

Generalized Higher Degree Total Variation (HDTV) Regularization

Yue Hu[†], *Student Member, IEEE*, Greg Ongie[†], *Student Member, IEEE*, Sathish Ramani, *Member, IEEE*
and Mathews Jacob, *Senior Member, IEEE*

Abstract

We introduce a family of novel image regularization penalties called generalized higher degree total variation (HDTV). These penalties further extend our previously introduced HDTV penalties, which generalize the popular total variation (TV) penalty to incorporate higher degree image derivatives. We show that many of the proposed second degree extensions of TV are special cases or are closely approximated by a generalized HDTV penalty. Additionally, we propose a novel fast alternating minimization algorithm for solving image recovery problems with HDTV and generalized HDTV regularization. The new algorithm enjoys a ten-fold speed up compared to the iteratively reweighted majorize minimize algorithm proposed in a previous work. Numerical experiments on 3D magnetic resonance images and 3D microscopy images show that HDTV and generalized HDTV improve the image quality significantly compared with TV.

I. INTRODUCTION

The total variation (TV) image regularization penalty is widely used in many image recovery problems, including denoising, compressed sensing, and deblurring [1]. The good performance of the TV penalty may be attributed to its desirable properties such as convexity, invariance to rotations and translations, and ability to preserve image edges. However, the main challenges associated with this scheme are the undesirable patchy or staircase-like artifacts in reconstructed images, which arise because TV regularization promotes sparse gradients.

We recently introduced a family of novel image regularization penalties termed as higher degree TV (HDTV) to overcome the above problems [2]. These penalties are defined as the L_1 - L_p norm ($p = 1$ or 2) of the n th degree directional image derivatives. The HDTV penalties inherit the desirable properties of the TV functional mentioned above. Experiments on two-dimensional (2D) images demonstrate that HDTV regularization provides improved reconstructions, both visually and quantitatively. Notably, it minimizes the staircase and patchy artifacts characteristic of TV, while still enhancing edge and ridge-like features in the image. The HDTV penalties were originally designed

[†]Both authors have contributed equally to the paper. Y. Hu is with the Department of Electrical and Computer Engineering, University of Rochester, NY, USA. G. Ongie is with the Department of Mathematics, University of Iowa, IA, USA. S. Ramani is with GE Global Research Center, Niskayuna, NY, 12309. M. Jacob is with the Department of Electrical and Computer Engineering, University of Iowa, IA, USA.

This work is supported by grants NSF CCF-0844812, NSF CCF-1116067, NIH 1R21HL109710-01A1, ACS RSG-11-267-01-CCE, and ONR-N000141310202.

for 2D image reconstruction problems and were defined solely in terms of 2D directional derivatives. The direct extension of the current scheme to 3D is challenging due to the high computational complexity of our current implementation. Specifically, the iteratively reweighted majorize minimize (IRMM) algorithm that we used in [2] is considerably slower than state-of-the-art TV algorithms [3]–[5].

In this work we extend HDTV to higher dimensions and to a wider class of penalties based on higher degree differential operators, and devise an efficient algorithm to solve inverse problems with these penalties; we term the proposed scheme *generalized HDTV*. The generalized HDTV penalties are defined as the L_1 - L_p norm, $p \geq 1$, of all rotations of an n th degree differential operator. By design, the generalized HDTV penalties also inherit the desirable properties of TV and HDTV such as translation- and rotation-invariance, scale covariance, as well as convexity. Furthermore, generalized HDTV penalties allow for a diversity of image priors that behave differently in preserving or enhancing various image features—such as edges or ridges—and may be finely tuned for the specific image reconstruction task at hand. Our new algorithm is based on an alternating minimization scheme, which alternates between two efficiently solved subproblems given by a shrinkage and the inversion of a linear system. The latter subproblem is much simpler to solve if the measurement operator has a diagonal form in the Fourier domain, as is the case for many practical inverse problems, such as denoising, deblurring, and single coil compressed sensing magnetic resonance (MR) image recovery. We find that this new algorithm improves the convergence rate by a factor of ten compared to the IRMM scheme, making the framework comparable in run time to the state-of-the-art TV methods.

We study the relationship between the generalized HDTV scheme and existing second degree TV generalizations [6]–[12]. Specifically, we show that many of the second degree TV generalizations (e.g. Laplacian penalty [6]–[8], the Frobenius norm of the Hessian [9], [13], and the recently introduced Hessian-Shatten norms [12]) are special cases or equivalent to the proposed HDTV scheme, when the differential operator in the HDTV regularization is chosen appropriately. The main benefit of the generalized HDTV framework is that it extends to higher degree image derivatives ($n \geq 2$) and higher dimensions easily. Furthermore, our current implementation is considerably faster than many of the existing TV generalizations. We also observe that some of the current TV generalizations may result in poor reconstructions. For example, the penalties that promote the sparsity of the Laplacian operators have a large null space [14], and inverse problems regularized with such penalties may still be ill-posed. Moreover, Laplacian-based penalties are known to preserve point-like features rather than line-like features, which is also undesirable in many image reconstruction settings.

We compare the convergence of the proposed algorithm with our previous IRMM implementation. Our results show that the proposed scheme is around ten times faster than the IRMM method. We also demonstrate the utility of HDTV and generalized HDTV regularization in the context of practical inverse problems arising in medical imaging, including deblurring and denoising of 3D fluorescence microscope images, and compressed sensing MR image recovery of 3D angiography datasets. We show that 3D-HDTV routinely outperforms TV in terms of the SNR of reconstructed images and its ability to preserve ridge-like details in the datasets. We restrict our comparisons with TV since the implementations of many of the current extensions are only available in 2D; comparisons with

these methods are available in [2]. Moreover, some of the TV extensions, like total generalized variation [11], are hybrid methods that combines derivatives of different degrees. The proposed HDTV scheme may be extended in a similar fashion, but it is beyond the scope of the present work.

II. BACKGROUND

A. Image Recovery Problems

We consider the recovery of a continuously differentiable d -dimensional signal $f : \Omega \rightarrow \mathbb{C}$ from its noisy and degraded measurements \mathbf{b} . Here $\Omega \subset \mathbb{R}^d$ is the spatial support of the image. We model the measurements as $y = \mathcal{A}(f) + \eta$, where η is assumed to be Gaussian distributed white noise and \mathcal{A} is a linear operator representing the degradation process. For example, \mathcal{A} may be a blurring (or convolution) operator in the deconvolution setting, a Fourier domain undersampling operator in the case of compressed sensing MR images reconstruction, or identity in the case of denoising. The operator \mathcal{A} may be severely ill-conditioned or non-invertible, so that in general recovering f from its measurements requires some form of regularization of the image to ensure well-posedness. Hence, we formulate the recovery of f as the following optimization problem

$$\min_f \|\mathcal{A}(f) - \mathbf{b}\|^2 + \lambda \mathcal{J}(f), \quad (1)$$

where $\|\mathcal{A}(f) - \mathbf{b}\|^2$ is the data fidelity term, $\mathcal{J}(f)$ is a regularization penalty, and the parameter λ balances the two terms, and is chosen so that the signal-to-error ratio is maximized.

B. Two-dimensional HDTV

In 2D, the standard isotropic TV regularization penalty is the L_1 norm of the gradient magnitude, specified as

$$\text{TV}(f) = \int_{\Omega} |\nabla f(\mathbf{r})| d\mathbf{r} = \int_{\Omega} \sqrt{(\partial_x f(\mathbf{r}))^2 + (\partial_y f(\mathbf{r}))^2} d\mathbf{r}.$$

In [2] we showed that the 2D-TV penalty can be reinterpreted as the mixed L_1 - L_2 norm or the L_1 - L_1 of image directional derivatives. This observation led us to propose two families of HDTV regularization penalties in 2D, specified by

$$\text{I-HDTV}_n(f) = \int_{\Omega} \left(\frac{1}{2\pi} \int_0^{2\pi} |f_{\theta,n}(\mathbf{r})|^2 d\theta \right)^{\frac{1}{2}} d\mathbf{r}, \quad (2)$$

$$\text{A-HDTV}_n(f) = \int_{\Omega} \left(\frac{1}{2\pi} \int_0^{2\pi} |f_{\theta,n}(\mathbf{r})| d\theta \right) d\mathbf{r}, \quad (3)$$

where $f_{\theta,n}$ is the n th degree directional derivative operator in the direction $\mathbf{u}_{\theta} = [\cos(\theta), \sin(\theta)]$, defined as

$$f_{\theta,n}(\mathbf{r}) = \left. \frac{\partial^n}{\partial \gamma^n} f(\mathbf{r} + \gamma \mathbf{u}_{\theta}) \right|_{\gamma=0}.$$

The family of penalties defined by (2) and (3) were termed as *isotropic* and *anisotropic HDTV*, respectively. It is evident from (2) and (3) that 2D-HDTV penalties preserve many of the desirable properties of the standard TV penalty, such as invariance under translations and rotations and scale covariance. Furthermore, practical experiments in [2] demonstrate that HDTV regularization outperforms TV regularization in many image recovery tasks, in terms

of both SNR and the visual quality of reconstructed images. Our experiments also indicate that the anisotropic case, which corresponds to the fully separable L_1 - L_1 penalty, typically exhibits better performance in image recovery tasks over isotropic HDTV.

III. GENERALIZED HDTV REGULARIZATION PENALTIES

The 2D-HDTV penalties given in (2) and (3) may be described as penalizing all rotations in the plane of the n th degree differential operator ∂_x^n . This interpretation suggests an extension of HDTV to higher dimensions, and to a wider class of rotation invariant penalties based on n th degree image derivatives, by penalizing all rotations in d -dimensions of an arbitrary n th degree differential operator $\mathcal{D} = \sum_{|\alpha|=n} c_\alpha \partial^\alpha$, where α is a multi-index and the c_α are constants. Thus, given a specific n th degree differential operator \mathcal{D} , and $p \geq 1$, we define the *generalized HDTV penalty* for d -dimensional signals $f : \Omega \rightarrow \mathbb{C}$ as

$$\mathcal{G}[\mathcal{D}, p](f) = \int_{\Omega} \left(\int_{SO(d)} |\mathcal{D}_{\mathbf{U}} f(\mathbf{r})|^p d\mathbf{U} \right)^{\frac{1}{p}} d\mathbf{r}, \quad (4)$$

where $SO(d) = \{\mathbf{U} \in \mathbb{R}^{d \times d} : \mathbf{U}^T = \mathbf{U}^{-1}, \det \mathbf{U} = 1\}$ is the special orthogonal group, i.e. the group of all proper rotations in \mathbb{R}^d , and $\mathcal{D}_{\mathbf{U}}$ is the rotated operator defined as

$$\mathcal{D}_{\mathbf{U}} f(\mathbf{r}_0) = \mathcal{D}[f(\mathbf{r}_0 + \mathbf{U}\mathbf{r})] \Big|_{\mathbf{r}=\mathbf{0}}.$$

By design the generalized HDTV penalties are guaranteed to be rotation and translation invariant, and convex for all $p \geq 1$. It is also clear they are contrast and scale covariant, i.e. for all $\alpha \in \mathbb{R}$, $\mathcal{G}(\alpha \cdot f) = |\alpha| \mathcal{G}(f)$ and $\mathcal{G}(f_\alpha) = |\alpha|^{n-d} \mathcal{G}(f)$, where $f_\alpha(x) := f(\alpha \cdot x)$. Below we discuss some particular cases for which (4) affords simplifications.

1) *Generalized HDTV penalties in 2D*: The 2D rotation group $SO(2)$ can be identified with the unit circle $\mathbb{S}^1 = \{(\cos \theta, \sin \theta) : \theta \in [0, 2\pi]\}$, which allows us rewrite the integral in (4) as a one-dimensional integral over $\theta \in [0, 2\pi]$. By choosing $\mathcal{D} = \partial_x^n$ and $p = 2, 1$ we recover the isotropic and anisotropic HDTV penalties specified in (2) and (3). In this work we also consider 2D-HDTV penalties for arbitrary $p \geq 1$,

$$\text{HDTV}[n, p](f) = \int_{\Omega} \left(\frac{1}{2\pi} \int_0^{2\pi} |f_{\theta, n}(\mathbf{r})|^p d\theta \right)^{\frac{1}{p}} d\mathbf{r}. \quad (5)$$

For the $p = 1$ case we will simply write HDTV n , i.e. $\text{HDTV}n = \text{HDTV}[n, 1]$.

For a generalized HDTV penalty in 2D, where \mathcal{D} is an arbitrary n th degree differential operator, we may also write

$$\mathcal{G}[\mathcal{D}, p](f) = \int_{\Omega} \left(\frac{1}{2\pi} \int_0^{2\pi} |\mathcal{D}_{\theta} f(\mathbf{r})|^p d\theta \right)^{\frac{1}{p}} d\mathbf{r}, \quad (6)$$

where $\mathcal{D}_{\theta} f(\mathbf{r}_0) = \mathcal{D}[f(\mathbf{r}_0 + \mathbf{U}_{\theta}\mathbf{r})] \Big|_{\mathbf{r}=\mathbf{0}}$, and \mathbf{U}_{θ} is a coordinate rotation about the origin by the angle θ .

2) *Generalized HDTV penalties in 3D*: The 3D rotation group $SO(3)$ has a more complicated structure so that in general we cannot simplify the integral in (4). However, all rotations in 3D of the standard HDTV operator $\mathcal{D} = \partial_x^n$ are specified solely by the orientations $\mathbf{u} \in \mathbb{S}^2 = \{\mathbf{u} \in \mathbb{R}^3 : \|\mathbf{u}\| = 1\}$. Thus, in this case the integral in (4) simplifies to

$$\text{HDTV}[n, p](f) = \int_{\Omega} \left(\int_{\mathbb{S}^2} |f_{\mathbf{u}, n}(\mathbf{r})|^p d\mathbf{u} \right)^{\frac{1}{p}} d\mathbf{r}, \quad (7)$$

where $f_{\mathbf{u}, n}$ is the n th degree directional derivative defined as

$$f_{\mathbf{u}, n}(\mathbf{r}) = \left. \frac{\partial^n}{\partial \gamma^n} f(\mathbf{r} + \gamma \mathbf{u}) \right|_{\gamma=0}, \quad \mathbf{u} \in \mathbb{S}^2.$$

Likewise, if an operator \mathcal{D} is rotation symmetric about an axis, then all rotations of \mathcal{D} in 3D can be specified by the unit directions ¹ $\mathbf{u} \in \mathbb{S}^2$. For example, the 2D Laplacian $\Delta = \partial_{xx} + \partial_{yy}$ embedded in 3D is rotation symmetric about the z -axis, so that any rotation of Δ is specified solely by the $\mathbf{u} \in \mathbb{S}^2$ to which $\mathbf{z} = [0, 0, 1]$ is mapped. For this class of operator the integral in (4) simplifies to

$$\mathcal{G}[\mathcal{D}, p](f) = \int_{\Omega} \left(\int_{\mathbb{S}^2} |\mathcal{D}_{\mathbf{u}} f(\mathbf{r})|^p d\mathbf{u} \right)^{\frac{1}{p}} d\mathbf{r}, \quad (8)$$

where $\mathcal{D}_{\mathbf{u}} f(\mathbf{r}_0) = \mathcal{D}[f(\mathbf{r}_0 + \mathbf{U}\mathbf{r})]|_{\mathbf{r}=0}$ for any $\mathbf{U} \in SO(3)$ that maps the axis of symmetry of \mathcal{D} to \mathbf{u} .

3) *Rotation Steerability of Derivative Operators*: The direct evaluation of the above integrals by their discretization over the rotation group is computationally expensive. The computational complexity of implementing HDTV regularization penalties can be considerably reduced by exploiting the *rotation steerable* property of n th degree differential operators. Towards this end, note that the first degree directional derivatives $f_{\mathbf{u}, 1}$ have the equivalent expression

$$f_{\mathbf{u}, 1}(\mathbf{r}) = \mathbf{u}^T \nabla f(\mathbf{r}).$$

Similarly, higher degree directional derivatives $f_{\mathbf{u}, n}(\mathbf{r})$ can be expressed as the separable vector product

$$f_{\mathbf{u}, n}(\mathbf{r}) = \mathbf{s}(\mathbf{u})^T \nabla_n f(\mathbf{r}),$$

where, $\mathbf{s}(\mathbf{u})$ is vector of polynomials in the components of \mathbf{u} and $\nabla_n f(\mathbf{r})$ is the vector of all n th degree partial derivatives of f . For example, in the second degree case ($n = 2$) in 2D, we may choose

$$\mathbf{s}(\mathbf{u}) = [u_x^2, 2u_x u_y, u_y^2]^T; \quad \nabla_2 f(\mathbf{r}) = [f_{xx}(\mathbf{r}), f_{xy}(\mathbf{r}), f_{yy}(\mathbf{r})]^T. \quad (9)$$

In the case of a general n th degree differential operator \mathcal{D} , by repeated application of the chain rule we may write the rotated operator $\mathcal{D}_{\mathbf{U}}$ for any $\mathbf{U} \in SO(d)$ as

$$\begin{aligned} \mathcal{D}_{\mathbf{U}} f(\mathbf{r}_0) &= \mathcal{D} f(\mathbf{r}_0 + \mathbf{U}\mathbf{r})|_{\mathbf{r}=0} \\ &= \mathbf{s}(\mathbf{U})^T \nabla_n f(\mathbf{r}_0), \end{aligned} \quad (10)$$

¹This is due to Euler's rotation theorem [15] which states that every rotation in 3D can be represented as a rotation by an angle θ about a fixed axis $\mathbf{u} \in \mathbb{S}^2$.

where $\mathbf{s}(\mathbf{U})$ is a vector of polynomials in the components of \mathbf{U} , whose exact expression will depend on the choice of operator \mathcal{D} . For example, the second degree 2D operator $\mathcal{D} = \partial_{xx} + \alpha\partial_{yy}$ would have

$$\mathbf{s}(\mathbf{u}) = \left[u_x^2 + \alpha u_y^2, 2(1 - \alpha)u_x u_y, u_y^2 + \alpha u_x^2 \right]^T.$$

Note that (10) shows that the choice of differential operator \mathcal{D} defining a generalized HDTV penalty only amounts to a different choice of *steering function* $\mathbf{s}(\mathbf{U})$ as specified in (10). This property will be useful in deriving a unified discrete framework for generalized HDTV penalties.

We now study the relationship of the generalized HDTV scheme with second degree extensions of TV in the recent literature. We show that many of them are related to the second degree ($n = 2$) generalized HDTV penalties, when the derivative operator is chosen appropriately. Specifically, the general second degree derivative operator has the form $\mathcal{D} = \sum_{|\alpha|=2} c_\alpha \partial^\alpha$, and we will show that different choices of the coefficients c_α and p in (4) encompass many of the proposed regularizers.

A. Laplacian Penalty

One choice of \mathcal{D} is the Laplacian Δ , where $\Delta = \partial_{xx} + \partial_{yy}$ in 2D and $\Delta = \partial_{xx} + \partial_{yy} + \partial_{zz}$ in 3D. Note that the Laplacian is rotation invariant, so that $\Delta_{\mathbf{U}}f = \Delta f$ for all $\mathbf{U} \in SO(d)$. Thus, for any choice of p in (4), we obtain the penalty

$$\mathcal{G}_\Delta(f) = \int_\Omega |\Delta f(\mathbf{r})| d\mathbf{r},$$

which was introduced for image denoising in [6]. This penalty has two major disadvantages. First of all, it has a large null space. Specifically, any function that satisfies the Laplace equation ($\Delta f(\mathbf{r}) = 0$) will result in $\mathcal{G}_\Delta(f) = 0$. As a result, the use of this regularizer to constrain general ill-posed inverse problems is not desirable. Another problem is that due to the Laplacian being isotropic, its use as a penalty results in the enhancement of point-like features rather than line-like features.

B. Frobenius Norm of Hessian

Another interesting case corresponds to the second degree 2D operator $\mathcal{D} = \partial_{xx} - (2\sqrt{2} - 3)\partial_{yy}$. The corresponding isotropic ($p = 2$) generalized HDTV penalty is thus given by

$$\mathcal{G}[\mathcal{D}, 2](f) = \int_\Omega \left(\frac{1}{2\pi} \int_0^{2\pi} |f_{\theta,2}(\mathbf{r}) - (2\sqrt{2} - 3)f_{\theta^\perp,2}(\mathbf{r})|^2 d\theta \right)^{\frac{1}{2}} d\mathbf{r},$$

where $f_{\theta^\perp,2}(\mathbf{r})$ is the second derivative of f along $\theta^\perp = \theta + \frac{\pi}{2}$. Using the rotation steerability of second degree directional derivatives we have $f_{\theta,2}(\mathbf{r}) = f_{xx}(\mathbf{r}) \cos^2 \theta + f_{yy}(\mathbf{r}) \sin^2 \theta + 2f_{xy}(\mathbf{r}) \cos \theta \sin \theta$, and the expression for $\mathcal{G}[\mathcal{D}, 2](f)$ simplifies to

$$\mathcal{G}[\mathcal{D}, 2](f) = c \int_\Omega \sqrt{|f_{xx}(\mathbf{r})|^2 + |f_{yy}(\mathbf{r})|^2 + 2|f_{xy}(\mathbf{r})|^2} d\mathbf{r}, \quad (11)$$

for a constant c . This functional can be expressed as $\mathcal{G}[\mathcal{D}, 2](f) = \int_\Omega \|\mathcal{H}f\|_F d\mathbf{r}$, where $\mathcal{H}f$ is the Hessian matrix of $f(\mathbf{r})$ and $\|\cdot\|_F$ is the Frobenius norm. This second order penalty was proposed by [9], and can also be thought

of as the straightforward extension of the classical second-degree Duchon’s seminorm [14]. A similar argument shows an isotropic generalized HDTV penalty is equivalent to the Frobenius norm of the Hessian in 3D, as well.

C. Hessian-Shatten Norms

One family of second degree penalties that generalize (11) are the Hessian-Shatten norms recently introduced by Lefkimmiatis *et al.* in [12]. These penalties are defined as

$$\text{HS}_p(f) = \int_{\Omega} \|\mathcal{H}f(\mathbf{r})\|_{\mathcal{S}_p}, \quad \forall 1 \leq p \leq \infty, \quad (12)$$

where $\mathcal{H}f$ is the Hessian matrix of f , and $\|\cdot\|_{\mathcal{S}_p}$ is the Shatten p -norm defined as $\|X\|_{\mathcal{S}_p} = \|\sigma(X)\|_p$, where $\sigma(X)$ is a vector containing the singular values of the matrix X . The Schatten norm is equal to the Frobenius norm when $p = 2$, thus the penalty HS2 is equivalent to the generalized HDTV penalty $\mathcal{G}[\mathcal{D}, 2]$ given in (11). For $p \neq 2$, the Hessian-Shatten norms are not directly equivalent to a generalized HDTV penalty. However, there is a close relationship between the $p = 1$ case of (12) and the anisotropic second degree HDTV penalty, HDTV2. Specifically, we have the following proposition, which we prove in the Appendix:

Proposition 1. *The penalties HDTV2 and HS1 are equivalent as semi-norms over $\mathcal{C}^2(\Omega, \mathbb{R})$, $\Omega \subset \mathbb{R}^d$, in dimension $d = 2$ or 3, with bounds*

$$(1 - \delta) \cdot \text{HS1}(f) \leq C \cdot \text{HDTV2}(f) \leq \text{HS1}(f), \quad \forall f \in \mathcal{C}^2(\Omega, \mathbb{R}),$$

where $\delta = 0.37$ for $d = 2$, $\delta = 0.43$ for $d = 3$, and C is a normalization constant independent of f .

Note that in the Appendix we show $C \cdot \text{HDTV2}(f) = \text{HS1}(f)$ if the Hessian matrices of f at all spatial locations are either positive or negative semi-definite, i.e. have all non-negative eigenvalues or all non-positive eigenvalues. In natural images only a fraction of the pixels or voxels will have Hessian eigenvalues with mixed sign, thus we expect the HS1 and HDTV2 penalties to be nearly proportional and to behave very similarly in applications. Our experiments in the results section are consistent with this observation.

D. Benefits of HDTV

The preceding shows that the generalized HDTV penalties encompass many of the proposed image regularization penalties based on second degree derivatives, or in the case of certain Hessian-Shatten norms, closely approximate them. The generalized HDTV penalties have the additional benefit of being extendible to derivatives of arbitrary degree $n > 2$. Additionally, a wide class of image recovery problems regularized with any of the various HDTV penalties—regardless of dimension d , degree n , choice of differential operator \mathcal{D} —can all be put in a unified discrete framework, and solved with the same fast algorithm, which we introduce in the following section.

IV. FAST ALTERNATING MINIMIZATION ALGORITHM FOR HDTV REGULARIZED INVERSE PROBLEMS

A. Discrete Formulation

We now give a discrete formulation of the problem (1) with generalized HDTV regularization. In this setting we consider the recovery of a discrete d -dimensional image ($d = 2$ or 3), according to a linear observation model

$$\mathbf{b} = \mathbf{A}\mathbf{x} + \mathbf{n},$$

where matrix $\mathbf{A} \in \mathbb{C}^{M \times N}$ represents the linear degradation operator, $\mathbf{x} \in \mathbb{C}^N$ and $\mathbf{b} \in \mathbb{C}^M$ are vectorized versions of the signal to be recovered and its measurements, respectively, and $\mathbf{n} \in \mathbb{C}^M$ is a vector of Gaussian distributed white noise.

We represent the rotated discrete derivative operators $\mathcal{D}_{\mathbf{u}}$ for $\mathbf{u} \in \mathcal{S}$, where $\mathcal{S} = SO(d)$ or \mathbb{S}^{d-1} where appropriate, by block-circulant² $N \times N$ matrices $\mathbf{D}_{\mathbf{u}}$; that is, the multiplication $\mathbf{D}_{\mathbf{u}}\mathbf{x}$ corresponds to the multi-dimensional convolution of \mathbf{x} with a discrete filter approximating $\mathcal{D}_{\mathbf{u}}$. Thus, the image recovery problem (1) in the discrete setting with generalized HDTV regularization is given by

$$\min_{\mathbf{x}} \|\mathbf{A}\mathbf{x} - \mathbf{b}\|^2 + \lambda \int_{\mathcal{S}} \|\mathbf{D}_{\mathbf{u}}\mathbf{x}\|_{\ell_p} d\mathbf{u}. \quad (13)$$

In the case that $p > 1$, designing an efficient algorithm for solving (13) is challenging due to the non-separability of the generalized HDTV penalty. Moreover, our experiments from [2] indicate the anisotropic case $p = 1$ typically exhibits better performance in image recovery tasks over the $p > 1$ case. Accordingly, for our new algorithm we will focus on the $p = 1$ case:

$$\min_{\mathbf{x}} \|\mathbf{A}\mathbf{x} - \mathbf{b}\|^2 + \lambda \int_{\mathcal{S}} \|\mathbf{D}_{\mathbf{u}}\mathbf{x}\|_{\ell_1} d\mathbf{u}. \quad (14)$$

Note that the regularization is essentially the sum of absolute values of all the entries of the signals $\mathbf{D}_{\mathbf{u}}\mathbf{x}$. Extending our algorithm to general p , including the nonconvex case $0 < p < 1$, is reserved for a future work.

B. Algorithm

To realize a computationally efficient algorithm for solving (14), we modify the half-quadratic minimization method [16] used in TV regularization [3], [17] to the HDTV setting. We approximate the absolute value function inside the ℓ_1 norm with the *Huber function*:

$$\varphi_{\beta}(x) = \begin{cases} |x| - 1/2\beta & \text{if } |x| \geq \frac{1}{\beta} \\ \beta |x|^2 / 2 & \text{else .} \end{cases}$$

The approximate optimization problem for a fixed $\beta > 0$ is thus specified by

$$\min_{\mathbf{x}} \|\mathbf{A}\mathbf{x} - \mathbf{b}\|^2 + \lambda \int_{\mathcal{S}} \sum_{j=1}^N \varphi_{\beta}([\mathbf{D}_{\mathbf{u}}\mathbf{x}]_j) d\mathbf{u}. \quad (15)$$

²We assume periodic boundary conditions in the convolution. Hence the resulting matrix operator is circulant.

Here, $[x]_j$ denotes the j^{th} element of \mathbf{x} . Note that this approximation tends to the original HDTV penalty as $\beta \rightarrow \infty$. Our use of the Huber function is motivated by its half-quadratic dual relation [17]:

$$\varphi_\beta(y) = \min_z \left\{ \frac{\beta}{2} |y - z|^2 + |z| \right\}.$$

This enables us to equivalently express (15) as

$$\min_{\mathbf{x}, \mathbf{z}} \|\mathbf{A}\mathbf{x} - \mathbf{b}\|^2 + \lambda \int_{\mathcal{S}} \left\{ \frac{\beta}{2} \|\mathbf{D}_u \mathbf{x} - \mathbf{z}_u\|^2 + \|\mathbf{z}_u\|_{\ell_1} \right\} d\mathbf{u}, \quad (16)$$

where $\mathbf{z}_u \in \mathbb{C}^N$ are auxiliary variables that we also collect together as $\mathbf{z} \in \mathbb{C}^N \times \mathcal{S}$. We rely on an alternating minimization strategy to solve (16) for \mathbf{x} and \mathbf{z} . This results in two efficiently solved subproblems: the *z-subproblem*, which can be solved exactly in one shrinkage step, and the *x-subproblem* which involves the inversion of a linear system that often can be solved in one step using discrete Fourier transforms. The details of these two subproblems are presented below. To obtain solutions to the original problem (14), we rely on a continuation strategy on β . Specifically, we solve the problems (15) for a sequence $\beta_n \rightarrow \infty$, warm starting at each iteration with the solution to the previous iteration, and stopping the algorithm when the relative error of successive iterates is within a specified tolerance.

C. The *z*-subproblem: Minimization with respect to \mathbf{z} , assuming \mathbf{x} fixed

Assuming \mathbf{x} to be fixed, we minimize the cost function in (16) with respect to \mathbf{z} , which gives

$$\min_{\mathbf{z}} \int_{\mathcal{S}} \left\{ \frac{\beta}{2} \|\mathbf{D}_u \mathbf{x} - \mathbf{z}_u\|^2 + \|\mathbf{z}_u\|_{\ell_1} \right\} d\mathbf{u}.$$

Note that since the objective is separable in each component $[\mathbf{z}_u]_j$ of \mathbf{z}_u we may minimize the expression over each $[\mathbf{z}_u]_j$ independently. Thus, the exact solution to the above problem is given component-wise by the soft-shrinkage formula

$$[\mathbf{z}_u]_j = \max \left(|[\mathbf{D}_u \mathbf{x}]_j| - \frac{1}{\beta}, 0 \right) \frac{[\mathbf{D}_u \mathbf{x}]_j}{|[\mathbf{D}_u \mathbf{x}]_j|}, \quad (17)$$

where we follow the convention $0 \cdot \frac{0}{0} = 0$. While performing the shrinkage step in (17) is not computationally expensive, the need to store the auxiliary variables $\mathbf{z}_u \in \mathbb{C}^N$ for many $\mathbf{u} \in \mathcal{S}$ will make the algorithm very memory intensive. However, we will see in the next subsection that the rest of the algorithm does not need the variable \mathbf{z} explicitly, but only its projection onto a small subspace, which significantly reduces the memory demand.

D. The *x*-subproblem: Minimization with respect to \mathbf{x} , assuming \mathbf{z} to be fixed

Assuming that \mathbf{z} is fixed, we now minimize (16) with respect to \mathbf{x} , which yields

$$\min_{\mathbf{x}} \|\mathbf{A}\mathbf{x} - \mathbf{b}\|^2 + \frac{\lambda\beta}{2} \int_{\mathcal{S}} \|\mathbf{D}_u \mathbf{x} - \mathbf{z}_u\|^2 d\mathbf{u}.$$

The above objective is quadratic in \mathbf{x} , and from the normal equations its minimizer satisfies

$$\left(2\mathbf{A}^T \mathbf{A} + \lambda\beta \int_{\mathcal{S}} \mathbf{D}_u^T \mathbf{D}_u d\mathbf{u} \right) \mathbf{x} = 2\mathbf{A}^T \mathbf{b} + \lambda\beta \int_{\mathcal{S}} \mathbf{D}_u^T \mathbf{z}_u d\mathbf{u}. \quad (18)$$

We now exploit the steerability of the derivative operators to obtain simple expressions for the operator $\int_{\mathcal{S}} \mathbf{D}_{\mathbf{u}}^T \mathbf{D}_{\mathbf{u}} d\mathbf{u}$ and the vector $\int_{\mathcal{S}} \mathbf{D}_{\mathbf{u}}^T \mathbf{z}_{\mathbf{u}} d\mathbf{u}$. The discretization of (10) yields $\mathbf{D}_{\mathbf{u}} = \sum_j s_j(\mathbf{u}) \mathbf{E}_j$, where \mathbf{E}_j for $j = 1, \dots, P$ are circulant matrices corresponding to convolution with discretized partial derivatives and $s_i(\mathbf{u})$ are the steering functions. This expression can be compactly represented as $\mathbf{S}(\mathbf{u})\mathbf{E}$, where $\mathbf{E}^T = [\mathbf{E}_1^T, \dots, \mathbf{E}_P^T]$ and $\mathbf{S}(\mathbf{u}) = [s_1(\mathbf{u}) \mathbf{I}, s_2(\mathbf{u}) \mathbf{I}, \dots, s_P(\mathbf{u}) \mathbf{I}]$. Here, \mathbf{I} denotes the $N \times N$ identity matrix, where N is the number of pixels ($\mathbf{x} \in \mathbb{C}^N$). Thus,

$$\int_{\mathcal{S}} \mathbf{D}_{\mathbf{u}}^T \mathbf{D}_{\mathbf{u}} d\mathbf{u} = \mathbf{E}^T \underbrace{\int_{\mathcal{S}} \mathbf{S}(\mathbf{u})^T \mathbf{S}(\mathbf{u}) d\mathbf{u}}_{\mathbf{C}} \mathbf{E}.$$

The \mathbf{C} matrix is essentially the tensor product between $\mathbf{Q} = \int_{\mathcal{S}} \mathbf{s}(\mathbf{u})^T \mathbf{s}(\mathbf{u}) d\mathbf{u}$ and $\mathbf{I}_{N \times N}$. Hence we may write $\int_{\mathcal{S}} \mathbf{D}_{\mathbf{u}}^T \mathbf{D}_{\mathbf{u}} d\mathbf{u} = \sum_{i,j} q_{i,j} \mathbf{E}_i^T \mathbf{E}_j$, where $q_{i,j}$ is the (i, j) entry of \mathbf{Q} . Note that the matrix \mathbf{Q} can be computed exactly using expressions for the steering functions $\mathbf{s}(\mathbf{u})$. For example, with 2D-HDTV2 regularization (i.e. $\mathbf{s}(\mathbf{u})$ as given in (9)) one can show that up to a scaling factor,

$$\mathbf{Q} = \begin{bmatrix} 3 & 0 & 1 \\ 0 & 4 & 0 \\ 1 & 0 & 3 \end{bmatrix},$$

which gives $\int_{\mathcal{S}} \mathbf{D}_{\mathbf{u}}^T \mathbf{D}_{\mathbf{u}} d\mathbf{u} = 3\mathbf{E}_1^T \mathbf{E}_1 + 4\mathbf{E}_2^T \mathbf{E}_2 + 3\mathbf{E}_3^T \mathbf{E}_3 + \mathbf{E}_3^T \mathbf{E}_1 + \mathbf{E}_1^T \mathbf{E}_3$.

Similarly, we rewrite

$$\int_{\mathcal{S}} \mathbf{D}_{\mathbf{u}}^T \mathbf{z}_{\mathbf{u}} d\mathbf{u} = \mathbf{E}^T \underbrace{\int_{\mathcal{S}} \mathbf{S}(\mathbf{u})^T \mathbf{z}_{\mathbf{u}} d\mathbf{u}}_{\mathbf{q}} = \sum_{j=1}^P \mathbf{E}_j^T \mathbf{q}_j. \quad (19)$$

To compute \mathbf{q} we may use a numerical quadrature scheme to approximate the integral over \mathcal{S} , i.e. we approximate $\mathbf{q} \approx \sum_{i=1}^K w_i \mathbf{S}(\mathbf{u}_i) \mathbf{z}_{\mathbf{u}_i}$, where the samples $\mathbf{u}_i \in \mathcal{S}$ and weights w_i , $i = 1, \dots, K$, are determined by the choice of quadrature; more details on the choice and performance of specific quadratures are given below. Note that the above equation implies that we only need to store P images specified by the vector \mathbf{q} , which is much smaller than the number of samples K required to ensure a good approximation of the integral. This considerably reduces the memory demand of the algorithm.

In general, the linear equation (18) may be readily solved using a fast matrix solver such as the conjugate gradient algorithm. Note that the matrices \mathbf{E}_j are circulant matrices and hence are diagonalizable with the discrete Fourier transform (DFT). When the measurement operator \mathbf{A} is also diagonalizable with the DFT³, (18) can be solved efficiently in the DFT domain, as we now show in a few specific cases:

1) *Fourier Sampling*: Suppose \mathbf{A} samples some specified subset of the Fourier coefficients of an input image \mathbf{x} . If the Fourier samples are on a Cartesian grid, then we may write $\mathbf{A} = \mathbf{S}\mathcal{F}$, where \mathcal{F} is the d -dimensional discrete Fourier transform and $\mathbf{S} \in \mathbb{R}^{M \times N}$ is the sampling operator. Then (18) can be simplified by evaluating the discrete

³Or, more generally, the gram matrix $\mathbf{A}^* \mathbf{A}$ need be diagonalizable with the DFT.

Fourier transform of both sides. We obtain an analytical expression for \mathbf{x} as:

$$\mathbf{x} = \mathcal{F}^{-1} \left\{ \frac{2 \mathbf{S}^T \mathbf{b} + \lambda \beta \mathcal{F} [\mathbf{E}^T \mathbf{q}]}{2 \mathbf{S}^T \mathbf{S} + \lambda \beta \mathcal{F} [\mathbf{E}^T \mathbf{C} \mathbf{E}]} \right\}. \quad (20)$$

Here, $\mathcal{F} [\mathbf{E}^T \mathbf{C} \mathbf{E}]$ is the transfer function of the convolution operator $\mathbf{E}^T \mathbf{C} \mathbf{E}$. When the Fourier samples are not on the Cartesian grid (for example, in parallel imaging), where the one step solution is not applicable, we could still solve (18) using preconditioned conjugate gradient iterations.

2) *Deconvolution*: Suppose \mathbf{A} is given by a circular convolution, i.e. $\mathbf{A} \mathbf{x} = \mathbf{h} * \mathbf{x}$, then $\mathcal{F}[\mathbf{A} \mathbf{x}] = \mathbf{H} \cdot \mathcal{F}[\mathbf{x}]$ and $\mathcal{F}[\mathbf{A}^T \mathbf{y}] = \bar{\mathbf{H}} \cdot \mathcal{F}[\mathbf{y}]$, where $\mathbf{H} = \mathcal{F}[\mathbf{h}]$ and $\bar{\mathbf{H}}$ denotes the complex conjugate of \mathbf{H} . Taking the discrete Fourier transform on both sides, (18) can be solved as:

$$\mathbf{x} = \mathcal{F}^{-1} \left\{ \frac{2 \bar{\mathbf{H}} \cdot \mathcal{F}[\mathbf{b}] + \lambda \beta \mathcal{F} [\mathbf{E}^T \mathbf{q}]}{2 |\mathbf{H}|^2 + \lambda \beta \mathcal{F} [\mathbf{E}^T \mathbf{C} \mathbf{E}]} \right\}. \quad (21)$$

The denoising setting corresponds to the choice $\mathbf{A} = \mathbf{I}$, the identity operator, in which case $\mathbf{H} = \mathbf{1}$, the vector of all ones.

E. Discretization of the derivative operators

The standard approach to approximate the partial derivatives is using finite difference operators. For example, the derivative of a 2D signal along the x dimension is approximated as $q[k_1, k_2] = f[k_1 + 1, k_2] - f[k_1, k_2] = \Delta_1 * f$. This approximation can be viewed as the convolution of f by $\Delta_1[k] = \varphi(k + \frac{1}{2})$, where $\varphi(x) = \partial B_1(x)/\partial x$ and $B_1(x)$ is the first degree B-spline [18]. However, this approximation does not possess rotation steerability, i.e. the directional derivative can not be expressed as the linear combination of the finite differences along x and y directions.

To obtain discrete operators that are approximately rotation steerable, in the 2D case we approximate the n th order partial derivatives, $\partial^{n_1, n_2} := \partial_x^{n_1} \partial_y^{n_2}$ for all $n_1 + n_2 = n$, as the convolution of the signal with the tensor product of derivatives of one-dimensional B-spline functions:

$$\partial^{n_1, n_2} f[k_1, k_2] = \left[B_n^{(n_1)}(k_1 + \delta) \otimes B_n^{(n_2)}(k_2 + \delta) \right] * f[k_1, k_2], \quad (22)$$

for all $k_1, k_2 \in \mathbb{N}$, where $B_n^{(m)}(x)$ denotes the m th order derivative of a n th degree B-spline. In order to obtain filters with small spacial support, we choose the δ according to the rule

$$\delta = \begin{cases} \frac{1}{2} & \text{if } n \text{ is odd} \\ 0 & \text{else.} \end{cases}$$

The shift δ implies that we are evaluating the image derivatives at the intersection of the pixels and not at the pixel midpoints. This scheme will result in filters that are spatially supported in a $(n+1) \times (n+1)$ pixel window. Likewise, in the 3D case we approximate the n th order partial derivatives, $\partial^{n_1, n_2, n_3} := \partial_x^{n_1} \partial_y^{n_2} \partial_z^{n_3}$ for all $n_1 + n_2 + n_3 = n$, as

$$\partial^{n_1, n_2, n_3} f[k_1, k_2, k_3] = \left[B_n^{(n_1)}(k_1 + \delta) \otimes B_n^{(n_2)}(k_2 + \delta) \otimes B_n^{(n_3)}(k_3 + \delta) \right] * f[k_1, k_2, k_3], \quad (23)$$

for all $k_1, k_2, k_3 \in \mathbb{N}$ with the same rule for choosing δ , which results in filters supported in a $(n+1)^3$ volume.

While the tensor product of B-spline functions are not strictly rotation steerable, B-splines approximate Gaussian functions as their degree increases, and the tensor product of Gaussians is exactly steerable. Hence, the approximation of derivatives we define above is approximately rotation steerable; see Fig. 1. However, the support of the filters required for exact rotation steerability is much larger than the B-spline filters. These larger filters were observed to provide worse reconstruction performance than the B-spline filters. Thus the B-spline filters represent a compromise between filters that are exactly steerable but have large support, and filters that have small support but are poorly steerable.

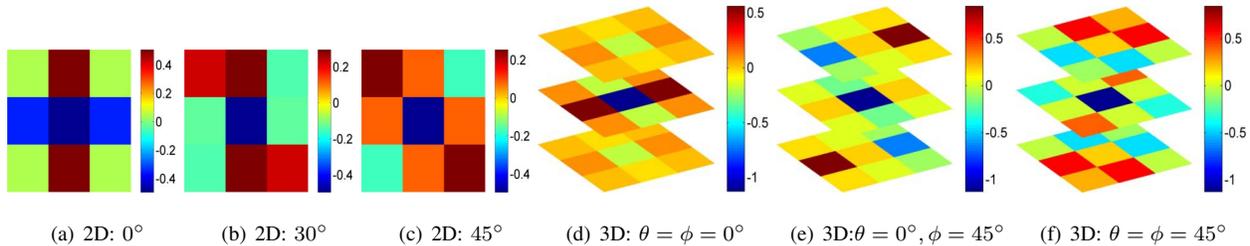


Fig. 1. 2D and 3D B-spline directional derivative operators at different angles. Note that the operators are approximately rotation steerable.

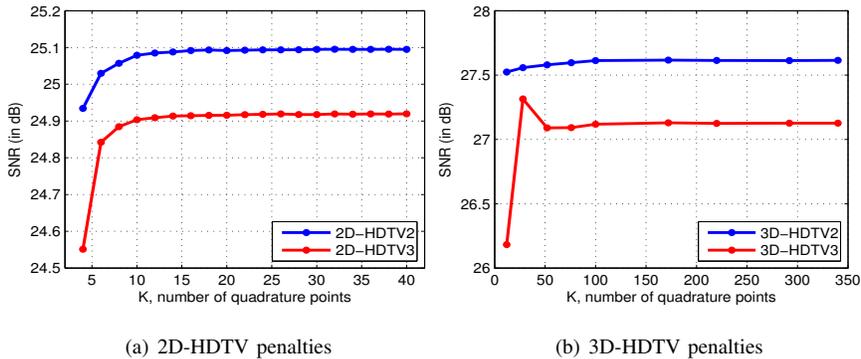


Fig. 2. Performance of proposed quadrature schemes. In (a) and (b) we display the SNR (as defined in (24)) in a denoising experiment as a function of the number of quadrature points K used to approximate the integral in (14). The same inputs were used for each K , except for the regularization parameter λ was tuned in each case to optimize the resulting SNR. In both the 2D and 3D case we observe an initial gain in SNR, demonstrating the value in better approximating the integral, but the change in SNR slows after a certain threshold. Namely, in the 2D experiment (a) we see that the change in SNR is within 0.01 dB after $K \approx 16$ for both the HDTV2 and HDTV3 penalties. Likewise, in the 3D experiment (b), the change in SNR is within 0.05 dB after $K \approx 76$.

F. Numerical quadrature schemes for $SO(d)$ and \mathbb{S}^{d-1}

Our algorithm requires us to compute the projected shrinkage \mathbf{q} in (19), which is defined as an integral over $\mathcal{S} = SO(d)$ or \mathbb{S}^{d-1} . We approximate this quantity using various numerical quadrature schemes depending on the space \mathcal{S} . In the 2D case, we may simply parameterize $\mathbf{u} \in \mathbb{S}^1$ as $\mathbf{u}_\theta = [\cos(\theta), \sin(\theta)]$, then approximate with a Riemann sum by discretizing the parameter θ as $\theta_i = i\frac{2\pi}{K}$, for $i = 1, \dots, K$, where K is the specified number

of sample points. In this case we find $K \geq 16$ samples yields a suitable approximation, in the sense that the optimal SNR of reconstructions obtained under various generalized HDTV penalties are unaffected by increasing the number of samples; see plot (a) in Fig. 2.

In higher dimensions the analogous Riemann sum approximations become inefficient, and instead we make use of more sophisticated numerical quadrature rules. To approximate integrals over the sphere \mathbb{S}^2 we apply Lebedev quadrature schemes [19]. These schemes exactly integrate all polynomials on the sphere up to a certain degree, while preserving certain rotational symmetries among the sample points. This is advantageous because the number of sample points can be significantly reduced if the derivative operators $\mathbf{D}_{\mathbf{u}}$ obey any of the same rotational symmetries. In general, we find that Lebedev schemes with $K \geq 76$ sample points provide a suitable approximation; see plot (b) in Fig. 2. Additionally, we note that for integrals over $SO(3)$ we may design efficient quadrature schemes by taking the product of a 1D uniform quadrature scheme and a Lebedev scheme. We refer the interested reader to [20] for more details.

G. Algorithm Overview

Algorithm : FAST HDTV($\mathbf{A}, \mathbf{b}, \lambda$)

$M \leftarrow 1$

$\beta \leftarrow \beta_{\text{init}}, \mathbf{x} \leftarrow \mathbf{A}^T \mathbf{b}$

while $M < \text{MaxOuterIterations}$

{	do	$m \leftarrow 1$
		while $m < \text{MaxInnerIterations}$
		Compute partial derivatives using (22) or (23)
		Compute rotated operator outputs $\mathbf{D}_{\mathbf{u}_i} \mathbf{x}$ using (10)
		Update $\mathbf{z}_{\mathbf{u}_i}$ based on $[\mathbf{D}_{\mathbf{u}_i} \mathbf{x}]_j$ using shrinkage rule (17)
		Compute projected shrinkage \mathbf{q} using (19)
		Update \mathbf{x} based on \mathbf{q} using (20), (21)
		$m \leftarrow m + 1$
		$\beta \leftarrow \beta * \beta_{\text{incfactor}}$
		$M \leftarrow M + 1$

return (\mathbf{x})

The pseudocode for the fast alternating HDTV algorithm is shown above. In our experiments we use the continuation scheme $\beta_{n+1} = \beta_{\text{inc}} \cdot \beta_n$ for some constant $\beta_{\text{inc}} > 1$ and initial β_0 . We warm start each new iteration with the estimate from the previous iteration, and stop when a given convergence tolerance has been reached; we evaluate the performance of the algorithm under different choices of β_0 and β_{inc} in the results section. We typically use 10 outer iterations ($\text{MaxOuterIterations} = 10$) and a maximum of 10 inner iterations ($\text{MaxInnerIterations} = 10$). The algorithm is terminated when the relative change in the cost function is less than a specified threshold.

V. RESULTS

We compare the performance of the proposed fast 2D-HDTV algorithm with our previous implementation based on IRMM. We also study the improvement offered by the proposed 2D- and 3D-HDTV schemes over classical TV methods in the context of applications such as deblurring and recovery of MRI data from under sampled measurements. We omit comparisons with other 2D-TV generalizations since extensive comparisons were performed in our previous paper [2]. In each case, we optimize the regularization parameters to obtain the optimized SNR to ensure fair comparisons between different schemes. The signal to noise ratio (SNR) of the reconstruction is computed as:

$$\text{SNR} = -10 \log_{10} \left(\frac{\|\mathbf{x}_{\text{orig}} - \hat{\mathbf{x}}\|_F^2}{\|\mathbf{x}_{\text{orig}}\|_F^2} \right), \quad (24)$$

where $\hat{\mathbf{x}}$ is the reconstructed image, \mathbf{x}_{orig} is the original image, and $\|\cdot\|_F$ is the Frobenius norm.

A. Two-dimensional HDTV using fast algorithm

1) *Convergence of the fast HDTV algorithm:* We investigate the effect of the continuation parameter β and the increment rate β_{inc} on the convergence and the accuracy of the algorithm. For this experiment, we consider the reconstruction of a MR brain image with acceleration factor 1.65 using the fast HDTV algorithm. The cost as a function of the number of iterations and the SNR as a function of the CPU time is plotted in Fig. 3. We observe that with different combinations of starting values of β and increment rate β_{inc} , the convergence rates of the algorithms are approximately the same and the SNRs of the reconstructed images are around the same value. However, when we choose the parameters as $\beta = 15$ and $\beta_{\text{inc}} = 2$, which are the smallest among the parameters chosen in the experiments, the SNR of the recovered image is comparatively lower than the others. This implies that in order to enforce full convergence the final value of β needs to be sufficiently large.

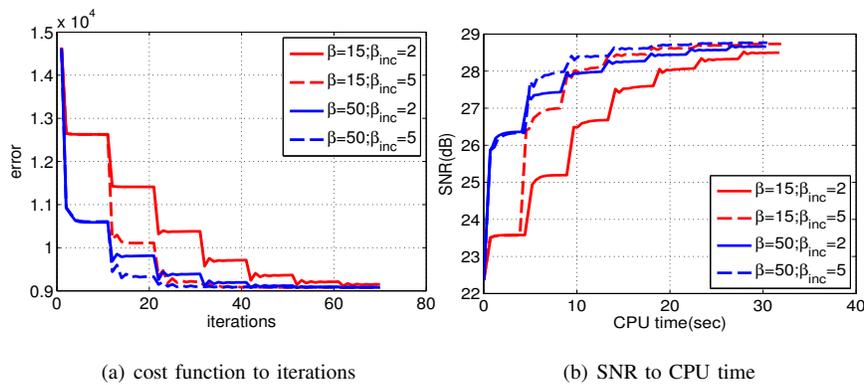


Fig. 3. Performance of the continuation scheme. We plot the cost as a function of the number of iterations in (a) and SNR as a function of CPU time in (b). We investigate four different combinations of the parameters β and β_{inc} . It is shown in (a) that the convergence rates of different combinations are approximately the same. We also observe in (b) that the SNR's of the reconstructed images in four settings are similar except that when the final value of β is not large enough ($\beta = 15$, $\beta_{\text{inc}} = 2$) the SNR is comparatively lower than the others.

2) *Comparison of the fast HDTV algorithm with iteratively reweighted HDTV algorithm:* In this experiment, we compare the proposed fast HDTV algorithm with the IRMM algorithm in the context of the recovery of a brain MR image with acceleration factor of 4 in Fig. 4. Here we plot the SNR as a function of the CPU time using TV and second degree HDTV with the IRMM algorithm and the proposed algorithm, respectively. We observe that the proposed algorithm (blue curve) takes around 20 seconds to converge compared to 120 seconds by IRMM algorithm (blue dotted curve) using TV penalty, and 30 seconds (red curve) compared to 300 seconds (red dotted curve) using second degree HDTV regularization. Thus, we see that the proposed algorithm accelerates the problem significantly (ten-fold) compared to IRMM method.

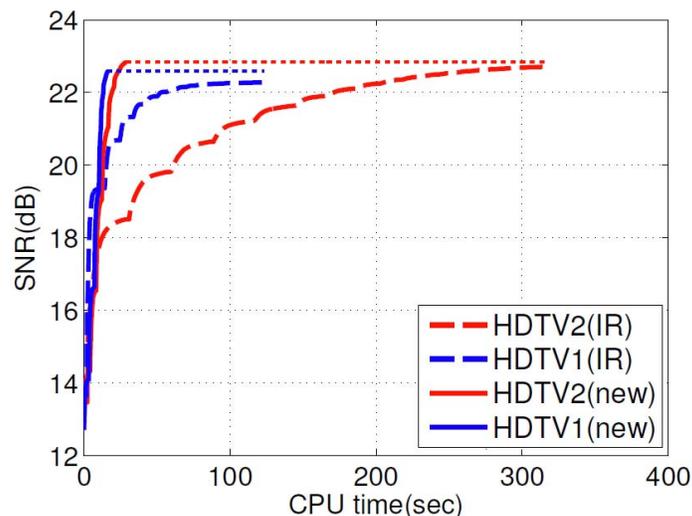


Fig. 4. IRMM algorithm versus proposed fast HDTV algorithm in different settings. The blue, blue dotted, red, red dotted curves correspond to HDTV1 using proposed algorithm, HDTV1 using IRMM, HDTV2 using proposed algorithm, HDTV2 using IRMM algorithm, respectively. We extend (solid lines) the original plot by dotted lines for easier comparisons of the final SNRs. We see that the proposed algorithm takes 1/6 of the time taken by IRMM for HDTV1, and 1/10 of the time taken by IRMM for HDTV2.

3) *Comparison of related algorithms:* In order to demonstrate the utility of HDTV, we compare its performance with standard TV and two state-of-the-art schemes using higher order image derivatives, i.e, 1) the Hessian-Shatten norm $p = 1$ regularization penalty [12], which we refer to as HS1 regularization; 2) the total generalized variation scheme [11], referred to as TGV regularization. In Fig. 5, we have compared second and third degree HDTV with TV, HS1, and TGV regularization in the context of deblurring of a microscopy cell image of size 450×450 . The original image is blurred with a 5×5 Gaussian filter with standard deviation 1.5, with additive Gaussian noise of standard deviation 0.05. We observe that the TV deblurred image has patchy artifacts and washes out the cell textures, while the higher degree schemes, including HDTV2, HDTV3, and HS1, gave very similar results (shown from (c) to (f)) with more accurate reconstructions, improving the SNR over standard TV by approximately 0.5 dB.

In Table I we present quantitative comparisons of the performance of the regularization schemes on six test images, in the context of compressed sensing, denoising, and deblurring. We observe that HDTV regularization

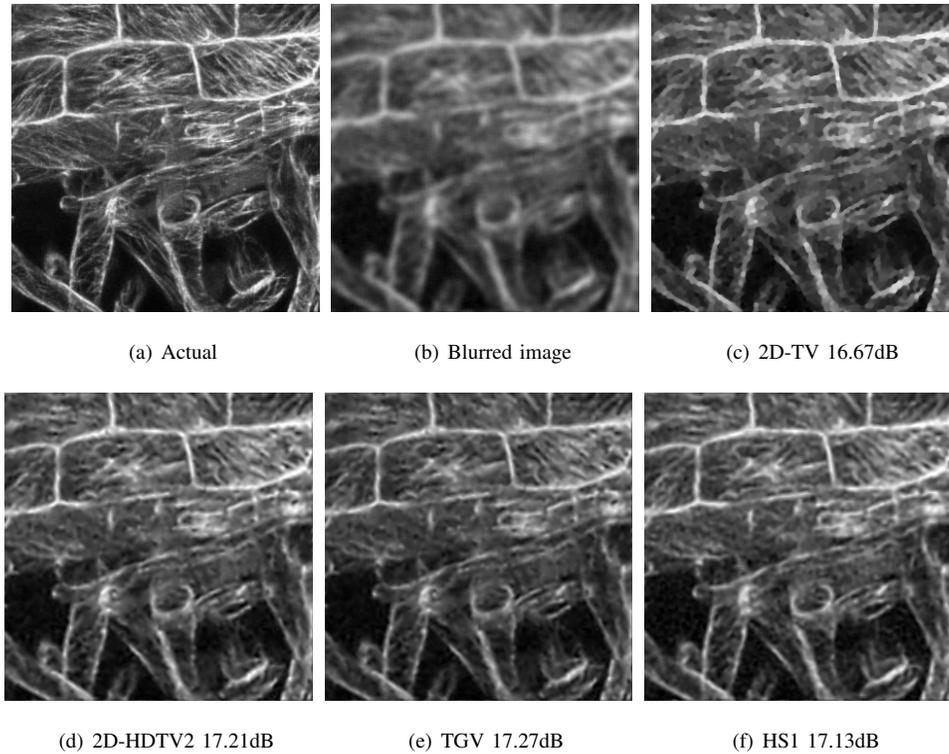


Fig. 5. Deblurring of a microscopy cell image. (a) is the actual cell image. (b) is the blurred image. (c)-(f) show the deblurred images using TV, HDTV2, TGV, and HS1 schemes, respectively. The results show that TV brings in patchy artifacts, while higher degree TV schemes preserve more details. HDTV2, TGV, and HS1 methods provide almost similar results both visually and in SNR, with a 0.5 dB SNR improvement over standard TV.

provides the best SNR among schemes that only rely on single degree derivatives. The comparison of the proposed methods against TGV, which is a hybrid method involving first and second degree derivatives, show that in some cases the TGV scheme provides slightly higher SNR than the HDTV methods. However, we did not observe significant perceptual differences between the images. All higher degree schemes routinely outperform standard TV.

		CS-brain	CS-wrist	Denoise-brain	Denoise-Lena	Deblur-cell1	Deblur-cell2
single-degree	2D-TV	22.77	20.96	27.60	27.35	15.66	16.67
	2D-HDTV2	22.82	21.20	28.05	27.65	16.19	17.21
	2D-HDTV3	22.53	21.02	28.30	27.45	16.17	17.20
	HS1	22.50	20.51	28.08	27.51	16.17	17.13
multiple-degree	TGV	22.80	21.25	28.17	27.78	16.25	17.27

TABLE I
COMPARISON OF 2D TV-RELATED METHODS (SNR)

We also note that in Proposition 1 we demonstrated a theoretical equivalence between HDTV2 and HS1 regularization penalties in a continuous setting. These experiments confirm that the discrete versions of these penalties perform similarly in image reconstruction tasks.

B. Three-dimensional HDTV

In the following experiments we investigate the utility of HDTV regularization in the context of compressed sensing, denoising, and deconvolution of 3D datasets. Specifically, we compare image reconstructions obtained using the second and third degree 3D-HDTV penalty versus the standard 3D-TV penalty.

1) *Compressed Sensing*: In these experiments we consider the compressed sensing recovery of a 3D MR angiography dataset from noisy and undersampled measurements. We experiment on a $512 \times 512 \times 76$ MR dataset obtained from [21], which we retroactively undersampled using a variable density random Fourier encoding with acceleration factor of 1.5. To these samples we also added 5 dB Gaussian noise with standard deviation 0.53. Shown in Fig. 6 are the maximum intensity projections (MIP) of the reconstructions obtained using various schemes. We observe that there is a 0.4 dB improvement in 3D-HDTV over standard 3D-TV, and we also note that 3D-HDTV preserves more line details compared with standard 3D-TV. In Fig. 7 we present zoomed details of the two marked regions in Fig. 6. We observe that 3D-HDTV provides more accurate and natural-looking reconstructions, while 3D-TV result has some patchy artifacts that blur the details in the image.

2) *Deconvolution*: We compare the deconvolution performance of the 3D-HDTV with 3D-TV. Fig. 8 shows the deconvolution results of a 3D fluorescence microscope dataset ($1024 \times 1024 \times 17$). The original image is blurred with a Gaussian filter of size $5 \times 5 \times 5$ with standard deviation of 1, with additive Gaussian noise of standard deviation 0.01. The results show that 3D-HDTV scheme is capable of recovering the fine image features of the cell image, resulting in a 0.3 dB improvement in SNR over 3D-TV.

The SNRs of the recovered images in the context of different applications are shown in Table II. We observe that the HDTV methods provide the best overall SNR for all of the cases, in which 3D-HDTV2 gives the best SNR for compressed sensing settings, and the 3D-HDTV3 method provides the best SNR for denoising and deblurring cases. Compared with 3D-TV scheme, the 3D-HDTV schemes improve the SNR of the reconstructed images by around 0.5 dB.

	CS-MRA(A=5)	CS-MRA(A=1.5)	CS-Cardiac	Denoise-1	Denoise-2	Deblur-1	Deblur-2	Deblur-3
3D-TV	13.87	14.53	18.37	17.12	16.25	19.02	16.43	14.50
3D-HDTV2	14.23	15.11	18.56	17.25	16.70	19.15	16.60	14.87
3D-HDTV3	14.01	14.70	18.50	17.68	17.14	19.73	17.43	15.23

TABLE II
COMPARISON OF 3D METHODS (SNR)

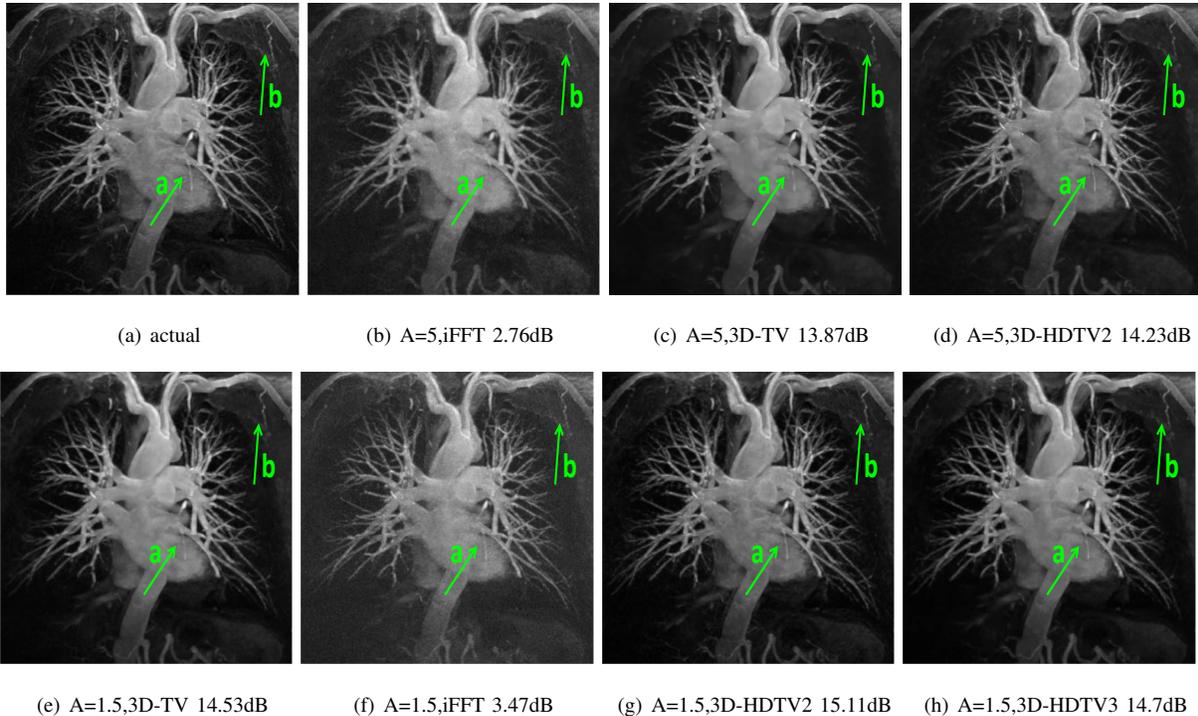


Fig. 6. Compressed sensing recovery of MR angiography data from noisy and undersampled Fourier data (acceleration of 1.5 with 5dB additive Gaussian noise). (a) through (h) are the maximum intensity projection images of the dataset. (a) is the original image. (b) to (d) show the reconstructions with acceleration of 5, using direct iFFT, 3D-TV, 3D-HDTV2, separately. (e) to (h) indicate the reconstructed images with acceleration of 1.5, using 3D-TV, direct iFFT, 3D-HDTV2, 3D-HDTV3, separately. We observe that 3D-HDTV method preserves more details that are lost in 3D-TV reconstruction. The arrows highlight two regions that are shown in zoomed detail in Fig. 7.

VI. DISCUSSION AND CONCLUSION

We introduce a family of novel image regularization penalties called generalized higher degree total variation (HDTV). These penalties are essentially the sum of ℓ_p , $p \geq 1$, norms of the convolution of the image by all rotations of a specific derivative operator. Many of the second degree TV extensions can be viewed as special cases of the proposed penalty or are closely related. We also introduce an alternating minimization algorithm that is considerably faster than our previous implementation for HDTV penalties; the extension of the proposed scheme to three dimensions is mainly enabled by this speedup. Our experiments demonstrate the improvement in image quality offered by the proposed scheme in a range of image processing applications.

This work could be further extended to account for other noise models. We assumed the quadratic data-consistency term in (1) for simplicity. Since it is the negative log-likelihood of the Gaussian distribution, this choice is only optimal for measurements that are corrupted by Gaussian noise. However, the proposed framework can be easily extended to other noise distributions by replacing the data fidelity term by the negative log-likelihood of the specified noise distribution. We do not consider these extensions in this work since our focus is only on modifying the regularization penalty.

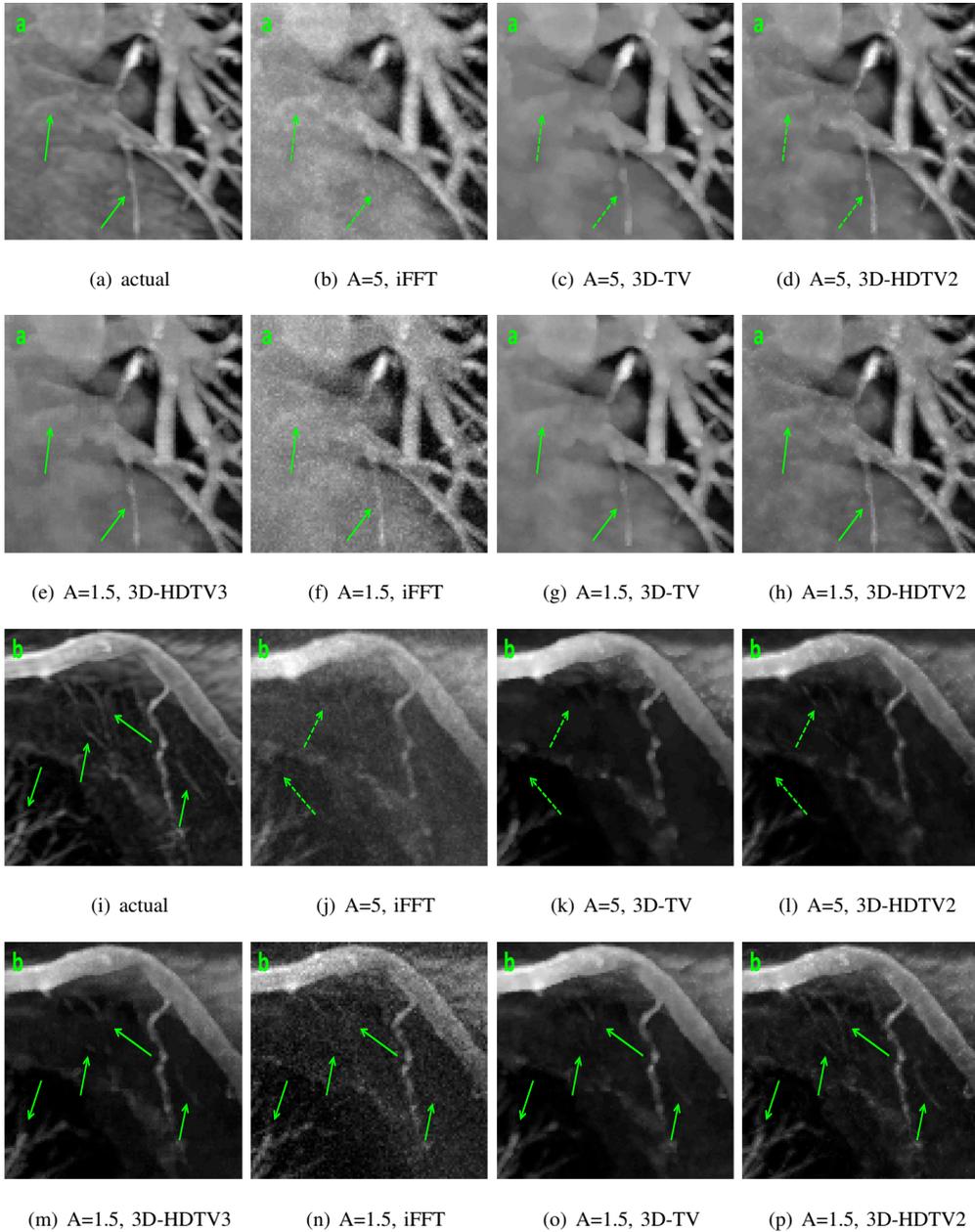


Fig. 7. The zoomed images of the two regions highlighted in Fig. 6. The first two rows display the zoomed region (a) of the reconstructions with acceleration of 1.5 and 5 using direct inverse Fourier transform, 3D-TV, 3D-HDTV, separately. The third and fourth rows display the zoomed region (b) of the reconstructions. We observe that 3D-HDTV methods preserve more line-like features compared with 3D-TV (indicated by green arrows).

Another direction for future research is to further improve on our algorithm. The proposed version is based on a half-quadratic minimization method, which requires the parameter β to go infinity to ensure the equivalence of the auxiliary variables \mathbf{z}_u to \mathbf{D}_u (see (16)). It is shown by several authors that high β values are often associated

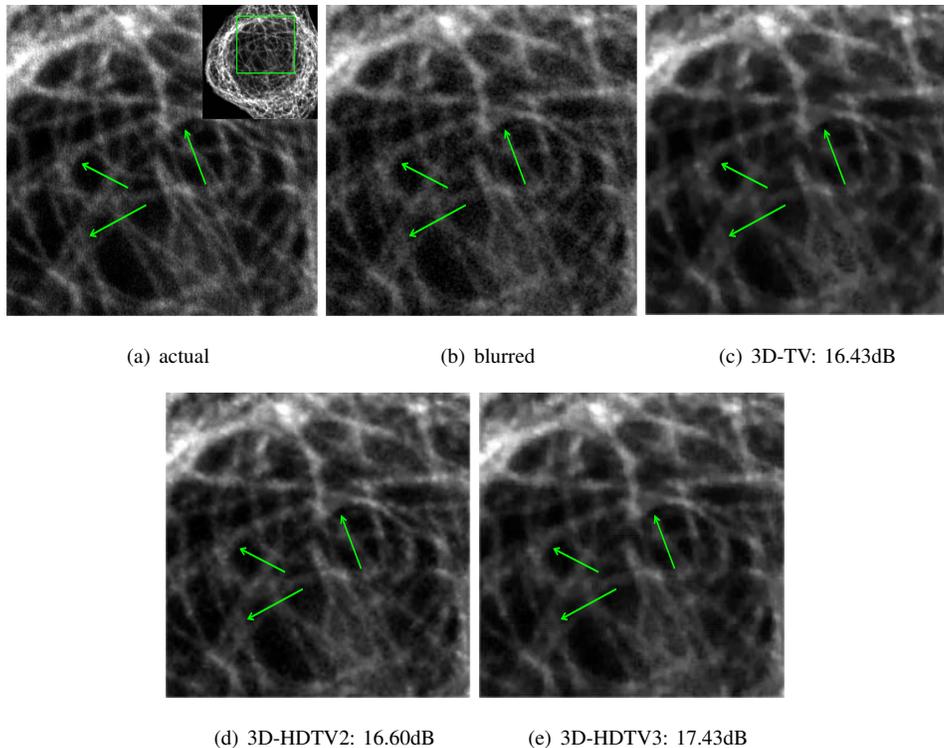


Fig. 8. Deconvolution of a 3D fluorescence microscope dataset. (a) is the original image. (b) is the blurred image using a Gaussian filter with standard deviation of 1 and size $5 \times 5 \times 5$ with additive Gaussian noise ($\sigma = 0.01$). (c) to (e) are deblurred images using 3D-TV, 3D-HDTV2, 3D-HDTV3, separately. We observe that the 3D-TV recovery is very patchy and some small details are lost, while the HDTV methods preserve the line-like features (indicated by green arrows) with a 0.2 dB improvement in SNR.

with slow convergence and stability issues; augmented Lagrangian (AL) methods or the split Bregman methods were introduced to avoid these problem in the context of total variation and ℓ_1 minimization schemes [22], [23]. These methods introduce additional Lagrange multiplier terms to enforce the equivalence, thus avoiding the need for large β values. However, these schemes are infeasible in our setting since we need as many Lagrange multiplier terms as the number of directions \mathbf{u}_i needed for accurate discretization; the associated memory demand is large. We implemented AL schemes in the 2D setting, but the improvement in the convergence was very small and did not justify the significant increase in memory demand. The challenge then is to design an algorithm for HDTV that exhibits the enhanced convergence properties of the AL method while maintaining a low memory footprint. This is something we hope to explore in a future work.

VII. APPENDIX

A. Proof of Proposition 1

The HDTV2 penalty can be expressed using the Hessian matrix as

$$\text{HDTV2}(f) = \int_{\Omega} \Phi(\mathcal{H}f(\mathbf{r})) d\mathbf{r},$$

where $\Phi(\mathcal{H}f(\mathbf{r})) := \int_{\mathbb{S}^{d-1}} |\mathbf{u}^T \mathcal{H}f(\mathbf{r}) \mathbf{u}| d\mathbf{u}$. Since the Hessian is a real symmetric matrix, it has the eigenvalue decomposition $\mathcal{H}f(\mathbf{r}) = \mathbf{U} \text{diag}(\lambda_f(\mathbf{r})) \mathbf{U}^T$ where $\mathbf{U} = \mathbf{U}(\mathbf{r})$ is a unitary matrix, and $\lambda_f(\mathbf{r})$ is a vector of the Hessian eigenvalues. Thus, by a change of variables

$$\Phi(\mathcal{H}f(\mathbf{r})) = \int_{\mathbb{S}^{d-1}} |\mathbf{u}^T \text{diag}(\lambda_f(\mathbf{r})) \mathbf{u}| d\mathbf{u}. \quad (25)$$

Because the singular values of a symmetric matrix are the absolute value of its eigenvalues, we have $\|\mathcal{H}f(\mathbf{r})\|_{S_1} = \|\lambda_f(\mathbf{r})\|_1$, and together with (25) this gives the factorization

$$\Phi(\mathcal{H}f(\mathbf{r})) = \underbrace{\int_{\mathbb{S}^{d-1}} \frac{|\mathbf{u}^T \text{diag}(\lambda_f(\mathbf{r})) \mathbf{u}| d\mathbf{u}}{\|\lambda_f(\mathbf{r})\|_1}}_{\Phi_0(\lambda_f(\mathbf{r}))} \|\mathcal{H}f(\mathbf{r})\|_{S_1}. \quad (26)$$

To prove the claim it suffices to establish bounds for Φ_0 . By the triangle inequality we have

$$\int_{\mathbb{S}^{d-1}} |\mathbf{u}^T \text{diag}(\lambda_f(\mathbf{r})) \mathbf{u}| d\mathbf{u} \leq \int_{\mathbb{S}^{d-1}} \mathbf{u}^T \text{diag}(|\lambda_f(\mathbf{r})|) \mathbf{u} d\mathbf{u} = V_d \|\lambda_f(\mathbf{r})\|_1.$$

If we further suppose the Hessian eigenvalues $\lambda_f(\mathbf{r}) = (\lambda_1(\mathbf{r}), \dots, \lambda_d(\mathbf{r}))$ are all non-negative, then $\mathbf{u}^* \text{diag}(\lambda_f(\mathbf{r})) \mathbf{u} \geq 0$ for all $\mathbf{u} \in \mathbb{S}^{d-1}$, and so we may remove the absolute value from the integrand in Φ_0 . Consider the vector-valued function $\mathbf{F}(\mathbf{v}) = \text{diag}(\lambda_f(\mathbf{r})) \mathbf{v}$ defined on the unit ball $B^d = \{\mathbf{v} \in \mathbb{R}^d : |\mathbf{v}| \leq 1\}$. Note that for $\mathbf{u} \in \mathbb{S}^{d-1}$, the surface normal $\mathbf{n}(\mathbf{u}) = \mathbf{u}$, hence we have

$$\mathbf{F}(\mathbf{u}) \cdot \mathbf{n}(\mathbf{u}) = \mathbf{u}^T \text{diag}(\lambda_f(\mathbf{r})) \mathbf{u}, \quad \forall \mathbf{u} \in \mathbb{S}^{d-1},$$

and so by the divergence theorem

$$\int_{\mathbb{S}^{d-1}} \mathbf{u}^T \text{diag}(\lambda_f(\mathbf{r})) \mathbf{u} d\mathbf{u} = \int_{\mathbb{S}^{d-1}} \mathbf{F} \cdot \mathbf{n} d\mathbf{u} = \int_{B^d} \nabla \cdot \mathbf{F} d\mathbf{v} = \int_{B^d} \sum_{k=1}^d \lambda_k(\mathbf{r}) d\mathbf{v} = V_d \|\lambda_f(\mathbf{r})\|_1,$$

where V_d is the volume of B^d . A similar argument holds when the Hessian eigenvalues $\lambda_f(\mathbf{r})$ are all non-positive.

Thus, for Φ_0 (re-scaled by V_d) we have

$$\Phi_0(\lambda_f(\mathbf{r})) = \frac{\int_{\mathbb{S}^{d-1}} |\mathbf{u}^T \text{diag}(\lambda_f(\mathbf{r})) \mathbf{u}| d\mathbf{u}}{V_d \|\lambda_f(\mathbf{r})\|_1} \leq 1,$$

where equality holds when the Hessian eigenvalues $\lambda_f(\mathbf{r})$ are all non-negative or non-positive. We now derive lower bounds for Φ_0 in the 2D and 3D cases.

1) *2D Bound:* Fix \mathbf{r} , and let $\lambda_f(\mathbf{r}) = (\lambda_1, \lambda_2)$. In 2D we have

$$\Phi_0(\lambda_1, \lambda_2) = \frac{\int_{\mathbb{S}^1} |\lambda_1 x^2 + \lambda_2 y^2| d\mathbf{u}}{\pi(|\lambda_1| + |\lambda_2|)} = \frac{\int_0^{2\pi} |\lambda_1 \cos^2(\theta) + \lambda_2 \sin^2(\theta)| d\theta}{\pi(|\lambda_1| + |\lambda_2|)}.$$

We have $\Phi_0(\lambda_1, \lambda_2) < 1$ only when λ_1 and λ_2 are both non-zero and differ in sign. By a scaling argument, it suffices to look at the case where $\lambda_1 = 1$ and $\lambda_2 = -\alpha$, for some α with $0 < \alpha \leq 1$. Thus, the minimum of Φ_0 coincides with the function

$$\Psi(\alpha) = \frac{\int_0^{2\pi} |\cos^2(\theta) - \alpha \sin^2(\theta)| d\theta}{\pi(1 + \alpha)}, \quad 0 < \alpha \leq 1.$$

By the identity $\cos^2(\theta) - \alpha \sin^2(\theta) = (\frac{1-\alpha}{2}) + (\frac{1+\alpha}{2}) \cos(2\theta)$ we have $\Psi(\alpha) = \frac{1}{2\pi} \int_0^{2\pi} \left| \frac{1-\alpha}{1+\alpha} + \cos(2\theta) \right| d\theta$. Setting $\Psi'(\alpha) = 0$ gives the necessary condition $\int_0^{2\pi} \text{sgn} \left[\frac{1-\alpha}{1+\alpha} + \cos(2\theta) \right] d\theta = 0$, which is true only if $\alpha = 1$. Therefore, we obtain the bound $\Phi_0(\lambda_1, \lambda_2) \geq \frac{1}{2\pi} \int_0^{2\pi} |\cos(2\theta)| d\theta = \frac{2}{\pi} \approx 0.63$.

B. 3D Bound

In the 3D case we have

$$\Phi_0(\lambda_1, \lambda_2, \lambda_3) = \frac{3}{4\pi} \int_{\mathbb{S}^2} \frac{|\lambda_1 x^2 + \lambda_2 y^2 + \lambda_3 z^2| d\mathbf{u}}{|\lambda_1| + |\lambda_2| + |\lambda_3|} = \frac{3}{4\pi} \int_0^{2\pi} \int_0^\pi \frac{|\lambda_1 x^2 + \lambda_2 y^2 + \lambda_3 z^2|}{|\lambda_1| + |\lambda_2| + |\lambda_3|} \sin \phi d\phi d\theta,$$

where we use spherical coordinates $(x, y, z) = (\cos \theta \sin \phi, \sin \theta \sin \phi, \cos \phi)$, with $0 \leq \theta \leq 2\pi$, $0 \leq \phi \leq \pi$.

Note that $\Phi_0(\lambda_1, \lambda_2, \lambda_3) < 1$ only if for some $i \neq j$, λ_i and λ_j are both non-zero and differ in sign. By a scaling argument, it suffices to look at the case where $\lambda_1 = -\alpha$, $\lambda_2 = -\beta$, and $\lambda_3 = 2$ for some α and β with $0 \leq \alpha, \beta \leq 2$.

Thus, it is equivalent to minimize the function

$$\Psi(\alpha, \beta) = \frac{3}{4\pi} \int_0^{2\pi} \int_0^\pi \frac{|2z^2 - \alpha x^2 - \beta y^2|}{2 + \alpha + \beta} \sin \phi d\phi d\theta, \quad 0 \leq \alpha, \beta \leq 2.$$

Using the identity $x^2 + y^2 + z^2 = 1$, one can show that

$$\begin{aligned} 2z^2 - \alpha x^2 - \beta y^2 &= \left(\frac{\beta - \alpha}{2}\right) (x^2 - y^2) + \left(\frac{4 + \alpha + \beta}{6}\right) (3z^2 - 1) + \frac{2 - \alpha - \beta}{3} \\ &= \frac{a}{2} \sin^2(\phi) \cos(2\theta) + \left(\frac{4 + b}{6}\right) (3 \cos^2(\phi) - 1) + \frac{2 - b}{3}, \end{aligned}$$

where we have set $a = \beta - \alpha$ and $b = \alpha + \beta$. We may write this as $A \cos(2\theta) + B$, where $A = A(a, \phi) = \frac{a}{2} \sin^2(\phi)$ and $B = B(b, \phi)$ is given by the above equation. Then the minimum of Ψ coincides with the minimum of the function

$$\tilde{\Psi}(a, b) = \frac{3}{4\pi(2+b)} \int_0^\pi \int_0^{2\pi} |A \cos(2\theta) + B| d\theta d\phi, \quad -2 \leq a \leq 2; \quad 0 \leq b \leq 4.$$

Observe that

$$\int_0^{2\pi} |A \cos(2\theta) + B| d\theta \geq \int_0^{2\pi} |B| d\theta,$$

which implies $\tilde{\Psi}(a, b) \geq \tilde{\Psi}(0, b)$, and so a necessary condition for a minimum is $a = 0$, or equivalently $\alpha = \beta$.

Thus, we evaluate

$$\begin{aligned} \Psi(\alpha, \alpha) &= \frac{3}{8\pi(1+\alpha)} \int_{\mathbb{S}^2} |2z^2 - \alpha(x^2 + y^2)| d\mathbf{u} \\ &= \frac{3}{4(1+\alpha)} \int_0^\pi |2 \cos^2 \phi - \alpha \sin^2 \phi| \sin \phi d\phi \\ &= \frac{1}{1+\alpha} \left(2\alpha \sqrt{\frac{\alpha}{2+\alpha}} - \alpha + 1 \right), \end{aligned}$$

which can be shown to have a minimum of ≈ 0.57 at $\alpha \approx 1.1$.

REFERENCES

- [1] L. Rudin, S. Osher, and E. Fatemi, "Nonlinear total variation based noise removal algorithms," *Physica D: Nonlinear Phenomena*, vol. 60, no. 1-4, pp. 259–268, Jan 1992.
- [2] Y. Hu and M. Jacob, "Higher degree total variation (HDTV) regularization for image recovery," *IEEE Transactions on Image Processing*, vol. 21, no. 5, pp. 2559–2571, May 2012.
- [3] Y. Wang, J. Yang, W. Yin, and Y. Zhang, "A new alternating minimization algorithm for total variation image reconstruction," *SIAM J. Imaging Sciences*, vol. 1, no. 3, pp. 248–272, 2008.

- [4] C. Li, W. Yin, H. Jiang, and Y. Zhang, “An efficient augmented lagrangian method with applications to total variation minimization,” Rice University, CAAM Technical Report 12-14, 2012.
- [5] C. Wu and X.-C. Tai, “Augmented Lagrangian method, dual methods and split Bregman iteration for ROF, vectorial TV, and higher order models,” *SIAM J. Imaging Sciences*, vol. 3, no. 3, pp. 300–339, 2010.
- [6] T. Chan, A. Marquina, and P. Mulet, “Higher-order total variation-based image restoration,” *SIAM J. Sci. Computing*, vol. 22, no. 2, pp. 503–516, Jul 2000.
- [7] G. Steidl, S. Didas, and J. Neumann, “Splines in higher order TV regularization,” *International Journal of Computer Vision*, vol. 70, no. 3, pp. 241–255, 2006.
- [8] Y. You and M. Kaveh, “Fourth-order partial differential equations for noise removal,” *IEEE Transactions on Image Processing*, vol. 9, no. 10, pp. 1723 – 1730, Oct 2000.
- [9] M. Lysaker, A. Lundervold, and X.-C. Tai, “Noise removal using fourth-order partial differential equation with applications to medical magnetic resonance images in space and time,” *IEEE Transactions on Image Processing*, vol. 12, no. 12, pp. 1579–1590, Dec 2003.
- [10] S. Lefkimiatis, A. Bourquard, and M. Unser, “Hessian-based norm regularization for image restoration with biomedical applications,” *IEEE Transactions on Image Processing*, vol. 21, no. 3, pp. 983–995, March 2012.
- [11] K. Bredies, K. Kunisch, and T. Pock, “Total generalized variation,” *SIAM J. Imaging Sciences*, vol. 3, no. 3, pp. 492–526, 2010.
- [12] S. Lefkimiatis, J. Ward, and M. Unser, “Hessian Schatten-norm regularization for linear inverse problems,” *Image Processing, IEEE Transactions on*, vol. 22, no. 5, pp. 1873–1888, 2013.
- [13] S. Lefkimiatis, A. Bourquard, and M. Unser, “Hessian-based regularization for 3-d microscopy image restoration,” in *Biomedical Imaging (ISBI), 2012 9th IEEE International Symposium on*. IEEE, 2012, pp. 1731–1734.
- [14] J. Kybic, T. Blu, and M. Unser, “Generalized sampling: a variational approach, part I,” *IEEE Trans. Signal Processing*, vol. 50, pp. 1965–1976, 2002.
- [15] K. Kanatani, *Group Theoretical Methods in Image Understanding*. Springer-Verlag New York, Inc., 1990.
- [16] D. Geman and C. Yang, “Nonlinear image recovery with half-quadratic regularization,” *Image Processing, IEEE Transactions on*, vol. 4, no. 7, pp. 932–946, 1995.
- [17] M. Nikolova and M. K. Ng, “Analysis of half-quadratic minimization methods for signal and image recovery,” *SIAM Journal on Scientific computing*, vol. 27, no. 3, pp. 937–966, 2005.
- [18] M. Unser and T. Blu, “Wavelet theory demystified,” *Signal Processing, IEEE Transactions on*, vol. 51, no. 2, pp. 470–483, 2003.
- [19] V. I. Lebedev and D. Laikov, “A quadrature formula for the sphere of the 131st algebraic order of accuracy,” in *Doklady. Mathematics*, vol. 59, no. 3. MAIK Nauka/Interperiodica, 1999, pp. 477–481.
- [20] M. Gräf and D. Potts, “Sampling sets and quadrature formulae on the rotation group,” *Numerical Functional Analysis and Optimization*, vol. 30, no. 7-8, pp. 665–688, 2009.
- [21] “<http://physionet.org/physiobank/database/images/>.”
- [22] C. Wu, X.-C. Tai *et al.*, “Augmented lagrangian method, dual methods, and split bregman iteration for rof, vectorial tv, and high order models,” *SIAM J. Imaging Sciences*, vol. 3, no. 3, pp. 300–339, 2010.
- [23] T. Goldstein and S. Osher, “The split bregman method for l1-regularized problems,” *SIAM Journal on Imaging Sciences*, vol. 2, no. 2, pp. 323–343, 2009.