf{I}_{n_t}\otimes\mathbf{I}_{n_h}\otimes\bar{\mathbf{L}}_v \\ \mathbf{I}_{n_t}\otimes\bar{\mathbf{L}}_h\otimes\mathbf{I}_{n_v} \\ \mathbf{L}_t\otimes\mathbf{I}_{n_h}\otimes\mathbf{I}_{n_v} \end{bmatrix} \qquad \text{and} \qquad \mathbf{W}_5^{(k)} = \begin{bmatrix} \mathbf{I}_2\otimes\operatorname{diag}\left(\mathbf{w}^s_{(k)}\right) & \\ & \operatorname{diag}\left(\mathbf{w}^t_{(k)}\right) \end{bmatrix},$$

where

$$\mathbf{w}^s_{(k)} = \left((\bar{\mathbf{z}}_v(\mathbf{u}_{(k)}))^2 + (\bar{\mathbf{z}}_h(\mathbf{u}_{(k)}))^2 + \epsilon^2\right)^{-1/4} \qquad \text{and} \qquad \mathbf{w}^t_{(k)} = \left((\mathbf{z}_t(\mathbf{u}_{(k)}))^2 + \epsilon^2\right)^{-1/4}.$$

Here $\bar{\mathbf{z}}_d(\mathbf{u}_{(k)})$ are again the vectors $\bar{\mathbf{z}}_d(\mathbf{u})$ in (3.8), $d = v, h$, evaluated at $\mathbf{u} = \mathbf{u}_{(k)}$, i.e., at the kth iteration.

A.5. GS. Corresponding to the regularization operator $\mathcal{R}_6$, we can define the optimization problem:

$$\min_{\mathbf{u}\in\mathbb{R}^n} \mathcal{J}_6(\mathbf{u}) := \mathcal{F}(\mathbf{u}) + \lambda\mathcal{R}_6(\mathbf{u}),$$

where $\lambda > 0$. We can apply the MM approach similar to Section 3.1. We now seek a quadratic tangent majorant for a smoothed version of $\mathcal{R}_6(\mathbf{u})$. To this end, let $\mathbf{u}_{(k)}$ be the current iterate (similarly, define $\mathbf{z}_{(k)} = (\mathbf{I}_{n_t} \otimes \mathbf{L}_s)\mathbf{u}_{(k)}$). Then, we have that

$$\mathcal{R}_6(\mathbf{u}) \leq \sum_{\ell=1}^{n'_s} \sqrt{\|\mathbf{z}_\ell\|_2^2 + \epsilon^2} =: \mathcal{R}_{6\epsilon}(\mathbf{u}) \leq \sum_{\ell=1}^{n'_s} \frac{\|\mathbf{z}_\ell\|_2^2}{2\sqrt{\|(\mathbf{I}_{n_t} \otimes \mathbf{e}_\ell^T \mathbf{L}_s)\mathbf{u}_{(k)}\|_2^2 + \epsilon^2}} + \tilde{c}_6,$$

where $\tilde{c}_6$ is a constant independent of $\mathbf{z}_\ell$ and $\mathbf{u}$. The corresponding smoothed optimization function is defined as $\mathcal{J}_{6\epsilon}(\mathbf{u})$. Let us define the weighting matrix $\mathbf{W}_6^{(k)}$ of size $n'_s \times n'_s$ as

$$\mathbf{W}_6^{(k)} := \operatorname{diag}\left(\frac{1}{\sqrt{\|(\mathbf{I}_{n_t} \otimes \mathbf{e}_1^T \mathbf{L}_s)\mathbf{u}_{(k)}\|_2^2 + \epsilon^2}}, \ldots, \frac{1}{\sqrt{\|(\mathbf{I}_{n_t} \otimes \mathbf{e}_{n'_s}^T \mathbf{L}_s)\mathbf{u}_{(k)}\|_2^2 + \epsilon^2}}\right)^{1/2}.$$

We can use this weighting matrix to define the quadratic tangent majorant

$$\mathcal{Q}_6(\mathbf{u}; \mathbf{u}_{(k)}) := \mathcal{F}(\mathbf{u}) + \frac{\lambda}{2}\|\mathbf{M}_6^{(k)}\mathbf{u}\|_2^2 + c_6,$$

where $c_6 = \lambda\tilde{c}_6$ and the matrix $\mathbf{M}_6^{(k)}$ takes the form

$$\text{(A.7)} \qquad \mathbf{M}_6^{(k)} := (\mathbf{I}_{n_t} \otimes \mathbf{W}_6^{(k)})\mathbf{D}_6, \qquad \text{with} \qquad \mathbf{D}_6 := (\mathbf{I}_{n_t} \otimes \mathbf{L}_s).$$

REFERENCES

[1] S. Achenbach, T. Giesler, D. Ropers, S. Ulzheimer, H. Derlien, C. Schulte, E. Wenkel, W. Moshage, W. Bautz, W. G. Daniel, W. A. Kalender, and U. Baum, *Detection of coronary artery stenoses by contrast-enhanced, retrospectively electrocardiographically-gated, multislice spiral computed tomography*, Circulation, 103 (2001), pp. 2535–2538.

[2] S. T. Acton, *Diffusion partial differential equations for edge detection*, in The Essential Guide to Image Processing, A. Bovik, ed., Academic Press, Burlington, 2009, pp. 525–552.

[3] F. Bach, *Exploring large feature spaces with hierarchical multiple kernel learning*, in Advances in Neural Information Processing Systems 21 (NIPS 2008), D. Koller, D. Schuurmans, Y. Bengio, and L. Bottou, eds., Curran Associates, Red Hook, 2008, pp. 105–112.

[4] F. Bach, R. Jenatton, J. Mairal, and G. Obozinski, *Optimization with sparsity-inducing penalties*, Found. Trends Mach. Learn., 4 (2012), pp. 1–106.

[5] O. Balabanov and L. Grigori, *Randomized Gram-Schmidt process with application to GMRES*, SIAM J. Sci. Comput., 44 (2022), pp. A1450–A1474.

[6] M. Belge, M. E. Kilmer, and E. L. Miller, *Efficient determination of multiple regularization parameters in a generalized L-curve framework*, Inverse Problems, 18 (2002), pp. 1161–1183.

[7] Å. Björck, *Numerical Methods for Least Squares Problems*, SIAM, Philadelphia, 1996.

[8] C. Blondel, R. Vaillant, G. Malandain, and N. Ayache, *3D tomographic reconstruction of coronary arteries using a precomputed 4D motion field*, Phys. Med. & Biology, 49 (2004), pp. 2197–2208.

[9] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, *Distributed optimization and statistical learning via the alternating direction method of multipliers*, Found. Trends Mach. Learn., 3 (2011), pp. 1–122.

[10] T. A. Bubba, T. Heikkilä, H. Help, S. Huotari, Y. Salmon, and S. Siltanen, *Sparse dynamic tomography: a shearlet-based approach for iodine perfusion in plant stems*, Inverse Problems, 36 (2020), Art. 094002, 34 pages.

[11] A. Buccini, M. Pasha, and L. Reichel, *Modulus-based iterative methods for constrained ℓ_p–ℓ_q minimization*, Inverse Problems, 36 (2020), Art. 084001, 26 pages.
[12] ———, *Linearized Krylov subspace Bregman iteration with nonnegativity constraint*, Numer. Algorithms, 87 (2021), pp. 1177–1200.
[13] A. Buccini and L. Reichel, *Generalized cross validation for ℓ^p-ℓ^q minimization*, Numer. Algorithms, 88 (2021), pp. 1595–1616.
[14] ———, *Limited memory restarted ℓ^p-ℓ^q minimization methods using generalized Krylov subspaces*, Adv. Comput. Math., 49 (2023), Art. 26, 26 pages.
[15] M. Burger, H. Dirks, L. Frerking, A. Hauptmann, T. Helin, and S. Siltanen, *A variational reconstruction method for undersampled dynamic x-ray tomography based on physical motion models*, Inverse Problems, 33 (2017), Art. 124008, 24 pages.
[16] J.-F. Cai, S. Osher, and Z. Shen, *Split Bregman methods and frame based image restoration*, Multiscale Model. Simul., 8 (2009/10), pp. 337–369.
[17] D. Calvetti, S. Morigi, L. Reichel, and F. Sgallari, *Tikhonov regularization and the L-curve for large discrete ill-posed problems*, J. Comput. Appl. Math., 123 (2000), pp. 423–446.
[18] E. J. Candès, J. Romberg, and T. Tao, *Robust uncertainty principles: exact signal reconstruction from highly incomplete frequency information*, IEEE Trans. Inform. Theory, 52 (2006), pp. 489–509.
[19] V. Caselles, A. Chambolle, and M. Novaga, *Total variation in imaging*, in Handbook of Mathematical Methods in Imaging. Vol. 1, 2, 3, O. Scherzer, ed., Springer, New York, 2015, pp. 1455–1499.
[20] S. H. Chan, R. Khoshabeh, K. B. Gibson, P. E. Gill, and T. Q. Nguyen, *An augmented Lagrangian method for total variation video restoration*, IEEE Trans. Image Process., 20 (2011), pp. 3097–3111.
[21] R. Chartrand, *Exact reconstruction of sparse signals via nonconvex minimization*, IEEE Signal Process. Lett., 14 (2007), pp. 707–710.
[22] J. Chung and L. Nguyen, *Motion estimation and correction in photoacoustic tomographic reconstruction*, SIAM J. Imaging Sci., 10 (2017), pp. 216–242.
[23] J. Chung, A. K. Saibaba, M. Brown, and E. Westman, *Efficient generalized Golub-Kahan based methods for dynamic inverse problems*, Inverse Problems, 34 (2018), Art. 024005, 29 pages.
[24] B. Cui, X. Ma, X. Xie, G. Ren, and Y. Ma, *Classification of visible and infrared hyperspectral images based on image segmentation and edge-preserving filtering*, Infrared Phys. & Techn., 81 (2017), pp. 79–88.
[25] I. Daubechies, *Ten Lectures on Wavelets*, SIAM, Philadelphia, 1992.
[26] M. Donatelli and L. Reichel, *Square smoothing regularization matrices with accurate boundary conditions*, J. Comput. Appl. Math., 272 (2014), pp. 334–349.
[27] H. W. Engl, M. Hanke, and A. Neubauer, *Regularization of Inverse Problems*, Kluwer, Dordrecht, 1996.
[28] S. Gazzola, P. C. Hansen, and J. G. Nagy, *IR Tools: a MATLAB package of iterative regularization methods and large-scale test problems*, Numer. Algorithms, 81 (2019), pp. 773–811.
[29] S. Gazzola, M. E. Kilmer, J. G. Nagy, O. Semerci, and E. L. Miller, *An inner-outer iterative method for edge preservation in image restoration and reconstruction*, Inverse Problems, 36 (2020), Art. 124004, 33 pages.
[30] S. Gazzola and P. Novati, *Multi-parameter Arnoldi-Tikhonov methods*, Electron. Trans. Numer. Anal., 40 (2013), pp. 452–475.
https://etna.ricam.oeaw.ac.at/vol.40.2013/pp452-475.dir/pp452-475.pdf
[31] S. Gazzola and L. Reichel, *A new framework for multi-parameter regularization*, BIT Numer. Math., 56 (2016), pp. 919–949.
[32] S. Gazzola and Y. Wiaux, *Fast nonnegative least squares through flexible Krylov subspaces*, SIAM J. Sci. Comput., 39 (2017), pp. A655–A679.
[33] A. Gholami and S. Gazzola, *Automatic balancing parameter selection for Tikhonov-TV regularization*, BIT Numer. Math., 62 (2022), pp. 1873–1898.
[34] E. Giraldo, J. Castaño-Candamil, and G. Castellanos-Dominguez, *A weighted dynamic inverse problem for electroencephalographic current density reconstruction*, in 2013 6th International IEEE/EMBS Conference on Neural Engineering (NER), IEEE Conference Proceedings, Los Alamitos, 2013, pp. 521–524.
[35] G. González, V. Kolehmainen, and A. Seppänen, *Isotropic and anisotropic total variation regularization in electrical impedance tomography*, Comput. Math. Appl., 74 (2017), pp. 564–576.
[36] F. Guo, C. Zhang, and M. Zhang, *Edge-preserving image denoising*, IET Image Process., 12 (2018), pp. 1394–1401.
[37] P. C. Hansen, *Rank-Deficient and Discrete Ill-Posed Problems*, SIAM, Philadelphia, 1998.
[38] A. Hauptmann, O. Öktem, and C. Schönlieb, *Image reconstruction in dynamic inverse problems with temporal models*, in Handbook of Mathematical Models and Algorithms in Computer Vision and Imaging—Mathematical Imaging and Vision, K. Chen, C.-B. Schönlieb, X.-C. Tai, and L. Younes, eds., Springer, Cham, 2023, pp. 1707–1737.
[39] K. He, D. Wang, and X. Zheng, *Image segmentation on adaptive edge-preserving smoothing*, J. Electron. Imaging, 25 (2016), Art. 053022, 14 pages.

[40] W. S. HOGE, M. E. KILMER, C. ZACARIAS-ALMARCHA, AND D. H. BROOKS, *Fast regularized reconstruction of non-uniformly subsampled partial-Fourier parallel MRI data*, in 2007 4th IEEE International Symposium on Biomedical Imaging: From Nano to Macro, IEEE Conference Proceedings, Los Alamitos, 2007, pp. 1012–1015.

[41] G. HUANG, A. LANZA, S. MORIGI, L. REICHEL, AND F. SGALLARI, *Majorization-minimization generalized Krylov subspace methods for ℓ_p-ℓ_q optimization applied to image restoration*, BIT Numer. Math., 57 (2017), pp. 351–378.

[42] D. R. HUNTER AND K. LANGE, *A tutorial on MM algorithms*, Amer. Statist., 58 (2004), pp. 30–37.

[43] J. KAIPIO AND E. SOMERSALO, *Statistical and Computational Inverse Problems*, Springer, New York, 2005.

[44] D. KAZANTSEV, J. S. JØRGENSEN, M. S. ANDERSEN, W. R. B. LIONHEART, P. D. LEE, AND P. J. WITHERS, *Joint image reconstruction method with correlative multi-channel prior for x-ray spectral computed tomography*, Inverse Problems, 34 (2018), Art. 064001, 26 pages.

[45] M. E. KILMER, P. C. HANSEN, AND M. I. ESPAÑOL, *A projection-based approach to general-form Tikhonov regularization*, SIAM J. Sci. Comput., 29 (2007), pp. 315–330.

[46] S. KIM AND E. P. XING, *Tree-guided group Lasso for multi-task regression with structured sparsity*, in ICML'10 Proceedings of the 27th International Conference on International Conference on Machine Learning, J. Fürnkranz and T. Joachims, eds., Ominpress, Madison, 2010, pp. 543–550.

[47] T. G. KOLDA AND B. W. BADER, *Tensor decompositions and applications*, SIAM Rev., 51 (2009), pp. 455–500.

[48] J. LAMPE, L. REICHEL, AND H. VOSS, *Large-scale Tikhonov regularization via reduction by orthogonal projection*, Linear Algebra Appl., 436 (2012), pp. 2845–2865.

[49] K. LANGE, *MM Optimization Algorithms*, SIAM, Philadelphia, 2016.

[50] T. LI, E. SCHREIBMANN, Y. YANG, AND L. XING, *Motion correction for improved target localization with on-board cone-beam computed tomography*, Phys. Med. Biol., 51 (2005), pp. 253–267.

[51] Y. LOU, T. ZENG, S. OSHER, AND J. XIN, *A weighted difference of anisotropic and isotropic total variation model for image processing*, SIAM J. Imaging Sci., 8 (2015), pp. 1798–1823.

[52] F. LUCKA, N. HUYNH, M. BETCKE, E. ZHANG, P. BEARD, B. COX, AND S. ARRIDGE, *Enhancing compressed sensing 4D photoacoustic tomography by simultaneous motion estimation*, SIAM J. Imaging Sci., 11 (2018), pp. 2224–2253.

[53] A. MEANEY, Z. PURISHA, AND S. SILTANEN, *Tomographic X-ray data of 3D emoji*, Preprint on arXiv, 2018. https://arxiv.org/abs/1802.09397

[54] Y. MEYER, *Wavelets and Operators*, Cambridge University Press, Cambridge, 1992.

[55] J. L. MUELLER AND S. SILTANEN, *Linear and Nonlinear Inverse Problems with Practical Applications*, SIAM, Philadelphia, 2012.

[56] C. C. PAIGE AND M. A. SAUNDERS, *LSQR: an algorithm for sparse linear equations and sparse least squares*, ACM Trans. Math. Software, 8 (1982), pp. 43–71.

[57] C. PAL, A. CHAKRABARTI, AND R. GHOSH, *A brief survey of recent edge-preserving smoothing algorithms on digital images*, Preprint on arXiv, 2015. https://arxiv.org/abs/1503.07297.

[58] R. A. RENAUT, S. VATANKHAH, AND V. E. ARDESTANI, *Hybrid and iteratively reweighted regularization by unbiased predictive risk and weighted GCV for projected systems*, SIAM J. Sci. Comput., 39 (2017), pp. B221–B243.

[59] L. I. RUDIN, S. OSHER, AND E. FATEMI, *Nonlinear total variation based noise removal algorithms*, Phys. D, 60 (1992), pp. 259–268.

[60] A. K. SAIBABA, J. CHUNG, AND K. PETROSKE, *Efficient Krylov subspace methods for uncertainty quantification in large Bayesian linear inverse problems*, Numer. Linear Algebra Appl., 27 (2020), Art. e2325, 28 pages.

[61] U. SCHMITT AND A. K. LOUIS, *Efficient algorithms for the regularization of dynamic inverse problems. I. Theory*, Inverse Problems, 18 (2002), pp. 645–658.

[62] U. SCHMITT, A. K. LOUIS, C. WOLTERS, AND M. VAUHKONEN, *Efficient algorithms for the regularization of dynamic inverse problems. II. Applications*, Inverse Problems, 18 (2002), pp. 659–676.

[63] O. SEMERCI, N. HAO, M. E. KILMER, AND E. L. MILLER, *Tensor-based formulation and nuclear norm regularization for multienergy computed tomography*, IEEE Trans. Image Process., 23 (2014), pp. 1678–1693.

[64] C. R. VOGEL, *Computational Methods for Inverse Problems*, SIAM, Philadelphia 2002.

[65] C. R. VOGEL AND M. E. OMAN, *Fast, robust total variation-based reconstruction of noisy, blurred images*, IEEE Trans. Image Process., 7 (1998), pp. 813–824.

[66] Z. WANG, A. C. BOVIK, H. R. SHEIKH, AND E. P. SIMONCELLI, *Image quality assessment: From error visibility to structural similarity*, IEEE Trans. Image Process., 13 (2004), pp. 600–612.

[67] E. WESTMAN, K. LUXBACHER, AND S. SCHAFRIK, *Passive seismic tomography for three-dimensional time-lapse imaging of mining-induced rock mass changes*, The Leading Edge, 31 (2012), pp. 338–345.

[68] R. WILLIAMS AND M. S. BECK, *Process Tomography: Principles, Techniques and Applications*, Butterworth-Heinemann, Oxford, 1995.

[69] B. WOHLBERG AND P. RODRIGUEZ, *An iteratively reweighted norm algorithm for minimization of total variation functionals*, IEEE Signal Process. Lett., 14 (2007), pp. 948–951.

[70] Z. XU, X. CHANG, F. XU, AND H. ZHANG, *$L_{1/2}$ regularization: a thresholding representation theory and a fast solver*, IEEE Trans. Neural Netw. Learn. Syst., 23 (2012), pp. 1013–1027.

[71] X. YANG, S. YAO, K. P. LIM, X. LIN, S. RAHARDJA, AND F. PAN, *An adaptive edge-preserving artifacts removal filter for video post-processing*, in 2005 IEEE International Symposium on Circuits and Systems, IEEE Conference Proceedings, Los Alamitos, 2005, pp. 4939–4942.

[72] W. YIN, D. GOLDFARB, AND S. OSHER, *The total variation regularized L^1 model for multiscale decomposition*, Multiscale Model. Simul., 6 (2007), pp. 190–211.

[73] H. ZHANG, S. SARKAR, M. N. TOKSÖZ, H. S. KULELI, AND F. AL-KINDY, *Passive seismic tomography using induced seismicity at a petroleum field in Oman*, Geophysics, 74 (2009), pp. WCB57–WCB69.