IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received March 16, 2022, accepted May 2, 2022, date of publication May 23, 2022, date of current version June 6, 2022.

Digital Object Identifier 10.1109/ACCESS.2022.3177277

Generalized Hessian-Schatten Norm Regularization for Image Reconstruction

MANU GHULYANI, DEEPAK G. SKARIAH, AND MUTHUVEL ARIGOVINDAN
Department of Electrical Engineering, Indian Institute of Science, Bengaluru, Karnataka 560012, India

Corresponding author: Muthuvel Arigovindan (mvel@iisc.ac.in)

This work was supported in part by the Prime Minister's Research Fellowship, Government of India; and in part by the Core Research Grant of Science and Engineering Research Board, Government of India, under Grant CRG/2019/004896.

Manu Ghulyani and Deepak G. Skariah contributed equally to this work.

ABSTRACT Regularization plays a crucial role in reliably utilizing imaging systems for scientific and medical investigations. It helps to stabilize the process of computationally undoing any degradation caused by physical limitations of the imaging process. In the past decades, total variation regularization played a dominant role in the literature. Two forms of total variation regularizations, namely the first-order and the second-order total variation (TV-1 and TV-2) have been widely used. TV-1 has a disadvantage: it reconstructs images in the form of piece-wise constants when the noise and/or under-sampling is severe, while TV-2 reconstructs natural-looking images under such scenarios. On the other hand, TV-1 can recover sharp jumps better than TV-2. Two forms of generalizations, namely Hessian-Schatten norm (HSN) regularization, and total generalized variation (TGV) regularization, have been proposed and have become significant developments in the area of regularization for imaging inverse problems owing to their performance. While the strength of TGV is that it can combine the advantages of TV-1 and TV-2, HSN has better structure-preserving property. Here, we develop a novel regularization for image recovery that combines the strengths of TGV and HSN. We achieve this by restricting the maximization space in the dual form of HSN in the same way that TGV is obtained from TV-2. We call the new regularization the generalized Hessian-Schatten norm regularization (GHSN). We develop a novel computational method for image reconstruction using the new form of regularization based on the well-known framework called the alternating direction method of multipliers (ADMM). We demonstrate the strength of the GHSN using some reconstruction examples.

INDEX TERMS Regularization, total variation, total generalized variation, Hessian-Schatten norm, inverse problems, MRI reconstruction.

I. INTRODUCTION

Images acquired using different imaging devices are inevitably corrupted due to the physical limitations of the image formation model. An estimation scheme that employs knowledge of the image formation forward model to generate a better quality estimate is known as image restoration/reconstruction [1]. The relative improvement in the image quality obtained by image restoration/reconstruction is typically significant in most modalities in general, and in particular, in modalities such as MRI imaging [2], [3], computed tomography [4], [5], confocal microscopy [6] and widefield microscopy [7]. One of the classical approaches to image restoration is the regularized approach [8]. It formulates the required reconstruction as the solution to the optimization problem as given below:

$$g_{opt} = \underset{g}{\operatorname{argmin}} \; J(g) = \big[G(m, g) + \lambda R(g)\big], \tag{1}$$

where $G(m, g)$ is the data fitting functional that measures the goodness of fit of the candidate image g to the measured image m, and $R(g)$ is the regularization functional that measures some kind of roughness of the image. The structure of the data-fitting term is dependent on the forward model of the imaging device and the assumed statistical model of the noise. The regularization functional [9] represents the prior information we have about the class of images we are trying to restore. The constant term λ is a user parameter that allows a tradeoff between the regularization term and the data fitting term.

The associate editor coordinating the review of this manuscript and approving it for publication was Hengyong Yu.

The recent leap in the computing power of desktop computers led to the application of deep neural networks (DNN) for image restoration [10]–[16]. The main argument that supports the use of DNN is that, while regularization methods impose *ad hoc* prior beliefs on the image to be restored, a trained DNN encompasses a more natural knowledge resulting from training data. However, a recent work by Hansen *et al.* [17] shows that such learned maps can be unstable. Further, the need for a large amount of training data limits the applicability of DNN based methods. In this paper, we limit our focus to regularization based methods.

Design of regularization functional $R(g)$ is a long-studied problem in signal and image processing [1]. One of the classic regularization techniques is Tikhonov regularization [18]. Tikhonov regularization was originally employed for solving integral equations, but it proved to be successful in imaging inverse problems as well [19]. The earliest form of Tikhonov regularization was constructed using sum-of-squares of the candidate image/signal [18], and then the later forms were constructed using the pixel-wise sum of squares of l_2 norm of gradients. When the image formation model is represented by blurring, reconstruction using Tikhonov regularization can be obtained in a single step using a simple formula expressed in the Fourier domain, which makes Tikhonov regularization attractive. However, Tikhonov regularization leads to over-smooth solutions. Nevertheless, Tikhonov regularization is still widely used, especially in large scale problems such as 3D deconvolution [20].

Later non-quadratic forms of regularization came into the picture owing to their better ability to preserve the resolution in the reconstructed images. Earliest forms of non-quadratic regularization were based on l_1 norm of the wavelet transform of the candidate image. Representative methods in this category include ISTA [21]–[23], FISTA [24] and TwIST [25]. These methods are primarily based on the fact that wavelet transform of typical images have low l_1 norm. The basis used in the wavelet transform can be adapted to data to give better performance in image restoration problems [26]. It should be emphasized that the factor that makes wavelet regularization better than the Tikhonov is that the wavelet transform has inbuilt derivatives and hence wavelet regularization in some sense becomes equivalent to l_1 minimization image derivatives. This leads to better preservation of structures than Tikhonov regularization which smooths out edges.

It was then recognized that minimizing non-quadratic measures of image derivative directly (instead of using the wavelet transform) gave better results, and this led to the era of total variation (TV) regularization in solving imaging inverse problems [27]–[36]. TV regularization was constructed using the pixel-wise sum of l_2-norm of gradients [27] which was found to yield better reconstructions. TV can better retain edges [27] in the reconstruction as compared to Tikhonov regularization [31], and in particular, it can recover sharp jumps in the reconstruction even in the presence of large amount of noise and/or undersampling. At the same time, it has a disadvantage that, in the presence of a large amount of noise and/or undersampling, it approximates smooth intensity variations in terms of piece-wise constant segments, which is known as staircase effect [37]. Higher-order extensions of TV [37] have been proposed to avoid the staircase effect and they deliver better restoration. Second-order TV (TV-2) [38]–[40] was constructed as the pixel-wise sum of Frobenius norm of image Hessians. TV-2 recovers linear intensity variations in the presence large amount of noise and/or under-sampling; however, it loses the ability to reproduce sharp intensity jumps that TV-1 can recover. More recently TV-2 was combined with a modified form of TV-1 regularization involving local averaging of image gradients to obtain improved image restoration [41]. A similar form was also applied for image inpainting problem in [42]. Another form of regularization that combines TV-1 and TV-2 in a spatially adaptive manner was proposed in [43] for improved image restoration.

A generalization of TV-2, known as Hessian-Schatten norm (HSN) regularization is obtained by replacing the Frobenius norm of the Hessian with the Schatten norm of the Hessian. Schatten norm is the l_p norm of the vector of eigenvalues of the Hessian [44], and when $p = 2$, this regularization becomes identical to TV-2. HSN regularization is better than TV-2 in the sense that setting p to be less than 2 typically yields improved results. HSN regularization has been applied for fluorescence microscopy [45], [46] and CT [47] imaging modalities. Image restoration with Hessian Schatten-norm can be solved with wide variety of assumptions on the noise model, e.g. Poisson [46], Gaussian [44] and mixed Poisson-Gaussian [48].

A generalization of total variation to higher-order terms, named total generalized variation (TGV) has been proposed [49]. It is generalized in two ways: it is formulated for any general derivative order; for any given order, it is generalized in the way how the derivatives are penalized. The second form of generalization is obtained by expressing the standard total variation regularization in dual form as a maximization, and then by imposing spatial smoothing constraints in the maximization problem. The second aspect of generalization has a more significant impact from a practical point of view. In fact, the version of TGV regularization that has been applied for non-trivial inverse problems (problems other than denoising) obtained by restricting the maximum order to be two (TGV-2) [50]; compared to the most widely used TV-2 regularization, it differs only by the second aspect of generalization. TGV-2 allows combining the best of TV-1 and TV-2: it allows retrieving sharp jumps as TV-1 does; at the same time, it gives reconstructions free of staircase artefacts as TV-2 does.

In this paper, we develop a novel type of regularization by generalizing the Hessian-Schatten norm regularization in the same way that TGV-2 regularization generalizes the TV-2 regularization. The resulting form of regularization includes TV-2, TGV-2, and Hessian-Schatten norm regularization as special cases. We call this regularization the generalized Hessian-Schatten norm regularization (GHSN), and

we develop a novel optimization method for image reconstruction using the GHSN. Our main contribution can be summarized as follows:

- We generalize the Hessian-Schatten norm regularization in the same way the second-order TGV regularization [49] generalizes second-order TV regularization, which we call the generalized Hessian-Schatten norm regularization (GHSN) [38]–[40].
- We develop this generalization in a way that is easily accessible without requiring the reader to study tensor calculus as required by the development given in [49].
- We propose a novel variable splitting scheme involving novel proximal operators for minimization resulting cost function using ADMM framework [51], [52], and develop formulas for these proximal operators. The resulting reconstruction method gives results that are better than the method proposed in [53] even if GHSN is restricted to be identical to the form used in [53]. This confirms the numerical advantage of the proposed minimization method.

We demonstrate the effectiveness of GHSN regularization using numerical experiments involving reconstructions from sparse Fourier samples and sparse spatial samples.

Notations and Mathematical Preliminaries:

1) Vectors are represented by lower-case bold faced letters with the elements represented by the same letter with a subscripted index. For example, $\mathbf{v}$ denotes a vector and its ith element is denoted by v_i. For a vector $\mathbf{v}$, $\|\mathbf{v}\|_p$ denotes the l_p norm given by $\|\mathbf{v}\|_p = (\sum_{i=1}^{n} |v_i|^p)^{1/p}$. It can be shown that $\|\mathbf{v}\|_p$ with $p \to \infty$ converges to the component of $\mathbf{v}$ that has the largest magnitude, and hence we write $\|\mathbf{v}\|_\infty = \max_i\{|v_i| : i = 1, 2, \ldots, n\}$.
2) We will deal with vector images; vector images are discrete 2D arrays where each pixel location has a vector quantity. It is denoted by lower-case bold-faced letter with a bold-faced lower-case letter as an argument. For example, $\mathbf{v}(\mathbf{r})$ is a vector image with $\mathbf{r} = [x \;\; y]^t$ representing a 2D pixel location. Depending on the context, the symbol denoting the pixel location may be omitted.
3) For a vector image, $\mathbf{v}(\mathbf{r})$, $\|\mathbf{v}\|_{1,2}$ denotes $\|\mathbf{v}\|_{1,2} = \sum_{\mathbf{r}} \|\mathbf{v}(\mathbf{r})\|_2$. It is the sum of pixel-wise l_2 norms, where $\sum_{\mathbf{r}}$ denotes the sum across pixel indices. Throughout the paper, we do not specify the bounds of summation as it is always the same and ranges from the first to last pixels. The norm $\|\mathbf{v}\|_{1,2}$ is a composition of norms, and is called the mixed-norm.
4) Matrices are represented by upper case bold faced letters. For a matrix $\mathbf{M}$, $\|\mathbf{M}\|_F$ denotes the Frobenius norm, which equals to the square root of sum of squares of its elements; it can be written as $\|\mathbf{M}\|_F = \sqrt{Tr(\mathbf{M}^t\mathbf{M})}$, where $Tr(\cdot)$ denotes the summing of diagonal elements. Next $\|\mathbf{M}\|_{S(p)}$ denotes the Schatten p-norm of the matrix, which is the l_p norm of the vector of singular values of the matrix $\mathbf{M}$. For a symmetric matrix, it is also the same as the l_p norm of the eigenvalues. In other words, if $\mathcal{E}$ denotes the operator that returns the vector of eigenvalues of a matrix, then $\|\mathbf{M}\|_{S(p)} = \|\mathcal{E}(\mathbf{M})\|_p$.
5) We will also deal with matrix images; matrix images are discrete 2D arrays where each pixel location has a matrix quantity. It is denoted by upper-case bold-faced letter with a bold-faced lower-case letter as an argument. For example, $\mathbf{M}(\mathbf{r})$ is a matrix image with $\mathbf{r}$ representing a 2D pixel location. Depending on the context, the symbol denoting the pixel location may be omitted. For a vector image, $\mathbf{v}(\mathbf{r})$, its jth scalar image is given by $v_j(\mathbf{r})$. For a matrix image, $\mathbf{M}(\mathbf{r})$, its (i, j)th scalar image is given by $m_{i,j}(\mathbf{r})$.
6) For an matrix image, $\mathbf{M}(\mathbf{r})$, let $\|\mathbf{M}\|_{1,S(p)}$ denote the l_1 norm of the pixel-wise Schatten p-norms. In other words, we have $\|\mathbf{M}\|_{1,S(p)} = \sum_{\mathbf{r}} \|\mathbf{M}(\mathbf{r})\|_{S(p)}$. Further, let $\|\mathbf{M}\|_{\infty,S(p)}$ denote the l_∞ norm of the pixel-wise Schatten p-norms. In other words, we have $\|\mathbf{M}\|_{\infty,S(p)} = \max_{\mathbf{r}} \|\mathbf{M}(\mathbf{r})\|_{S(p)}$. Note that the norms $\|\mathbf{M}\|_{1,S(p)}$ and $\|\mathbf{M}\|_{\infty,S(p)}$ are mixed-norm.
7) For square matrices of same dimension, $\mathbf{M}$, $\mathbf{N}$, let $\langle \mathbf{N}, \mathbf{M} \rangle = Tr(\mathbf{N}^t\mathbf{M})$. For a square matrix $\mathbf{N}$ let $\overline{\mathbf{N}} = \frac{\mathbf{N}+\mathbf{N}^t}{2}$. Then the following holds for any pair of square matrices: $\langle \mathbf{M}, \overline{\mathbf{N}} \rangle = \langle \overline{\mathbf{M}}, \mathbf{N} \rangle$.
8) If $\mathbf{M}$, and $\mathbf{N}$ denote matrix images such that for each pixel index $\mathbf{r}$, $\mathbf{M}(\mathbf{r})$, and $\mathbf{N}(\mathbf{r})$ denotes square matrices of the same dimensions, then, $\langle \mathbf{N}(\mathbf{r}), \mathbf{M}(\mathbf{r}) \rangle$ denotes $Tr(\mathbf{N}^t(\mathbf{r})\mathbf{M}(\mathbf{r}))$. On the other hand, $\langle \mathbf{N}, \mathbf{M} \rangle$ denotes $\sum_{\mathbf{r}} Tr(\mathbf{N}^t(\mathbf{r})\mathbf{M}(\mathbf{r}))$.
9) For a matrix image $\mathbf{M}(\mathbf{r})$, the norm $\|\mathbf{M}\|_{1,S(p)}$ can be expressed as a maximization of inner product of the form $\langle \cdot, \cdot \rangle$ [44]. Specifically we can write
$$\|\mathbf{M}\|_{1,S(p)} = \max_{\|\mathbf{N}\|_{\infty,S(q)} \leq 1} \langle \mathbf{N}, \mathbf{M} \rangle, \tag{2}$$
where q is the real positive number satisfying $1/p + 1/q = 1$. The notation $\max_{\|\mathbf{N}\|_{\infty,S(q)} \leq 1}$ denote maximization within the set of matrix images with the value of mixed norm $\|\mathbf{N}\|_{\infty,S(q)}$ upper-bounded by 1.
10) In this paper, * denotes the 2-D convolution operation, i.e $(x * y)(\mathbf{r}') = \sum_{\mathbf{r}} x(\mathbf{r})y(\mathbf{r}' - \mathbf{r})$. We extend notion of convolution to matrix images by using the rules of matrix multiplication. To be more specific, let $\mathbf{X}$ be an image of $m \times l$ matrices and let $\mathbf{Y}$ be image of $l \times n$ matrices. Then, $(\mathbf{X} * \mathbf{Y})_{i,j}(\mathbf{r}) = \sum_{p=1}^{l} (x_{i,p} * y_{p,j})(\mathbf{r})$ for $i = 1, \ldots, m$. As an example, let both $\mathbf{X}$ and $\mathbf{Y}$ be 2×2 matrix images. Then the elements of $\mathbf{Z} = \mathbf{X} * \mathbf{Y}$, can be written as
$$\begin{aligned} z_{1,1}(\mathbf{r}) &= (x_{1,1} * y_{1,1})(\mathbf{r}) + (x_{1,2} * y_{2,1})(\mathbf{r}), \\ z_{1,2}(\mathbf{r}) &= (x_{1,1} * y_{1,2})(\mathbf{r}) + (x_{1,2} * y_{2,2})(\mathbf{r}), \\ z_{2,1}(\mathbf{r}) &= (x_{2,1} * y_{1,1})(\mathbf{r}) + (x_{2,2} * y_{2,1})(\mathbf{r}), \\ z_{2,2}(\mathbf{r}) &= (x_{2,1} * y_{1,2})(\mathbf{r}) + (x_{2,2} * y_{2,2})(\mathbf{r}). \end{aligned}$$
11) For a scalar image g and an image of $m \times l$ matrices, $\mathbf{M}$, $\mathbf{M} * g$ is defined as $(\mathbf{M} * g)_{i,j}(\mathbf{r}) = (m_{i,j} * g)(\mathbf{r})$ for $i = 1, \ldots, m, j = 1, \ldots, n$ and any location $\mathbf{r}$.

12) For an image of $m \times l$ matrices, $\mathbf{M}$, we can define a new matrix image $\mathbf{N} = \mathbf{M}^t$ by extending the idea of transpose of a matrix to the case of matrix images. The (i, j) entry of $\mathbf{N}$ can be defined as $n_{i,j}(\mathbf{r}) = m_{j,i}(\mathbf{r})$ for $i = 1, \ldots, m, j = 1, \ldots, n$ and any location $\mathbf{r}$.
13) For scalar images u and v, and a scalar filter h, the convolved inner product $\langle u, h * v\rangle$ satisfies the relation $\langle u, h * v\rangle = \langle \tilde{h} * u, v\rangle$, where $\tilde{h}$ denotes the flipped filter, i.e., $\tilde{h}$ satisfies $\tilde{h}(\mathbf{r}) = h(-\mathbf{r})$. This relation can be easily verified by writing the inner product in Fourier domain.
14) We will extend the notion of flipping for vector filter in a straight forward way. In other words, for a vector filter $\mathbf{h}$, $\tilde{\mathbf{h}}$ denotes the vector filter obtaining by flipping each of its constituent scalar filters. For a vector image $\mathbf{u}$ and a vector filter $\mathbf{h}$, and a scalar image of appropriate size, v, we have $\langle \mathbf{u}, \mathbf{h} * v\rangle = \langle \tilde{\mathbf{h}}^t * \mathbf{u}, v\rangle$. The operation $\tilde{\mathbf{h}}^t * (\cdot)$ is the adjoint of the operation $\mathbf{h} * (\cdot)$.
15) Let $\mathbf{N}$ be an $m \times m$ matrix image, and let $\mathbf{u}$ and $\mathbf{v}$ be $m \times 1$ vector filters. Then using the ideas of the previous point, we can show that $\langle \mathbf{u}, \mathbf{N} * \tilde{\mathbf{v}}\rangle = \langle \mathbf{u} * \mathbf{v}^t, \mathbf{N}\rangle$. By a trivial extension of this relation, we can also show that $\langle \mathbf{N}, \mathbf{u} * \mathbf{v}^t * g\rangle = \langle \mathbf{N} * \tilde{\mathbf{v}}, \mathbf{u} * g\rangle$ for any scalar image g.
16) Let $\mathbf{d}(\mathbf{r}) = [d_x(\mathbf{r})\ d_y(\mathbf{r})]^t$ where $d_x(\mathbf{r})$ and $d_y(\mathbf{r})$ denote discrete filters implementing the derivative operators $\frac{\partial}{\partial x}$ and $\frac{\partial}{\partial y}$. Then for a scalar image, g, $\mathbf{d} * g$ is the discrete gradient of g. Its adjoint, $\tilde{\mathbf{d}}^t * (\cdot)$, which is defined for image of 2×1 vectors, is called the discrete divergence. For a 2×1 vector image $\mathbf{v}$, we write $div\ \mathbf{v} = \tilde{\mathbf{d}}^t * \mathbf{v}$.
17) The extension of the notion of gradient for vector images is called the Jacobian. For a vector image, $\mathbf{v}(\mathbf{r})$, the Jacobian is given by $(\mathbf{v} * \mathbf{d}^t)(\mathbf{r})$. For a 2×1 vector image, $\mathbf{v}(\mathbf{r})$, and 2×2 matrix image, $\mathbf{U}(\mathbf{r})$, the inner product $\langle \mathbf{d} * \mathbf{v}^t, \mathbf{U}\rangle$ satisfies $\langle \mathbf{d} * \mathbf{v}^t, \mathbf{U}\rangle = \langle \mathbf{v}^t, \mathbf{u}\rangle$ where $\mathbf{u}$ is the row vector image obtained by applying the adjoint of the operation $\mathbf{d} * (\cdot)$ on $\mathbf{U}$. $\mathbf{u}$ is given by $div\ \mathbf{U} = [div\ \mathbf{u}_1\ div\ \mathbf{u}_2]$.

II. GENERALIZED HESSIAN-SCHATTEN NORM REGULARIZATION

Let $g(\mathbf{r})$ be the discrete candidate image, where $\mathbf{r} = [x\ y]^t$ is the discrete pixel index. Let $\mathbf{H}(\mathbf{r}) = \begin{bmatrix} d_{xx}(\mathbf{r}) & d_{xy}(\mathbf{r}) \\ d_{xy}(\mathbf{r}) & d_{yy}(\mathbf{r}) \end{bmatrix}$ where $d_{xx}(\mathbf{r})$, $d_{yy}(\mathbf{r})$, and $d_{xy}(\mathbf{r})$ are discrete filters implementing second order derivatives $\frac{\partial^2}{\partial x^2}$, $\frac{\partial^2}{\partial y^2}$ and $\frac{\partial^2}{\partial x \partial y}$ respectively. Note that $\mathbf{H}(\mathbf{r})$ is discrete Hessian operator. Convolution of this operator with $g(\mathbf{r})$ (denoted by $(\mathbf{H} * g)(\mathbf{r})$) is the discretized Hessian of the candidate image. We regard $\mathbf{M}(\mathbf{r}) = (\mathbf{H} * g)(\mathbf{r})$ as an image of 2×2 matrices; in other words, for each pixel index, $\mathbf{r}$, $\mathbf{M}(\mathbf{r})$ is a 2×2 matrix. Using this formulation, the well-known second-order total variation regularization can be expressed as $R_{TV2}(g) = \sum_{\mathbf{r}} \|(\mathbf{H} * g)(\mathbf{r})\|_F$. Similarily, Hessian-Schatten norm regularization [44] of order p applied on the candidate image g can be expressed as

$$\mathcal{HS}_p(g, \alpha_s) = \alpha_s \|\mathbf{H} * g\|_{1,S(p)} = \alpha_s \sum_{\mathbf{r}} \|(\mathbf{H} * g)(\mathbf{r})\|_{S(p)}. \tag{3}$$

To develop the novel regularization by extending Hessian-Schatten norm, we use the dual form of the Schatten norm. Specifically, we write Hessian-Schatten norm on g as

$$\mathcal{HS}_p(g, \alpha_s) = \max_{\|\mathbf{N}\|_{\infty,S(q)} \leq \alpha_s} \langle \mathbf{N}, \mathbf{H} * g\rangle, \tag{4}$$

where q is the real number satisfying $1/p + 1/q = 1$. Note that the above maximization is within the space of 2×2 matrix images. As shown by Lefkimmiatis *et al.* [46], the maximizer will be a symmetric matrix image, because the Hessian, $\mathbf{H} * g$, is symmetric. Hence, $\mathcal{HS}_p(g, \alpha_s)$ will also be equal to the result of maximization within the space of 2×2 symmetric matrix images. This is again equivalent to writing

$$\mathcal{HS}_p(g, \alpha_s) = \max_{\|\overline{\mathbf{N}}\|_{\infty,S(q)} \leq \alpha_s} \langle \overline{\mathbf{N}}, \mathbf{H} * g\rangle. \tag{5}$$

To formulate the generalization, we write the above expression in expanded form given below:

$$\mathcal{HS}_p(g, \alpha_s) = \max_{\|\overline{\mathbf{N}}\|_{\infty,S(q)} \leq \alpha_s} \sum_{\mathbf{r}} \langle \overline{\mathbf{N}}(\mathbf{r}), (\mathbf{H} * g)(\mathbf{r})\rangle. \tag{6}$$

Note that imposing an upper bound on maximum value on a vector/image is the same as applying the same upper bound on its components independently. Hence the above equation can be written as

$$\mathcal{HS}_p(g, \alpha_s) = \sum_{\mathbf{r}} \max_{\|\overline{\mathbf{N}}(\mathbf{r})\|_{S(q)} \leq \alpha_s} \langle \overline{\mathbf{N}}(\mathbf{r}), (\mathbf{H} * g)(\mathbf{r})\rangle. \tag{7}$$

From the above form, it is clear that the maximization is carried out for each $\mathbf{r}$ independently. Suppose $\mathcal{B}_s(q)$ denotes the set of 2×2 symmetric matrices such that, for each $\mathbf{A} \in \mathcal{B}_s(q)$ we have $\|\mathbf{A}\|_{S(q)} \leq 1$. Then the above minimization is carried out within a set that has a size that is L^2 times the size of $\mathcal{B}_s(q)$, where $L \times L$ is the size of $g(\mathbf{r})$. We propose to generalize $\mathcal{HS}_p(g, \alpha_s)$ by restricting the maximization space in the same way the second order TGV generalizes the second order total variation regularization (TV-2) [49]. To this end, let $\overline{\mathbf{N}}(\mathbf{r})$ be of the form $\overline{\mathbf{N}}(\mathbf{r}) = \begin{bmatrix} n_{xx}(\mathbf{r}) & n_{xy}(\mathbf{r}) \\ n_{xy}(\mathbf{r}) & n_{yy}(\mathbf{r}) \end{bmatrix}$, and let $div\ (\cdot)$ for a matrix image be as defined in the previous section. Note that $(div\ \overline{\mathbf{N}})(\mathbf{r})$ is a vector image. With this, we express the generalization as given below:

$$\mathcal{GHS}_p(g, \alpha_s, \alpha_f) = \max_{\substack{\|\overline{\mathbf{N}}\|_{\infty,S(q)} \leq \alpha_s \\ \|div\ \overline{\mathbf{N}}\|_{\infty,2} \leq \alpha_f}} \langle \overline{\mathbf{N}}, \mathbf{H} * g\rangle. \tag{8}$$

By using the fact that $(div\ \overline{\mathbf{N}}) = \tilde{\mathbf{d}}^t * \overline{\mathbf{N}}$ and transposition does not affect norm, we can write $\mathcal{GHS}_p(g, \alpha_s, \alpha_f)$ as given below:

$$\mathcal{GHS}_p(g, \alpha_s, \alpha_f) = \max_{\substack{\|\overline{\mathbf{N}}\|_{\infty,S(q)} \leq \alpha_s \\ \|\overline{\mathbf{N}} * \tilde{d}\|_{\infty,2} \leq \alpha_f}} \langle \overline{\mathbf{N}}, \mathbf{H} * g\rangle. \tag{9}$$

Note that $\mathcal{GHS}_p(g, \alpha_s, \alpha_f)$ is a generalization in the sense that $\mathcal{HS}_p(g, \alpha_s)$ is special case of $\mathcal{GHS}_p(g, \alpha_s, \alpha_f)$, i.e., $\mathcal{GHS}_p(g, \alpha_s, \infty) = \mathcal{HS}_p(g, \alpha_s)$. The above form is not usable as a regularization for image recovery as it involves symmetrization operation. Interestingly, the above form can be translated into a form that appears very close to the form TGV-2 given in [49]. The following proposition gives this expression.

Proposition 1: The generalized Hessian-Schatten norm $\mathcal{GHS}_p(g, \alpha_s, \alpha_f)$ can be expressed by

$$\mathcal{GHS}_p(g, \alpha_s, \alpha_f) = \min_{\mathbf{u}} \ \alpha_f \|\mathbf{d} * g - \mathbf{u}\|_{1,2} + \alpha_s \|0.5(\mathbf{u} * \mathbf{d}^t + \mathbf{d} * \mathbf{u}^t)\|_{1,S(p)}. \quad (10)$$

Note that, in terms of pixel-wise summations, the above form of regularization can written as

$$\mathcal{GHS}_p(g, \alpha_s, \alpha_f) = \min_{\mathbf{u}} \ \alpha_f \sum_{\mathbf{r}} \|(\mathbf{d} * g)(\mathbf{r}) - \mathbf{u}(\mathbf{r})\|_2 + \alpha_s \sum_{r} \|0.5((\mathbf{u} * \mathbf{d}^t)(\mathbf{r}) + (\mathbf{d} * \mathbf{u}^t)(\mathbf{r}))\|_{S(p)}. \quad (11)$$

It can be verified that this form resembles with TGV-2 given in [49], [50], except the fact that $\|\cdot\|_F$ is replaced by $\|\cdot\|_{S(p)}$. By noting the fact that $\|\cdot\|_{S(p)}$ with $p = 2$ becomes the same as $\|\cdot\|_F$, we conclude that the proposed regularization includes TGV-2 as a special case. Further note that $\mathcal{GHS}_p(g, \alpha_s, \alpha_f)$ becomes the Hessian-Schatten norm regularization as $\alpha_f \to \infty$. Moreover, it becomes the TV-2 regularization with $p = 2$ as $\alpha_f \to \infty$. Because of the minimization with respect to the auxiliary image $u(\mathbf{r})$, $\mathcal{GHS}_p(g, \alpha_s, \alpha_f)$ gains spatial adaptability. Near edges, $\mathbf{u}(\mathbf{r})$ can become zero leading to TV-1-like behaviour which allows sharp jumps in the edges. On the other hand, in smoother regions, $\mathbf{u}(\mathbf{r})$ can approach $\mathbf{d} * g(\mathbf{r})$ leading to the structure-preserving ability of Hessian-Schatten norm.

III. IMAGE RECONSTRUCTION USING GHSN REGULARIZATION

A. THE COST FUNCTION

In our development, we will use an image measurement model that is a composition of convolution followed by a spatial sampling operation. We do this as this model can be used to describe most the imaging modalities such as MRI, blurring, and sparse sampling for super-resolution. Let $h(\mathbf{r})$ denote the impulse response of the imaging system and suppose that the noise is Gaussian. We allow the possibility that $h(\mathbf{r})$ can be complex, and hence the measurement can have complex values. Let $\mathcal{T}_s$ denote the spatial sampling operator that samples an image from a predefined set of points and returns the vector of samples. With this, we consider the following form of data-fitting functional,

$$\bar{G}(g, h, \mathcal{T}_s, \quad \mathbf{m}) = \|\mathcal{T}_s(h * g) - \mathbf{m}\|_2^2, \quad (12)$$

where $\mathbf{m}$ denotes the vector of measurements. By introducing the notation $G(c, \mathcal{T}_s, \mathbf{m}) = \|\mathcal{T}_s(c) - \mathbf{m}\|_2^2$, we can write $\bar{G}(g, h, \mathcal{T}_s, \mathbf{m}) = G(h * g, \mathcal{T}_s, \mathbf{m})$. For the ease of implementation, we restrict the sample locations of $\mathcal{T}_s$ to be a subset of the regular cartesian grid. Specifically, let the elements of $\mathbf{m}$, denoted by $\{m_i, i = 1, \ldots, N\}$, be the samples from locations $\{\mathbf{r}_i, i = 1, \ldots, N\}$ that forms a subset of the regular cartesian grid. Then the above cost can be expressed as

$$G(h * g, \mathcal{T}_s, \mathbf{m}) = \sum_{i=1}^{N} ((h * g)(\mathbf{r}_i) - m_i)^2 \quad (13)$$

The above summation over N samples can be equivalently expressed as summation over all pixel of the $L \times L$ image as given below,

$$G(h * g, \mathcal{T}_s, \mathbf{m}) = \sum_{\mathbf{r}} W(\mathbf{r})((h * g)(\mathbf{r}) - m(\mathbf{r}))^2 \quad (14)$$

where $m(\mathbf{r})$ represents the image obtained by embedding the measured spatial samples $\{m_i, i = 1, \ldots, N\}$ in an array of zeros, and $W(\mathbf{r})$ a binary image in which ones represents the points at which measurements were taken. Note that this model is more general than the simple blurring model in the sense that, if the whole blurred image is measured, then $\mathcal{T}_s(c)$ will simply denote the operation of lexicographically scanning the pixels of the image c into a vector form. It is also equivalent to say that, in this case, $W(\mathbf{r})$ becomes an image of all ones. With this, GHS regularized image reconstruction can be expressed as

$$g_{opt} = \operatorname*{argmin}_{g} \ G(h * g, \mathcal{T}_s, \mathbf{m}) + \mathcal{GHS}_p(g, \alpha_s, \alpha_f). \quad (15)$$

By accounting for the fact that $\mathcal{GHS}_p(g, \alpha_s, \alpha_f)$ itself is expressed via a minimization, we can also write the above problem as,

$$(g_{opt}, \mathbf{u}_{opt}) = \operatorname*{argmin}_{g,\mathbf{u}} \ G(h * g, \mathcal{T}_s, \mathbf{m}) + \mathcal{PGHS}_p(g, \mathbf{u}, \alpha_s, \alpha_f), \quad (16)$$

where

$$\mathcal{PGHS}_p(g, \mathbf{u}, \alpha_s, \alpha_f) = \alpha_f \|\mathbf{d} * g - \mathbf{u}\|_{1,2} + \alpha_s \|0.5(\mathbf{u} * \mathbf{d}^t + \mathbf{d} * \mathbf{u}^t)\|_{1,S(p)}. \quad (17)$$

For notational convenience in developing the minimization algorithm, we re-express the cost in terms of the combined variable $\mathbf{v}(\mathbf{r}) = [g(\mathbf{r}) \ \ \mathbf{u}^t(\mathbf{r})]^t$. To this end, we define the following:

$$\mathbf{h}(\mathbf{r}) = [h(\mathbf{r}) \ \ 0 \ \ 0]^t, \quad (18)$$

$$\mathbf{T}_f(\mathbf{r}) = \begin{bmatrix} d_x(\mathbf{r}) & 0 & 0 \\ d_y(\mathbf{r}) & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}, \quad (19)$$

$$\mathbf{T}_s(\mathbf{r}) = \begin{bmatrix} 0 & d_x(\mathbf{r}) & 0 \\ 0 & d_y(\mathbf{r}) & 0 \\ 0 & 0 & d_x(\mathbf{r}) \\ 0 & 0 & d_y(\mathbf{r}) \end{bmatrix}, \text{ and} \tag{20}$$

$$\mathbf{A}_f = \begin{bmatrix} 1/\sqrt{2} & 0 & -1/\sqrt{2} & 0 \\ 0 & 1/\sqrt{2} & 0 & -1/\sqrt{2} \end{bmatrix}. \tag{21}$$

We will overload the notation $\|\cdot\|_{S(p)}$, such that when used with a 4×1 vector $\mathbf{y} = [y_1 \ y_2 \ y_3 \ y_4]^t$ as the argument, it represent the following:

$$\|\mathbf{y}\|_{S(p)} = \left\| \begin{bmatrix} y_1 & y_2 \\ y_3 & y_4 \end{bmatrix} \right\|_{S(p)}.$$

Let $\mathbf{K}$ be the matrix such that $\mathbf{K}y = [y_1 \ \ 0.5(y_2+y_3) \ \ 0.5(y_2+y_3) \ \ y_4]^t$. With these, the overall cost to be minimized can be expressed as

$$J(\mathbf{v}, \alpha_s, \alpha_f, p) = G(\mathbf{h} * \mathbf{v}, \mathcal{T}_s, \mathbf{m}) + \alpha_f \sqrt{2} \|\mathbf{A}_f(\mathbf{T}_f * \mathbf{v})\|_{1,2} + \alpha_s \|\mathbf{K}(\mathbf{T}_s * \mathbf{v})\|_{1,S(p)}. \tag{22}$$

Note that definition of $\mathbf{A}_f$ creates an additional scale factor of $\sqrt{2}$ in the middle term in (22). But, this does not create any difference in the formulation as α_f is an independent parameter, which is tuned for optimum reconstruction quality. Hence we absorb the scaling $\sqrt{2}$ into the parameter α_f. Henceforth, the weight parameter in the middle term will be α_f instead of $\alpha_f \sqrt{2}$.

Note that $J(\mathbf{v}, \alpha_s, \alpha_f, p)$ with $p = 2$ is the well-known TGV-2 regularization. Guo *et al.* [53] developed an algorithm for image reconstruction using TGV-2 and shearlet regularization. With the shearlet part removed, their algorithm corresponds to the following constrained formulation of the reconstruction problem,

$$(\mathbf{v}^*, \mathbf{w}_s^*, \mathbf{w}_f^*) = \underset{\mathbf{v}, \mathbf{w}_s, \mathbf{w}_f}{\operatorname{argmin}} \ G(\mathbf{h} * \mathbf{v}, \mathcal{T}_s, \mathbf{m}) + \alpha_f \|\mathbf{w}_f\|_{1,2} + \alpha_s \|\mathbf{w}_s\|_{1,S(2)} \tag{23}$$

$$\text{subject to} \ \ \mathbf{w}_s = \mathbf{K}(\mathbf{T}_s * \mathbf{v}), \quad \mathbf{w}_f = \mathbf{A}_f(\mathbf{T}_f * \mathbf{v}), \tag{24}$$

except the fact that $\mathcal{T}_s$ will simply pick all the pixels of the image. The main advantage of this form is that the ADMM algorithm can be constructed using well known proximal operators. However, the algorithm can be badly conditioned in some cases as we will demonstrate in the experiment section.

B. PROPOSED ADMM METHOD

Before specifying the variable splitting scheme for the proposed ADMM algorithm, we first rewrite the cost of the equation (22) as

$$J(\mathbf{v}, \alpha_s, \alpha_f, p) = G(\mathbf{h} * \mathbf{v}, \mathcal{T}_s, \mathbf{m}) + \alpha_f \|\mathbf{A}_f(\mathbf{T}_f * \mathbf{v})\|_{1,2} + \alpha_s \|\mathbf{K}(\mathbf{T}_s * \mathbf{v})\|_{1,S(p)} + \mathcal{B}(\mathbf{e}^t * \mathbf{v}), \tag{25}$$

where

$$\mathcal{B}(g) = \begin{cases} 0 & LB \le g_i \le UB \quad \forall i \\ \infty & \text{otherwise} \end{cases}$$

enforces pixelwise lowerbound (LB) and upperbound (UB). Also, $\mathbf{e}^t(\mathbf{r}) = [\delta(\mathbf{r}) \ 0 \ 0]$ with $\delta(\mathbf{r})$ denoting Kronecker delta, and hence $\mathbf{e}^t * \mathbf{v} = v_1 = g$. We propose to build ADMM algorithm by means of the following constrained formulation:

$$\begin{aligned} (\mathbf{v}^*, \mathbf{w}_s^*, \mathbf{w}_f^*, w_b^*, w_m^*) = \underset{\mathbf{v}, \mathbf{w}_s, \mathbf{w}_f, w_b, w_m}{\operatorname{argmin}} \ & G(w_m, \mathcal{T}_s, \mathbf{m}) \\ & + \alpha_f \|\mathbf{A}_f \mathbf{w}_f\|_{1,2} + \alpha_s \|\mathbf{K}\mathbf{w}_s\|_{1,S(p)} + \mathcal{B}(w_b) \\ \text{subject to} \ \ & \mathbf{w}_s = \mathbf{T}_s * \mathbf{v}, \quad \mathbf{w}_f = \mathbf{T}_f * \mathbf{v}, \\ & \mathbf{e}^t * \mathbf{v} = w_b, \quad \mathbf{h} * \mathbf{v} = w_m \end{aligned} \tag{26}$$

The main difference from the previous formulation [53] is that the matrices $\mathbf{A}_f$ and $\mathbf{K}$ are not a part of the constraints but, are left as a part of the cost to be minimized. This leads to some numerical advantages, which we will clarify after completing the development of the algorithm. On the other hand, this splitting scheme requires constructing new proximal operators for implementing the ADMM algorithm. We do this in the next section, which is one of the important contributions of the paper. In the remainder of this section, we complete specifying the ADMM iteration for the splitting scheme specified above. In order to facilitate developing the algorithm in terms of compact expressions, we need to simplify the notations further. To this end, let

$$\mathbf{w} = [w_b \ \mathbf{w}_s^t \ \mathbf{w}_f^t \ w_m]^t \text{ and} \tag{27}$$

$$\mathbf{T}(\mathbf{r}) = \begin{bmatrix} \mathbf{e}^t(\mathbf{r}) \\ \mathbf{T}_s(\mathbf{r}) \\ \mathbf{T}_f(\mathbf{r}) \\ \mathbf{h}(\mathbf{r}) \end{bmatrix}. \tag{28}$$

With this, the above problem can be expressed as

$$(\mathbf{v}^*, \mathbf{w}^*) = \underset{\mathbf{v}, \mathbf{w}}{\operatorname{argmin}} \ R(\mathbf{w}, \alpha_s, \alpha_f, p) \quad \text{subject to} \ \ \mathbf{w} = \mathbf{T} * \mathbf{v}, \tag{29}$$

where

$$R(\mathbf{w}, \alpha_s, \alpha_f, p) = \alpha_f \|\mathbf{A}_f \mathbf{w}_f\|_{1,2} + \alpha_s \|\mathbf{K}\mathbf{w}_s\|_{1,S(p)} + \mathcal{B}(w_b) + G(w_m, \mathcal{T}_s, \mathbf{m}) \tag{30}$$

The next step towards developing the ADMM algorithm is to write the augmented Lagrangian of the above constrained optimization problem. The augmented Lagrangian is given by

$$L(\mathbf{v}, \mathbf{w}, \boldsymbol{\lambda}, \alpha_s, \alpha_f, p) = R(\mathbf{w}, \alpha_s, \alpha_f, p) + \frac{\beta}{2} \|\mathbf{T} * \mathbf{v} - \mathbf{w}\|_2^2 + \langle \boldsymbol{\lambda}, \mathbf{T} * \mathbf{v} - \mathbf{w} \rangle, \tag{31}$$

where $\boldsymbol{\lambda}$ is vector image of Lagrange's multiplier with its dimension equal to that of $\mathbf{w}$. Also, β is a fixed positive real number. The ADMM becomes series of minimizations with respect to $\mathbf{v}$ and $\mathbf{w}$ and updates on $\boldsymbol{\lambda}$. Given the current set of iterates, $\{\mathbf{v}^{(k)}, \mathbf{w}^{(k)}, \boldsymbol{\lambda}^{(k)}\}$ the ADMM methods proceeds as follows:

$$\mathbf{w}^{(k+1)} = \underset{\mathbf{w}}{\operatorname{argmin}} \ L(\mathbf{v}^{(k)}, \mathbf{w}, \boldsymbol{\lambda}^{(k)}, \alpha_s, \alpha_f, p), \tag{32}$$

$$\mathbf{v}^{(k+1)} = \underset{\mathbf{v}}{\operatorname{argmin}} \ L(\mathbf{v}, \mathbf{w}^{(k+1)}, \boldsymbol{\lambda}^{(k)}, \alpha_s, \alpha_f, p), \tag{33}$$

and

$$\boldsymbol{\lambda}^{(k+1)} = \boldsymbol{\lambda}^{(k)} + \beta\left(\mathbf{T} * \mathbf{v}^{(k+1)} - \mathbf{w}^{(k+1)}\right). \tag{34}$$

Results on covergence of ADMM scheme ensure that above iteration converges to the minimum provided that the following conditions are satisfied [54]:

- A1: The function $R(\mathbf{w}, \alpha_s, \alpha_f, p)$ must be proper, closed and convex.
- A2: The Lagrangian of the problem

$$\mathcal{L}(\mathbf{v}, \mathbf{w}, \boldsymbol{\lambda}, \alpha_s, \alpha_f, p) = R(\mathbf{w}, \alpha_s, \alpha_f, p) + \langle \boldsymbol{\lambda}, \mathbf{T} * \mathbf{v} - \mathbf{w} \rangle \tag{35}$$

should have a saddle point.

Since all the constituent functions in $R(\mathbf{w}, \alpha_s, \alpha_f, p)$ are proper, closed, and convex, it is also closed, proper and convex. This takes care of A1. Next, since we have a linear constrained problem, any point satisfying the equality is in the relative interior of the constraint set $\mathbf{T} * \mathbf{v} = \mathbf{w}$. This implies all points satisfying the equality satisfy Slaters constrain qualification [55]. Hence by the saddle point theorem [55], a saddle point exists for the Lagrangian. This takes care of A2. In summary, the proposed iterative scheme converges to the minimum.

IV. SOLVING THE SUB-PROBLEMS OF ADMM

A. THE w-PROBLEM

1) Expressing THE PIXEL-WISE SUB-PROBLEMS

Note that the solution to the minimization problem of equation (32), $\mathbf{w}^{(k+1)}$, is also the minimum of the following cost:

$$L_{w,k}(\mathbf{w}, \alpha_s, \alpha_f, p) = R(\mathbf{w}, \alpha_s, \alpha_f, p) + \frac{\beta}{2}\|\mathbf{w} - \mathbf{x}^{(k)}\|_2^2, \tag{36}$$

where,

$$\bar{\mathbf{x}}^{(k)} = \mathbf{T} * \mathbf{v}^{(k)} + \frac{1}{\beta}\boldsymbol{\lambda}^{(k)}. \tag{37}$$

For notational convenience, let $\mathbf{x} = \mathbf{x}^{(k)}$ and let $\hat{\mathbf{w}} = \mathbf{w}^{(k+1)}$. Note that the cost $L_{w,k}(\mathbf{w}, \alpha_s, \alpha_f, p)$ is separable across the subvectors of the variable, $\mathbf{w}$; in other words, it is separable across the subvectors w_b, w_m, $\mathbf{w}_s$, and $\mathbf{w}_f$, and we introduced the collective variable, $\mathbf{w}$ only for notational convenience in expressing the ADMM loop of equations (32), (33), (34). As we are focused on this specific sub-problem, we will now separate the constituent problems. To this end, let x_b, x_m, $\mathbf{x}_s$, and $\mathbf{x}_f$ denote the sub-vectors of $\mathbf{x}$ conferring to the partitioning given in the equation (27). Similarly, let $\hat{w}_b$, $\hat{w}_m$, $\hat{\mathbf{w}}_s$, and $\hat{\mathbf{w}}_f$ be the subvectors of $\hat{\mathbf{w}}$. With this the solution to the $\mathbf{w}$-problem can be expressed as

$$w_m\text{-prob.: } \hat{w}_m = \underset{w_m}{\operatorname{argmin}} \underbrace{\frac{\beta}{2}\|x_m - w_m\|_{2,2}^2 + G(w_m, \mathcal{T}_s, \mathbf{m})}_{\bar{L}_m(w_m, x_m)} \tag{38}$$

$$w_b\text{-prob.: } \hat{w}_b = \underset{w_b}{\operatorname{argmin}} \underbrace{\frac{\beta}{2}\|x_b - w_b\|_{2,2}^2 + \mathcal{B}(w_b)}_{\bar{L}_b(w_b, x_b)} \tag{39}$$

$$\mathbf{w}_s\text{-prob.: } \hat{\mathbf{w}}_s = \underset{\mathbf{w}_s}{\operatorname{argmin}} \underbrace{\frac{\beta}{2}\|\mathbf{x}_s - \mathbf{w}_s\|_{2,2}^2 + \alpha_s\|\mathbf{K}\mathbf{w}_s\|_{1,S(p)}}_{\bar{L}_s(\mathbf{w}_s, \mathbf{x}_s, \alpha_s)} \tag{40}$$

$$\mathbf{w}_f\text{-prob.: } \hat{\mathbf{w}}_f = \underset{\mathbf{w}_f}{\operatorname{argmin}} \underbrace{\frac{\beta}{2}\|\mathbf{x}_f - \mathbf{w}_f\|_{2,2}^2 + \alpha_s\|\mathbf{A}_f\mathbf{w}_f\|_{1,2}}_{\bar{L}_f(\mathbf{w}_f, \mathbf{x}_f, \alpha_f)} \tag{41}$$

Note that the functions $\bar{L}_b(w_b, x_b)$, $\bar{L}_s(\mathbf{w}_s, \mathbf{x}_s, \alpha_s)$, and $\bar{L}_f(\mathbf{w}_f, \mathbf{x}_f, \alpha_f)$ are separable across pixel indices since they are constructed as mixed norms composed elementary pixel-wise norms. Hence, they can be expressed as a sum of pixel-wise elementary functions. The form of these elementary functions can be clearly deduced from the form of the functions $\bar{L}_b(w_b, x_b)$, $\bar{L}_s(\mathbf{w}_s, \mathbf{x}_s, \alpha_s)$, and $\bar{L}_f(\mathbf{w}_f, \mathbf{x}_f, \alpha_f)$. We can express these functions as

$$\bar{L}_b(w_b, x_b) = \sum_{\mathbf{r}} \underbrace{\frac{\beta}{2}(x_b(\mathbf{r}) - w_b(\mathbf{r}))^2 + \mathcal{B}(w_b(\mathbf{r}))}_{L_b(w_b(\mathbf{r}), x_b(\mathbf{r}))}, \tag{42}$$

$$\bar{L}_s(\mathbf{w}_s, \mathbf{x}_s, \alpha_s) = \sum_{\mathbf{r}} \underbrace{\frac{\beta}{2}\|\mathbf{x}_s(\mathbf{r}) - \mathbf{w}_s(\mathbf{r})\|_2^2 + \alpha_s\|\mathbf{K}w_s(\mathbf{r})\|_{S(p)}}_{L_s(\mathbf{w}_s(\mathbf{r}), \mathbf{x}_s(\mathbf{r}), \alpha_s)}, \tag{43}$$

and

$$\bar{L}_f(\mathbf{w}_f, \mathbf{x}_f, \alpha_f) = \sum_{\mathbf{r}} \underbrace{\frac{\beta}{2}\|\mathbf{x}_f(\mathbf{r}) - \mathbf{w}_f(\mathbf{r})\|_2^2 + \alpha_f\|\mathbf{A}_f(\mathbf{w}_f(\mathbf{r}))\|_2}_{L_f(\mathbf{w}_f(\mathbf{r}), \mathbf{x}_f(\mathbf{r}), \alpha_f)}. \tag{44}$$

Now we consider the w_m-problem. First we rewrite the cost:

$$\bar{L}_m(w_m, x_m) = \frac{\beta}{2}\|x_m - w_m\|_{2,2}^2 + G(w_m, \mathcal{T}_s, \mathbf{m}) \tag{45}$$

$$= \frac{\beta}{2}\|x_m - w_m\|_{2,2}^2 + \|\mathcal{T}_s(w_m) - \mathbf{m}\|_2^2 \tag{46}$$

As mentioned before, we will consider a restricted case where $\mathcal{T}_s$ picks samples from locations that constitute a subset of a regular grid. In this case, $\|\mathcal{T}_s(w_m) - \mathbf{m}\|_2^2$ can written as

$$\|\mathcal{T}_s(w_m) - \mathbf{m}\|_2^2 = \frac{1}{2}\sum_{\mathbf{r}} W(\mathbf{r})(w_m(\mathbf{r}) - m(\mathbf{r}))^2,$$

where $W(\mathbf{r})$ is the image that has ones at pixels locations from where $\mathcal{T}_s$ picks samples, and zeros at other location. Further, $m(\mathbf{r})$ is the image obtained by embedding measured samples $\mathbf{m}$ into an image of zeros. Further, we note that we can express $\|x_m - w_m\|_{2,2}^2$ can be written as

$$\|x_m - w_m\|_{2,2}^2 = \sum_{\mathbf{r}} (x_m(\mathbf{r}) - w_m(\mathbf{r}))^2.$$

Hence, the cost $\bar{L}_m(w_m, x_m)$ can be written as given below:

$$\bar{L}_m(w_m, x_m) = \frac{\beta}{2}\sum_{\mathbf{r}}(x_m(\mathbf{r}) - w_m(\mathbf{r}))^2 + \sum_{\mathbf{r}} W(\mathbf{r})(w_m(\mathbf{r}) - m(\mathbf{r}))^2 \quad (47)$$

From the above equation, it is clear that the minimization problem is separable across pixels. We denote the pixel-wise cost by $L_m(\cdot,\cdot)$ and write $\bar{L}_m(w_m, x_m) = \sum_{\mathbf{r}} L_m(w_m(\mathbf{r}), x_m(\mathbf{r}))$ where

$$L_m(w_m(\mathbf{r}), x_m(\mathbf{r})) = \frac{\beta}{2}(x_m(\mathbf{r}) - w_m(\mathbf{r}))^2 + W(\mathbf{r})(w_m(\mathbf{r}) - m(\mathbf{r}))^2.$$

We have shown so far that all the sub-problems are separable across pixels. Hence the solution to the minimization problems of equations (38), (39), (40), and (41), can be expressed as following:

$$\hat{w}_m(\mathbf{r}) = \underset{z\in\mathbb{R}}{\text{argmin}}\ L_m(z, x_m(\mathbf{r})), \quad (48)$$

$$\hat{w}_b(\mathbf{r}) = \underset{z\in\mathbb{R}}{\text{argmin}}\ L_b(z, x_b(\mathbf{r})), \quad (49)$$

$$\hat{\mathbf{w}}_s(\mathbf{r}) = \underset{\mathbf{z}\in\mathbb{R}^4}{\text{argmin}}\ L_s(\mathbf{z}, \mathbf{x}_s(\mathbf{r}), \alpha_s), \text{ and} \quad (50)$$

$$\hat{\mathbf{w}}_f(\mathbf{r}) = \underset{\mathbf{z}\in\mathbb{R}^4}{\text{argmin}}\ L_f(\mathbf{z}, \mathbf{x}_f(\mathbf{r}), \alpha_f). \quad (51)$$

2) Solution TO THE PIXEL-WISE SUB-PROBLEMS

The solution to the w_m-problem very simple. The solution $\hat{w}_m(\mathbf{r})$ can be expressed as

$$\hat{w}_m(\mathbf{r}) = \frac{\beta x_m(\mathbf{r}) + W(\mathbf{r})m(\mathbf{r})}{\beta + W(\mathbf{r})}.$$

The solution to the w_b-problem is also simple, and it is the clipping of the pixels by bound that defines $\mathcal{B}(w_b)$ [56]. We express the solution as given below:

$$\hat{w}_b(\mathbf{r}) = \mathbb{P}_b(x_b(\mathbf{r})), \quad (52)$$

where $\mathbb{P}_b(\cdot)$ denotes the operation of clipping the pixel values within the specified bounds. Next, we consider expressing the solution for $\mathbf{w}_f$-problem. Note that, in the absence of the matrix $\mathbf{A}_f$, the solution will be the well-known shrinkage operation on the l_2 norm of $\mathbf{x}_f(\mathbf{r})$ [56]. Because of the presence of the matrix, $\mathbf{x}_f(\mathbf{r})$, the shrinkage operation is not directly applicable. The following proposition give the solution to the $\mathbf{w}(\mathbf{r})$ problem.

Proposition 2: The minimum of $L_f(\mathbf{z}, \mathbf{a}, \alpha_f) = \frac{\beta}{2}\|\mathbf{z} - \mathbf{a}\|_2^2 + \alpha_f\|\mathbf{A}_f\mathbf{z}\|_2$ with respect to $\mathbf{z}$ is given by $\mathcal{P}(\mathbf{a}, t) = \mathbf{a} - \min(\|\mathbf{A}_f\mathbf{a}\|_2, t)\mathbf{A}_f^t\mathbf{A}_f\mathbf{a}$, where $t = \alpha_f/\beta$.

Now, considering the $\mathbf{w}_s$ problem, i.e., considering the minimization of $\frac{\beta}{2}\|\mathbf{z} - \mathbf{a}\|_2^2 + \alpha_s\|\mathbf{K}z\|_{S(p)}$, the presence of $\mathbf{K}$ makes the problem more complex. Otherwise, the solution to the problem in the absence of $\mathbf{K}$ is well-known [46]. The following proposition gives expression for solution to this problem.

Proposition 3: The minimum of $L_s(\mathbf{z}, \mathbf{a}, \alpha_s) = \frac{1}{2}\|\mathbf{z} - \mathbf{a}\|_2^2 + t\|\mathbf{K}z\|_{S(p)}$ with respect to $\mathbf{z}$ is given by $\bar{\mathcal{P}}_s(\mathbf{a}, t) = \mathbf{L}\mathbf{a} + \mathcal{P}_s(\mathbf{K}\mathbf{a}, t)$ where $\mathcal{P}_s(\mathbf{b}, t)$ represents the minimum of the $\frac{1}{2}\|\mathbf{z}-\mathbf{b}\|_2^2 + t\|\mathbf{z}\|_{S(p)}$ with respect to $\mathbf{z}$, and and $\mathbf{L}$ is the matrix such that its operations on $\mathbf{a} = [a_1\ a_2\ a_3\ a_4]^t$ is defined as $\mathbf{L}\mathbf{a} = [0\ \ 0.5(a_2 - a_3)\ \ 0.5(a_3 - a_2)\ \ 0]^t$.

Note that $\mathcal{P}_s(\cdot, t)$ is the well-known proximal operator of Schatten norm (for details see [46]), and the above proposition expresses the required proximal operator—proximal operator for the modified Schatten norm, $\bar{\mathcal{P}}_s(\cdot, t)$—as a simple modification of $\mathcal{P}_s(\cdot, t)$. Note that we just need set $t = \alpha_s/\beta$ for applying the above proposition for the problem of equation (50). Although $\mathcal{P}_s(\cdot, t)$ is well known, in **Algorithm 3**, we provide detailed description of $\mathcal{P}_s(\cdot, t)$ to the level required for implementation along with a self-contained description of the overall algorithm. Note that, we have non-iterative exact formula for $\mathcal{P}_s(\cdot, t)$ only for $p = 1, 2$. Hence, in the experimental demonstration, we only consider these two values for p.

B. THE v-PROBLEM

Now we consider the minimization problem of equation (33). By taking the dependencies of the minimization variable $\mathbf{v}$, it can be deduced that, the solution to this problem is also the minimum of the following cost function:

$$L_{v,k}(\mathbf{v}) = \frac{1}{2}\|\mathbf{T} * \mathbf{v} - \mathbf{y}^{(k)}\|_2^2, \quad (53)$$

where

$$\mathbf{y}^{(k)} = \mathbf{w}^{(k+1)} - \frac{1}{\beta}\boldsymbol{\lambda}^{(k)}. \quad (54)$$

As done before for notational convenience, we let $\mathbf{y} = \mathbf{y}^{(k)}$, and $\hat{\mathbf{v}} = \mathbf{v}^{(k+1)}$. Let $\mathbf{v}(\mathbf{r}) = [v_1(\mathbf{r})\ \ v_2(\mathbf{r})\ \ v_3(\mathbf{r})]^t$, $\hat{\mathbf{v}}(\mathbf{r}) = [\hat{v}_1(\mathbf{r})\ \ \hat{v}_2(\mathbf{r})\ \ \hat{v}_3(\mathbf{r})]^t$. From the definition of $\mathbf{T}(\mathbf{r})$ given in the equation (28), and from the definition of $\mathbf{h}(\mathbf{r})$, it is clear that the cost $L_{v,k}(\mathbf{v})$ is separable across the components of $\mathbf{v}(\mathbf{r})$, which are $v_1(\mathbf{r})$, $v_2(\mathbf{r})$, and $v_3(\mathbf{r})$. By letting $\mathbf{y}(\mathbf{r}) = [y_b(\mathbf{r})\ \ y_{f,1}(\mathbf{r})\ \ y_{f,2}(\mathbf{r})\ \ y_{f,3}(\mathbf{r})\ \ y_{f,4}(\mathbf{r})\ \ y_{s,1}(\mathbf{r})\ \ y_{s,2}(\mathbf{r})\ \ y_{s,3}(\mathbf{r})\ \ y_{s,4}(\mathbf{r})\ \ y_m(\mathbf{r})]^t$, the costs separated for v_1, v_2, and v_3 are given by

$$\bar{L}_1(v_1) = \frac{1}{2}\|h * v_1 - y_m\|_{2,2}^2 + \frac{1}{2}\|v_1 - y_b\|_{2,2}^2 + \frac{1}{2}\|d_x * v_1 - y_{f,1}\|_{2,2}^2 + \frac{1}{2}\|d_y * v_1 - y_{f,2}\|_{2,2}^2, \quad (55)$$

$$\bar{L}_2(v_2) = \frac{1}{2}\|v_2 - y_{f,3}\|_{2,2}^2 + \frac{1}{2}\|d_x * v_2 - y_{s,1}\|_{2,2}^2 + \frac{1}{2}\|d_y * v_2 - y_{s,2}\|_{2,2}^2, \text{ and} \quad (56)$$

$$\bar{L}_3(v_3) = \frac{1}{2}\|v_3 - y_{f,4}\|_{2,2}^2 + \frac{1}{2}\|d_x * v_3 - y_{s,3}\|_{2,2}^2 + \frac{1}{2}\|d_y * v_3 - y_{s,4}\|_{2,2}^2. \quad (57)$$

These are quadratic functions involving simple discrete filtering operation. Hence, their minima can be expressed in

terms of simple Fourier inversion. To this end, we write the expression for the gradients below:

$$\begin{aligned}\nabla_{v_1}\bar{L}_1(v_1) &= Re(conj(\tilde{h}) * h) * v_1 \\ &\quad + (v_1 + \tilde{d}_x * d_x * v_1 + \tilde{d}_y * d_y * v_1) \\ &\quad - Re(conj(\tilde{h}) * y_m) \\ &\quad - (y_b + \tilde{d}_x * y_{f,1} + \tilde{d}_y * y_{f,2}), \qquad (58)\\ \nabla_{v_2}\bar{L}_2(v_2) &= (v_2 + \tilde{d}_x * d_x * v_2 + \tilde{d}_y * d_y * v_2) \\ &\quad - (y_{f,3} + \tilde{d}_x * y_{s,1} + \tilde{d}_y * y_{s,2}), \text{ and} \qquad (59)\\ \nabla_{v_3}\bar{L}_3(v_3) &= (v_3 + \tilde{d}_x * d_x * v_3 + \tilde{d}_y * d_y * v_3) \\ &\quad - (y_{f,4} + \tilde{d}_x * y_{s,3} + \tilde{d}_y * y_{s,4}). \qquad (60)\end{aligned}$$

In the above equation, $conj(\cdot)$ represents pointwise complex conjugate operation. Now, the minima $\hat{v}_1(\mathbf{r})$, $\hat{v}_2(\mathbf{r})$, and $\hat{v}_3(\mathbf{r})$ can be obtained by solving $\nabla_{v_1}\bar{L}_1(v_1) = 0$, $\nabla_{v_2}\bar{L}_2(v_2) = 0$, and $\nabla_{v_3}\bar{L}_3(v_3) = 0$ in Fourier domain. This is the main advantage of the proposed variable splitting: we are able to solve for $\hat{v}_1(\mathbf{r})$, $\hat{v}_2(\mathbf{r})$, and $\hat{v}_3(\mathbf{r})$ independently by simple Fourier division. On the other hand, the $\mathbf{v}$-problem encountered in the splitting proposed in [53], the components v_1, v_2, and v_3 are coupled, and hence, it requires solving 3×3 system of equations each of $L \times L$ frequency values where $L \times L$ is the size of the image to be reconstructed. Although the required inverses can be precomputed outside the ADMM loop, it turns out that overall numerical conditioning is worse than the splitting that we propose here, which we demonstrate experimentally in Section V. In **Algorithm 1**, we provide detailed expression of the solutions of $\nabla_{v_1}\bar{L}_1(v_1) = 0$, $\nabla_{v_2}\bar{L}_2(v_2) = 0$, and $\nabla_{v_3}\bar{L}_3(v_3) = 0$ to the level required for implementation. **Algorithm 1** also serves as the self-contained specification of the overall proposed reconstruction method.

V. EXPERIMENTS

Recall that we have two main contributions in this paper: (i) a generalization of Hessian-Schatten p-norm [44] with the resulting form that also generalizes second-order total generalized variation regularization (TGV-2) [49]; (ii) a novel ADMM based reconstruction algorithm with improved numerical behaviour owing to the novel variable splitting scheme. Implementing the algorithm with novel variable splitting is enabled by novel proximal operators derived in this paper. The goal of this section is to demonstrate the role of both of these contributions in improving the quality of reconstruction. To this end, we consider two types of measurement models:

- Quasi-random Fourier sampling scheme for the realization of fast MRI imaging
- Blurring followed random spatial sampling as a scheme for speeding-up scanning electron microscopy

For the first model, suppose $\mathcal{T}$ denotes the operation of obtaining the vector of Fourier samples from Fourier locations, and let $\mathbf{m}_f$ be the vector of complex Fourier samples. Since noise is Gaussian in MR imaging, for a candidate image g, the cost $\|\mathcal{T}g - \mathbf{m}_f\|_2^2$ is the negative log-likelihood. With this, we make a simplifying assumption that Fourier

Algorithm 1 GHSN Regularized Reconstruction: *GHSN-RR*$(m, h, \alpha_0, \alpha_1, Niter)$

Inputs:
m: $N \times N$ measured image
h: Impulse response of imaging system
α_f: regularization weight for first order term
α_s: regularization weight for second order term
Niter: Number of Iterations
Symbol definitions: d_x and d_y are first derivative filters
$\tilde{d}_x$ and $\tilde{d}_y$ are flipped versions
$\tilde{h}$ is flipped version of h

Initialization :
$k \leftarrow 0, \quad \mathbf{v}^{(k)} \leftarrow \mathbf{0}_{3\times N\times N}$
$\lambda_b^{(k)} \leftarrow \mathbf{0}_{N\times N}, \quad \lambda_m^{(k)} \leftarrow \mathbf{0}_{N\times N},$
$\boldsymbol{\lambda}_f^{(k)} \leftarrow \mathbf{0}_{4\times N\times N}, \quad \boldsymbol{\lambda}_s^{(k)} \leftarrow \mathbf{0}_{4\times N\times N}$
$\bar{x}_b^{(k)} \leftarrow \mathbf{0}_{N\times N}, \quad \bar{x}_m^{(k)} \leftarrow \mathbf{0}_{N\times N}$
$\bar{\mathbf{x}}_f^{(k)} \leftarrow \mathbf{0}_{4\times N\times N}, \quad \bar{\mathbf{x}}_s^{(k)} \leftarrow \mathbf{0}_{4\times N\times N}$
$Q_d \leftarrow \mathcal{F}\{Re(h * conj(\tilde{h}))\}$
$Q_r \leftarrow 1 + \mathcal{F}(d_x * \tilde{d}_x + d_x * \tilde{d}_x)$
for 1 *to* *Niter* **do**
$x_b^{(k)} \leftarrow \bar{x}_b^{(k)} + (1/\beta)\lambda_b^{(k)}, \quad x_m^{(k)} \leftarrow \bar{x}_m^{(k)} + (1/\beta)\lambda_m^{(k)}$
$\mathbf{x}_f^{(k)} \leftarrow \bar{\mathbf{x}}_f^{(k)} + (1/\beta)\boldsymbol{\lambda}_f^{(k)}, \quad \mathbf{x}_s^{(k)} \leftarrow \bar{\mathbf{x}}_s^{(k)} + (1/\beta)\boldsymbol{\lambda}_s^{(k)}$
$w_b^{(k+1)} \leftarrow \mathbb{P}_b(x_b^{(k)})$ (Clipping of elements)
$w_m^{(k+1)} \leftarrow (\beta x_m^{(k)} + Wm)/(\beta + W)$
$(\mathbf{w}_f^{(k+1)}, \mathbf{w}_s^{(k+1)}) \leftarrow \text{Prox}(\mathbf{x}_f^{(k)}, \mathbf{x}_s^{(k)}, \alpha_f, \alpha_s)$ (**Alg. 2**)
$y_b^{(k+1)} \leftarrow w_b^{(k+1)} - (1/\beta)\lambda_b^{(k)}$
$y_m^{(k+1)} \leftarrow w_m^{(k+1)} - (1/\beta)\lambda_m^{(k)}$
$\mathbf{y}_f^{(k+1)} \leftarrow \mathbf{w}_f^{(k+1)} - (1/\beta)\boldsymbol{\lambda}_f^{(k)}$
$\mathbf{y}_s^{(k+1)} \leftarrow \mathbf{w}_s^{(k+1)} - (1/\beta)\boldsymbol{\lambda}_s^{(k)}$
$B_1 \leftarrow \mathcal{F}(y_b^{(k+1)} + \tilde{d}_x * y_{f,1}^{(k+1)} + \tilde{d}_y * y_{f,2}^{(k+1)})$
$B_2 \leftarrow \mathcal{F}(y_{f,3}^{(k+1)} + \tilde{d}_x * y_{s,1}^{(k+1)} + \tilde{d}_y * y_{s,2}^{(k+1)})$
$B_3 \leftarrow \mathcal{F}(y_{f,4}^{(k+1)} + \tilde{d}_x * y_{s,3}^{(k+1)} + \tilde{d}_y * y_{s,4}^{(k+1)})$
$B_d \leftarrow \mathcal{F}\{Re(conj(\tilde{h}) * y_m^{(k+1)})\}$
$v_1^{(k+1)} \leftarrow \mathcal{F}^{-1}((B_d + B_1)/(Q_d + Q_r))$
$v_2^{(k+1)} \leftarrow \mathcal{F}^{-1}(B_2/Q_r)$
$v_3^{(k+1)} \leftarrow \mathcal{F}^{-1}(B_3/Q_r)$
$\bar{x}_b^{(k+1)} \leftarrow v_1^{(k+1)}, \quad \bar{x}_m^{(k+1)} \leftarrow h * v_1^{(k+1)}$
$\bar{\mathbf{x}}_f^{(k+1)} \leftarrow [v_1^{(k+1)} * d_x, v_1^{(k+1)} * d_y, v_2^{(k+1)}, v_3^{(k+1)}]^t$
$\bar{\mathbf{x}}_s^{(k+1)} \leftarrow [v_2^{(k+1)} * d_x, v_2^{(k+1)} * d_y, v_3^{(k+1)} * d_x, v_3^{(k+1)} * d_y]^t$
$\lambda_b^{(k+1)} \leftarrow \lambda_b^{(k)} + \beta(\bar{x}_b^{(k+1)} - w_b^{(k+1)})$
$\lambda_m^{(k+1)} \leftarrow \lambda_m^{(k)} + \beta(\bar{x}_m^{(k+1)} - w_m^{(k+1)})$
$\boldsymbol{\lambda}_f^{(k+1)} \leftarrow \boldsymbol{\lambda}_f^{(k)} + \beta(\bar{\mathbf{x}}_f^{(k+1)} - \mathbf{w}_f^{(k+1)})$
$\boldsymbol{\lambda}_s^{(k+1)} \leftarrow \boldsymbol{\lambda}_s^{(k)} + \beta(\bar{\mathbf{x}}_s^{(k+1)} - \mathbf{w}_s^{(k+1)})$
$k \leftarrow k + 1$
end
return $y = x^{(k)}$

Algorithm 2 Proximal Operations: $(\hat{\mathbf{a}}, \hat{\mathbf{c}}) \leftarrow Prox(\mathbf{a}, \mathbf{b}, \alpha_f, \alpha_s, p)$

Inputs:
$\mathbf{a} = [a_1\ a_2\ a_3\ a_4]^t$: 4×1 vector image corresponding to first order derivative
$\mathbf{b} = [b_1\ b_2\ b_3\ b_4]^t$: 4×1 vector image corresponding to second order derivative
α_f: regularization weight for second order term
α_s: regularization weight for second order term

for $\mathbf{r} \in [1, N]^2$ **do**

$d_1(\mathbf{r}) \leftarrow a_1(\mathbf{r}) - a_3(\mathbf{r}), \quad d_2(\mathbf{r}) \leftarrow a_2(\mathbf{r}) - a_4(\mathbf{r})$
$n(\mathbf{r}) \leftarrow \sqrt{0.5(d_1^2(\mathbf{r}) + d_2^2(\mathbf{r}))}$
$n(\mathbf{r}) \leftarrow 0.5 \min(1, \alpha_f / n(\mathbf{r}))$
$d_1(\mathbf{r}) \leftarrow n(\mathbf{r}) d_1(\mathbf{r}), \quad d_2(\mathbf{r}) \leftarrow n(\mathbf{r}) d_2(\mathbf{r})$
$\hat{\mathbf{a}}(\mathbf{r}) \leftarrow [a_1(\mathbf{r}) - d_1(\mathbf{r}), a_3(\mathbf{r}) + d_1(\mathbf{r}), a_2(\mathbf{r}) - d_2(\mathbf{r}), a_4(\mathbf{r}) + d_2(\mathbf{r})]^t$

$(b_1(\mathbf{r}), c(\mathbf{r}), b_4(\mathbf{r})) \leftarrow \text{ProxHS}(b_1(\mathbf{r}), 0.5(b_2(\mathbf{r}) + b_3(\mathbf{r})), b_4(\mathbf{r}), \alpha_s, p)$
$d(\mathbf{r}) \leftarrow 0.5(b_2(\mathbf{r}) - b_3(\mathbf{r}))$
$\hat{\mathbf{b}}(\mathbf{r}) \leftarrow [b_1(\mathbf{r})\ \ c(\mathbf{r}) + d(\mathbf{r})\ \ c(\mathbf{r}) - d(\mathbf{r})\ \ b_4(\mathbf{r})]^t$

end

Return $\hat{\mathbf{a}}, \hat{\mathbf{b}}$

Algorithm 3 Proximal Operator for Hessian-Schatten Norm: $(\hat{a}, \hat{c}, \hat{b}) \leftarrow \text{ProxHS}(a, c, b, \alpha_s, p)$

if $p = 2$ **then**

$n \leftarrow \sqrt{a^2 + b^2 + 2c^2}$
$n \leftarrow max(n - \alpha_s, 0)/n$
$(\hat{a}, \hat{b}, \hat{c}) \leftarrow (na, nb, nc)$

end
else if $p = 1$ **then**

$l_1 \leftarrow (a + b) + \sqrt{(a - b)^2 + 4c}$
$l_2 \leftarrow (a + b) - \sqrt{(a - b)^2 + 4c}$
$\mathbf{v}_1 \leftarrow [a - l_1\ \ c]^t$
$\mathbf{v}_1 \leftarrow \mathbf{v}_1 / \|\mathbf{v}_1\|_2$
$\mathbf{v}_2 \leftarrow [a - l_2\ \ c]^t$
$\mathbf{v}_2 \leftarrow \mathbf{v}_2 / \|\mathbf{v}_2\|_2$
$l_1 \leftarrow sign(l_1) max(abs(l_1) - \alpha_s, 0)$
$l_2 \leftarrow sign(l_2) max(abs(l_2) - \alpha_s, 0)$
$\hat{a} \leftarrow v_{11}^2 l_1 + v_{21}^2 l_2$
$\hat{b} \leftarrow v_{12}^2 l_1 + v_{22}^2 l_2$

$\hat{c} \leftarrow v_{11} v_{12} l_1 + v_{21} v_{22} l_2$

end

sample locations lie in a subset of the grid points corresponding to the DFT of the image so as to facilitate efficient implementation. In this case, the cost $\|\mathcal{T} g - \mathbf{m}_f\|_2^2$ becomes identical to the cost of the equation (14) in the following way: (i) h is the inverse Fourier transform of the array having ones at sampling location of the operator $\mathcal{T}$ and zero at other locations; (ii) $W(\mathbf{r})$ is the image of ones; (ii) m is the inverse Fourier transform of the image obtained by embedding the samples in $\mathbf{m}_f$ in an array of zeros. Note that h and m will typically be complex.

TABLE 1. Comparison of PSNR scores yielded by various method at various number of iteration.

Image	Algorithm	500 Iter.	1500 Iter.	10000 Iter.
Image 1	GHS-2(G)	32.52	32.53	32.53
	GHS-2	32.64	32.66	32.66
	GHS-1	33.36	33.37	33.37
Image 2	GHS-2(G)	36.79	36.79	36.79
	GHS-2	36.90	36.92	36.92
	GHS-1	37.24	37.25	37.25
Image 3	GHS-2(G)	35.92	35.92	35.92
	GHS-2	35.98	35.98	35.98
	GHS-1	36.27	36.27	36.27
Image 4	GHS-2(G)	32.88	32.88	32.88
	GHS-2	32.89	32.92	32.92
	GHS-1	33.19	33.21	33.21
Image 5	GHS-2(G)	30.11	30.22	30.22
	GHS-2	30.14	30.24	30.24
	GHS-1	30.00	30.27	30.27
Image 6	GHS-2(G)	41.77	42.12	42.12
	GHS-2	41.87	42.15	42.15
	GHS-1	41.61	42.13	42.13

For the second model, random sampling is applied to the image resulting from blurring with a point-spread function. If $\mathbf{r}_i,\ i = 1, \ldots, N$ denote the sample locations and $m_i,\ i = 1, \ldots, N$ denote the samples, the quadratic data fitting cost for a given candidate image g can be written as $\sum_{i=1}^{N} ((h * g)(\mathbf{r}_i) - m_i)^2$, where h is the point-spread function of the microscope. This cost is not the negative log-likelihood for the noise encountered in microscopy, the mixed Poisson-Gaussian noise. However, this approximation of often used for ease of implementation. Now, in this case also, we assume that the spatial sample locations lie in a subset of the cartesian grid. In this case, the cost $\sum_{i=1}^{N} ((h * g)(\mathbf{r}_i) - m_i)^2$ becomes identical to the cost of the equation (14) in the following way: m represents the image obtained by embedding the measured spatial samples $m_i,\ i = 1, \ldots, N$ in an array of zeros, and $W(\mathbf{r})$ a binary image in which ones represent the points at which measurements were taken.

In all experiments, we will use the model images for determining the regularization parameters α_f and α_s. Needless to say, for the practical application of our method, we need to use a method for determining these parameters from the measured images that undergo reconstruction. However, the main focus of our manuscript is to demonstrate the advantage of the generality of our GHSN regularization and hence we defer this task for our future work.

A. EXPERIMENT 1

In this experiment, we consider the problem of reconstructing images from quasi-random Fourier samples. We use the images given in Figure 1 as the models. To obtain

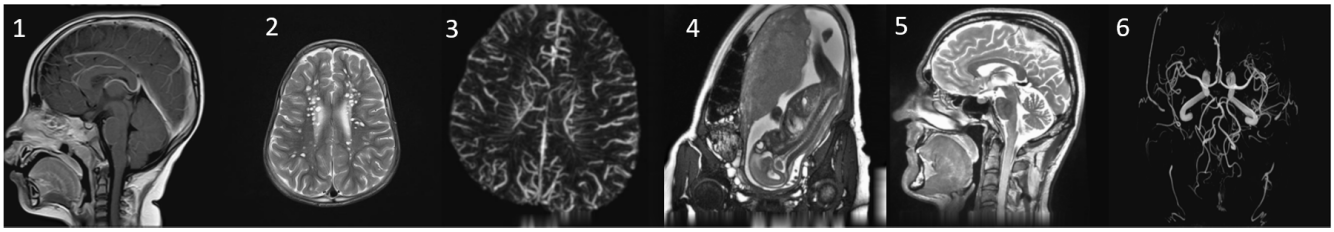

FIGURE 1. Images used in experiments.

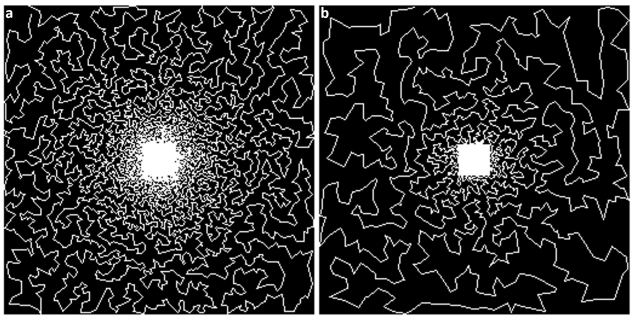

FIGURE 2. MRI Transfer Functions In Fourier domain a) TF1 (18%) b) TF2 (10%).

quasi-random Fourier samples, we use the trajectories generated by solving the travelling salesman problem as proposed by Chauffert *et al.* [57]. We use the sampling scheme given in Figure 2.a, which has a sampling density of 18% (TF1). We restrict p to be in $\{1, 2\}$. As done by Chauffert *et al.* [57], we quantize the sample locations generated from such trajectories by the grid corresponding to the DFT of the image. To demonstrate the relevance of our technical contributions, we consider three forms of reconstruction methods that differs from each other in terms of regularization and/or the minimization methods:

- GHS-1: Reconstruction using the Generalized Hessian Schatten norm with $p = 1$ and with proposed ADMM method for minimization
- GHS-2: Reconstruction using the Generalized Hessian Schatten norm with $p = 2$ and with proposed ADMM method for minimization
- GHS-2(G): Reconstruction using the Generalized Hessian Schatten norm with $p = 2$ with optimization method proposed by Guo *et al.* [53]

Note that, in GHS-2, the regularization is the same as the TGV-2 [49]. This means GHS-2 has novelty only in terms of optimization used, whereas GHS-1 has novelty both in terms of the regularization and the optimization method. GHS-2(G) entirely corresponds to the method of Guo *et al.* [53] with the part corresponding to the additional wavelet regularization removed. To evaluate these methods, The Fourier samples were corrupted by additive white Gaussian noise with variance 4. Table 1 compares the PSNR score of reconstruction obtained from all three methods with

TABLE 2. Comparison of reconstruction PSNR for experiment 2.

Name	TF	Noise	HS-2	HS-1	GHS-2	GHS-1	GHS-2(G)
Im 1	TF1	1	31.41	31.80	31.37	**31.88**	31.27
		2	30.78	31.10	30.76	**31.17**	30.66
	TF2	1	26.22	26.66	26.22	**26.68**	26.10
		2	26.00	26.36	26.00	**26.44**	25.88
Im 2	TF1	1	33.55	33.69	33.53	**33.70**	33.43
		2	32.44	32.57	32.41	**32.58**	32.31
	TF2	1	29.20	**29.37**	29.15	29.30	29.08
		2	28.72	28.83	28.68	**28.83**	28.60
Im 3	TF1	1	34.27	34.38	34.23	**34.38**	34.15
		2	33.63	33.72	33.59	**33.72**	33.52
	TF2	1	28.19	28.30	28.15	**28.34**	28.11
		2	28.04	**28.12**	27.96	28.10	27.92
Im 4	TF1	1	31.44	31.65	31.48	**31.74**	31.40
		2	30.82	30.97	30.87	**31.09**	30.79
	TF2	1	26.01	26.24	26.43	**26.47**	26.35
		2	25.91	26.07	26.21	**26.22**	26.13
Im 5	TF1	1	28.81	29.19	28.93	**29.23**	28.93
		2	28.43	28.78	28.41	**28.81**	28.34
	TF2	1	22.96	23.23	23.10	**23.25**	23.08
		2	22.85	23.12	22.96	**23.16**	22.92
Im 6	TF1	1	33.74	33.93	**34.22**	34.21	34.22
		2	32.50	32.68	**32.77**	32.76	32.77
	TF2	1	28.82	28.97	**29.31**	29.30	29.29
		2	28.47	28.60	28.63	**28.63**	28.62

500, 1500, and 10000 iterations. It is clear from the table that GHS-1 gives the best score with all three cases of the number of iterations for most cases of measured images, and GHS-2 comes next in PSNR. The PSNR score of GHS-2(G) is always the lowest. By considering the fact that GHS-2(G) differs from GHS-2 only by optimization, we conclude that the proposed ADMM method is more efficient than the optimization method proposed by Guo *et al.* [53]. We further note that, beyond 1500 iterations, there is no further improvement in the reconstructed image, and there is always a difference between the reconstruction obtained by GHS-2 and GHS-2(G). This again confirms that the proposed ADMM method is better conditioned numerically. Note that Table 1 confirms the following:

- GHS-1 attains higher PSNR than GHS-2 within a specified time duration. Since GHS-2 is identical to the regularization used in [53], the form of GHS-1 is the result of our generalization, this demonstrates the advantage of the generalization.
- GHS-2 attains higher PSNR than GHS-2(G) within a specified time duration. GHS-2(G) is identical to the published work [53], and GHS-2 differs from GHS-2(G)

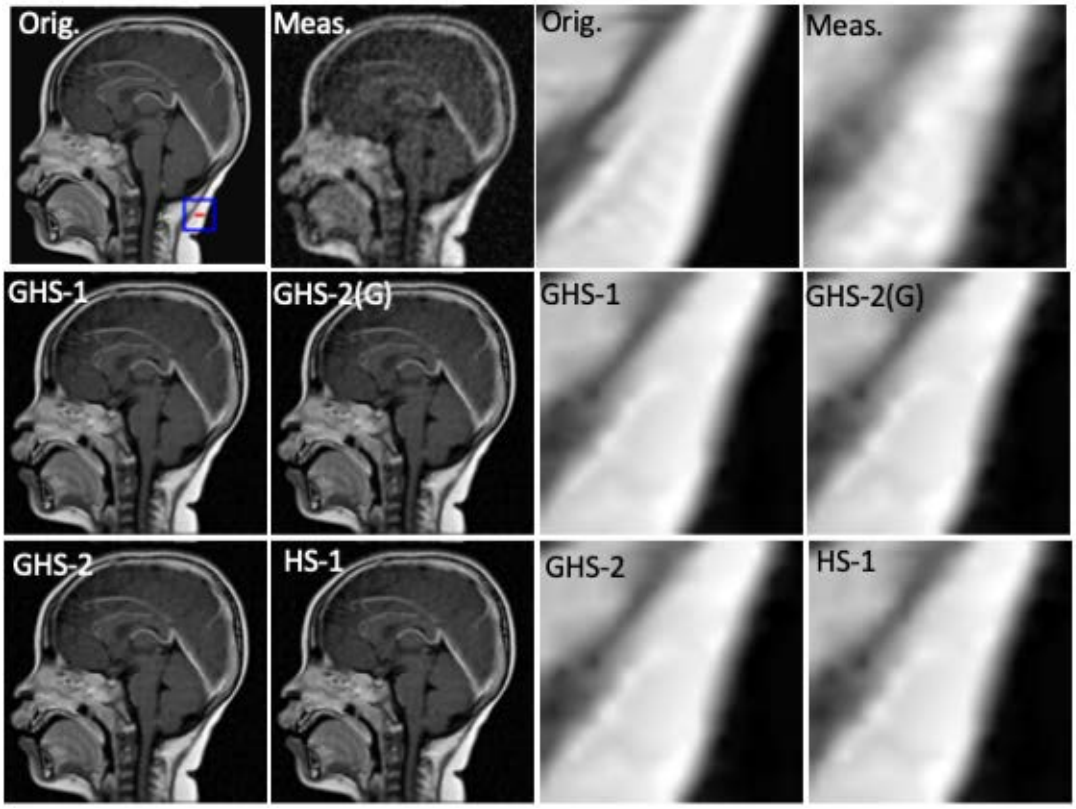

FIGURE 3. Comparison of restored images corresponding to image 1, TF1 and noise level 1. Images on the right are the zoomed-in portions from the images on the left. Location of the zoom-in portion is shown as box in the top left image.

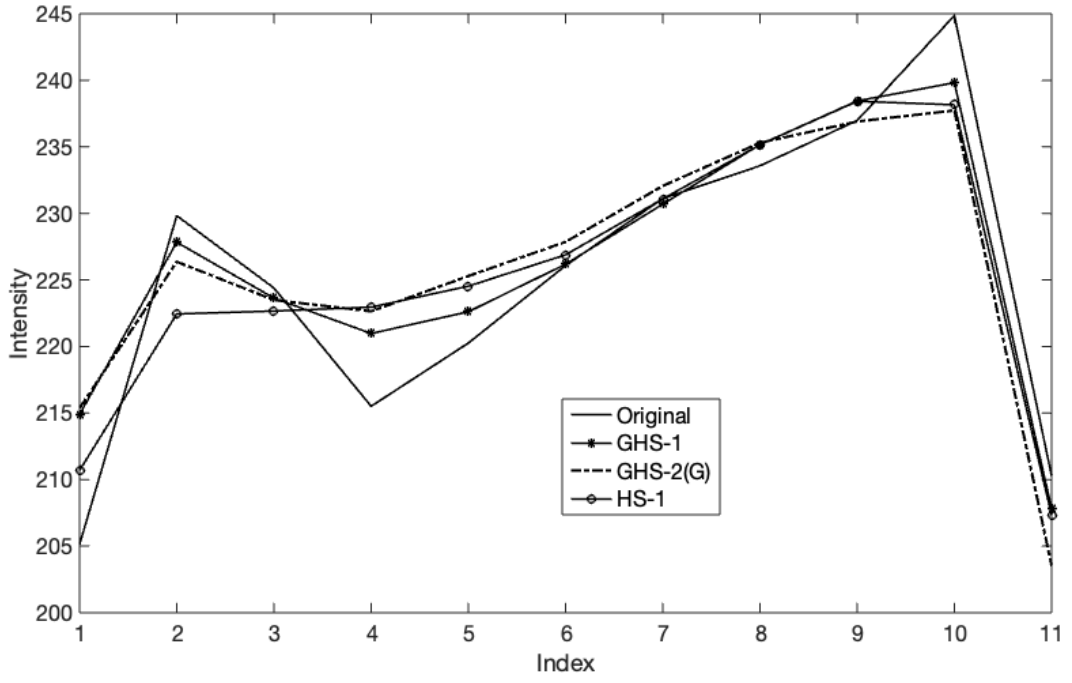

FIGURE 4. Scan lines from the results of Figure 3. The scan line location is shown in Figure 3.

only by the improved minimization method proposed by us. This demonstrates that the novel optimization method is faster.

For the first measurement set of Table 1, we also recorded the time taken by each of the methods to attain the PSNR of 32.52 dB. This PSNR is was attained in 9s, 21s,

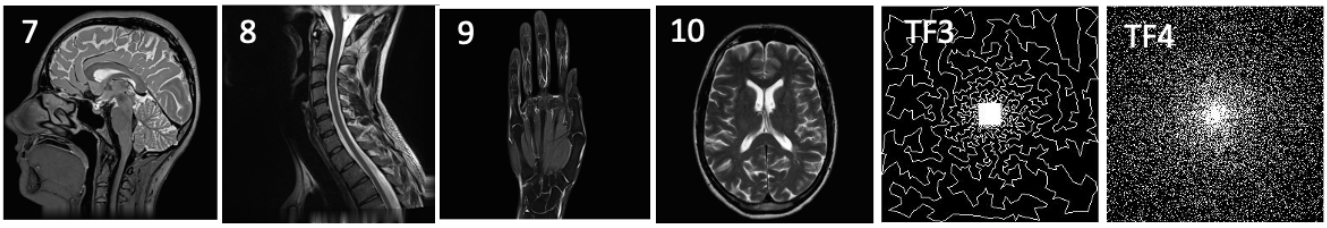

FIGURE 5. Additional images and MRI transfer functions for experiment 3.

and 32s by the methods GHS-1, GHS-2, and GHS-2(G) respectively.

B. EXPERIMENT 2

In the second experiment, we demonstrate the importance of novel regularization and the novel optimization under varied input settings for the same forward model considered in the previous experiment. To this end, we simulate measurements using both transfer functions given in the Figures 2.a and 2.b and add complex Gaussian noise on the Fourier samples with standard deviation values 5 and 7, which will be referred to as noise levels 1 and 2. This makes a total of 24 measurement sets. We evaluate all three methods listed in the previous experiment. We also evaluate with two additional methods: (i) HS-1: reconstruction using Hessian-Schatten 1-norm regularization with ADMM based minimization; (ii) HS-2: reconstruction using Hessian-Schatten 2-norm regularization with ADMM based minimization. Note that Hessian-Schatten 2-norm regularization is also the same as TV-2 regularization. We ran the algorithm for 2500 iterations for all the methods as this was found to be enough to obtain empirical residual convergence [54] for ADMM scheme. The results are displayed in Table 2. From the table, it is clear that GHS-1 is the best performing method. We also note that, among the three methods, GHS-1, GHS-2, GHS-2(G), we see the same pattern of relative performance as in the first experiment. Further, GHS-1 and GHS-2 are better than HS-1 and HS-2 in most cases. Moreover, while GHS-2 is consistently better than HS-2, GHS-2(G) is not always better than HS-2, although its regularization is the same as that of GHS-2. As GHS-2 differs from GHS-2(G) only by the optimization technique, this again confirms the importance of our novel optimization method. The images restored from the measurement simulated from image 1 using the transfer function TF1 with noise level 1 are displayed in Figure 3. We also display a zoomed-in region in each of the restored images. It is clear from the zoomed-in images that GHS-1 and GHS-2 schemes better recover the edges without any staircase effect. In Figure 4, we present scan lines from this image. As it is difficult to have clarity in the plot if several scan-lines are shown, we chose to show scan-line of reconstructions from three methods only, along with the scan-lines from the original image; we chose the best performing variant from the proposed method, GHS-1, and the closest competitors from the literature, GHS-2(G) and HS-1. The scan-lines confirms that GHS-1 follows the ground truth better than other methods.

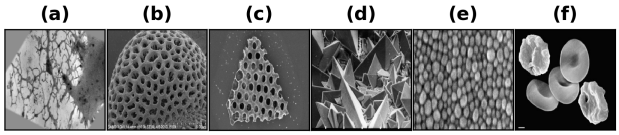

FIGURE 6. Scanning electron microscopic (SEM) model images.

C. EXPERIMENT 3

In the third experiment, we compare the performance of our algorithm against a method belonging to the class of plug and play methods. In this class of methods, a splitting based image reconstruction algorithm that is typically used for regularized reconstruction is modified by replacing the proximal operator corresponding to the regularization with a denoiser. For this experiment, we implemented a plug and play method with ADMM based splitting [58] and the well-known BM3D method as the denoiser. We denote this method by the abbreviation PnP-BM3D. We compare PnP-BM3D with two variants of the proposed method namely GHS-1 and GHS-2, which are defined in experiment I. We use all six images used in experiment I together with four additional images given in Figure 5. To simulate Fourier sampling, we use two transfer functions given in Figure 5 (labelled TF3 and TF4). TF3 covers 10% of Fourier samples and TF4 covers 20% of Fourier samples. We consider two levels of noise for each of the transfer functions. For TF4, we considered noise levels corresponding to standard deviations 0.025 and 0.0125, and for TF3, we considered noise levels corresponding to standard deviations 0.05 and 0.025. The results are presented in Table 3. The table confirms that GHS-1 algorithm produces a superior restoration in majority of the cases.

D. EXPERIMENT 4: RECONSTRUCTION FROM SIMULATED SEM SAMPLES

In this experiment, we consider the second forward model introduced at the beginning of Section V, namely blurring followed by random spatial sampling. Note that this forward model matches with the full generality of the data-fitting cost of the equation (14). For simulating the measurements, we use the images given in Figure 6 as model images ([59]–[62]).

TABLE 3. Comparison of proposed method with PnP-BM3D for reconstruction from Fourier samples.

Image	TF4, $\sigma = 0.025$			TF4, $\sigma = 0.05$			TF3, $\sigma = 0.0125$			TF3, $\sigma = 0.025$		
	GHS1	**GHS2**	**BM3D**	**GHS1**	**GHS2**	**BM3D**	**GHS1**	**GHS2**	**BM3D**	**GHS1**	**GHS2**	**BM3D**
Image 1	**0.886**	0.879	0.871	**0.875**	0.868	0.871	0.884	0.878	**0.891**	0.879	0.872	**0.889**
Image 2	**0.949**	0.947	0.891	**0.930**	0.928	0.889	**0.941**	0.939	0.938	0.929	0.926	**0.936**
Image 3	**0.890**	0.887	0.860	**0.873**	0.871	0.858	**0.888**	0.886	0.878	**0.883**	0.881	0.875
Image 4	0.842	0.840	**0.849**	0.826	0.822	**0.842**	**0.841**	0.840	0.836	**0.835**	0.832	0.834
Image 5	**0.919**	0.918	0.869	**0.902**	0.901	0.867	**0.916**	0.916	0.909	**0.909**	0.908	0.908
Image 6	**0.872**	0.872	0.822	**0.864**	0.863	0.821	**0.885**	0.885	0.867	0.881	**0.882**	0.867
Image 7	**0.965**	0.965	0.939	**0.945**	0.945	0.935	**0.963**	0.963	0.947	**0.950**	0.950	0.945
Image 8	**0.856**	0.845	0.816	**0.842**	0.833	0.814	**0.844**	0.836	0.844	0.842	0.831	**0.843**
Image 9	**0.952**	0.952	0.907	**0.928**	0.928	0.906	**0.947**	0.947	0.926	**0.932**	0.932	0.925
Image 10	**0.929**	0.928	0.892	**0.912**	0.909	0.892	**0.930**	0.928	0.927	0.920	0.919	**0.926**

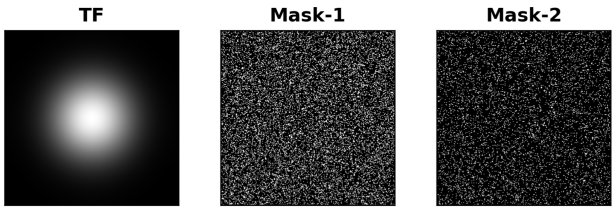

FIGURE 7. Blurring kernel and sampling mask used for SEM experiment.

TABLE 4. Comparison of proposed method with PnP-BM3D for reconstruction from simulated SEM samples.

Im	Mask	PnP-BM3D	GHS-2	GHS-1
1	1	0.810	0.829	**0.830**
	2	0.721	0.767	**0.768**
2	1	0.752	0.785	**0.785**
	2	0.660	**0.720**	0.719
3	1	0.838	0.857	**0.857**
	2	0.781	**0.808**	0.807
4	1	0.783	0.807	**0.808**
	2	0.685	0.728	**0.728**
5	1	0.826	0.885	**0.887**
	2	0.687	0.814	**0.819**
6	1	0.893	0.913	**0.913**
	2	0.835	0.882	**0.882**

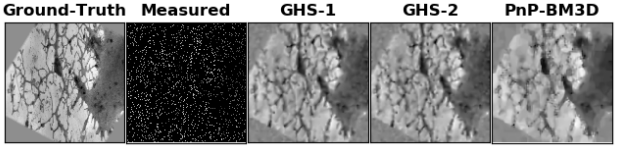

FIGURE 8. Comparison of reconstructions for model image (a) of Figure 6.

For blurring, we choose a Gaussian kernel with a standard deviation of 3; for spatial sampling, we used two random masks with uniform sampling densities of 10% and 20%. The Fourier modulus of this kernel and the masks are given in Figure 7. We also add 5% Gaussian noise. For comparison, we picked ADMM based plug-and-play method with BM3D denoiser (PnP-BM3D) as done in experiment 3. The plug and play method was found to be in need of a good initialization. Therefore, for a fair comparison, we obtained initialization as follows: each pixel in the initialization image was determined

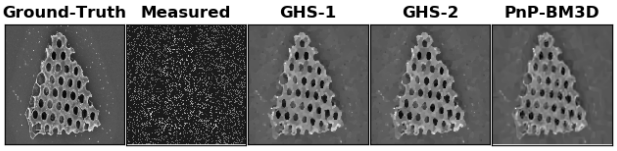

FIGURE 9. Comparison of reconstructions for model image (c) of Figure 6.

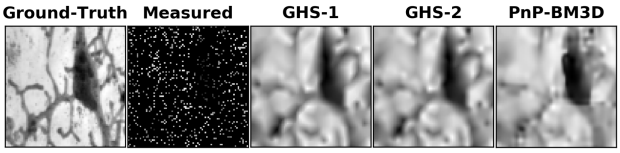

FIGURE 10. Zoom-in region from Figure 8.

by computing the median of the available samples within a 12×12 window. The results of the comparison are shown in Table 4. The table clearly shows that GHS-1 performs better than PnP-BM3D in all cases for this experiment. In Figure 8, we display the images reconstructed from samples obtained by applying the blurring and Mask-2 on model image (a). Further in Figure 9, we display the images reconstructed from samples obtained by applying the blurring and Mask-1 on model image (c). The displayed images clearly demonstrate the superiority of the proposed method. In particular, GHS-1 reconstructions are much sharper and more detailed than the reconstructions obtained by using PnP-BM3D method. In Figure 10 we have displayed zoomed-in regions from Figure 8, which again confirms the superiority of the proposed method.

VI. CONCLUSION

We proposed a new form of regularization named Generalized Hessian Schatten Norm (GHSN) regularization. GHSN generalizes all existing forms of second-order derivative-based regularization. We also developed a novel ADMM optimization method for image reconstruction using the GHSN. We demonstrated the advantage of generality in GHSN experimentally. We also demonstrated the effectiveness of the novel optimization method. In particular, even when parameters of GHSN is restricted such that it becomes the well-known form, called second-order total generalized (TGV-2)

variation, our optimization method yields superior reconstructions compared to the optimization method proposed for TGV-2 in the literature. In this work, our focus has been to demonstrate the advantage of adaptability arising from the generalization involved in GHSN. To make this advantage useful in a practical scenario, we need to develop a method to determine the parameters α_s and α_f from the measured data that undergoes reconstruction. This will be an important future research direction.

APPENDIX
PROOF OF PROPOSITIONS

Proof of Proposition 1: First we reproduce Eq. (9):

$$\mathcal{GHS}_p(g,\alpha_s,\alpha_f) = \max_{\substack{\|\overline{\mathbf{N}}\|_{\infty,S(q)}\leq\alpha_s \\ \|\overline{\mathbf{N}}*\tilde{d}\|_{\infty,2}\leq\alpha_f}} \langle\overline{\mathbf{N}}, \mathbf{H}*g\rangle. \tag{61}$$

Next, we denote the set $\{\mathbf{N}|\|\overline{\mathbf{N}}\|_{\infty,S(q)} \leq \alpha_s\}$ as $\mathcal{B}_S$ and, $\{\mathbf{z}|\|\mathbf{z}\|_{\infty,2} \leq \alpha_f\}$ as $\mathcal{B}_2$. This gives

$$\mathcal{GHS}_p(g,\alpha_s,\alpha_f) = \max_{(\mathbf{N},\overline{\mathbf{N}}*\tilde{d})\in\mathcal{B}_S\times\mathcal{B}_2} \langle\overline{\mathbf{N}}, \mathbf{H}*g\rangle. \tag{62}$$

Next, we note that $d_{xx} = d_x * d_x$, $d_y = d_y * d_y$, and $d_{xy} = d_x * d_y$. This means that we have $\mathbf{H}*g = g*\mathbf{d}*\mathbf{d}^t$. Substituting this gives

$$\mathcal{GHS}_p(g,\alpha_s,\alpha_f) = \max_{(\mathbf{N},\overline{\mathbf{N}}*\tilde{d})\in\mathcal{B}_S\times\mathcal{B}_2} \langle\overline{\mathbf{N}}, g*\mathbf{d}*\mathbf{d}^t\rangle. \tag{63}$$

By the property inner products with convolution, we can replace the operation $(\cdot)*\mathbf{d}^t$ applied on the second argument of the inner product by the adjoint operation $(\cdot)*\tilde{\mathbf{d}}$ applied on the second argument. This gives

$$\mathcal{GHS}_p(g,\alpha_s,\alpha_f) = \max_{(\mathbf{N},\overline{\mathbf{N}}*\tilde{d})\in\mathcal{B}_S\times\mathcal{B}_2} \langle\overline{\mathbf{N}}*\mathbf{d}, \mathbf{d}*g\rangle. \tag{64}$$

Now, we can replace the maximization by the minimization of the negated function and obtain

$$\mathcal{GHS}_p(g,\alpha_s,\alpha_f) = -\min_{(\mathbf{N},\overline{\mathbf{N}}*\tilde{d})\in\mathcal{B}_S\times\mathcal{B}_2} -\langle\overline{\mathbf{N}}*\tilde{\mathbf{d}}, \mathbf{d}*g\rangle. \tag{65}$$

The above minimization problem can be posed as constrained optimization problem as given below:

$$\mathcal{GHS}_p(g,\alpha_s,\alpha_f) = -\min_{(\mathbf{N},\mathbf{p})\in\mathcal{B}_S\times\mathcal{B}_2} -\langle\mathbf{p}, \mathbf{d}*g\rangle \quad \text{subject to } \overline{\mathbf{N}}*\tilde{\mathbf{d}} = \mathbf{p}. \tag{66}$$

Since the above problem is a convex optimization problem, $\mathcal{GHS}_p(g,\alpha_s,\alpha_f)$ can be determined using duality theory. To this end, we construct the Lagrange dual cost function of the minimization problem as given below:

$$q(\mathbf{u}) = \min_{(\mathbf{N},\mathbf{p})\in\mathcal{B}_S\times\mathcal{B}_2} -\langle\mathbf{p}, \mathbf{d}*g\rangle + \langle\mathbf{u}, \mathbf{p}-\overline{\mathbf{N}}*\tilde{\mathbf{d}}\rangle. \tag{67}$$

The above problem can be rearranged as given below by separating the terms for $\mathbf{p}$ and $\mathbf{N}$:

$$q(\mathbf{u}) = \min_{(\mathbf{N},\mathbf{p})\in\mathcal{B}_S\times\mathcal{B}_2} -\langle\mathbf{p}, \mathbf{d}*g-\mathbf{u}\rangle + \langle\mathbf{u}, -\overline{\mathbf{N}}*\tilde{\mathbf{d}}\rangle. \tag{68}$$

Next, we note that $\langle\mathbf{u}, -\overline{\mathbf{N}} * \tilde{\mathbf{d}}\rangle = \langle\mathbf{u} * \mathbf{d}^t, -\overline{\mathbf{N}}\rangle = \langle\overline{\mathbf{u}*\mathbf{d}^t}, -\mathbf{N}\rangle$. This gives

$$q(\mathbf{u}) = \min_{(\mathbf{N},\mathbf{p})\in\mathcal{B}_S\times\mathcal{B}_2} -\langle\mathbf{p}, \mathbf{d}*g-\mathbf{u}\rangle + \langle\overline{\mathbf{u}*\mathbf{d}^t}, -\mathbf{N}\rangle. \tag{69}$$

By duality theory, since strong duality holds ([63], Proposition 6.4.2), we have $-\mathcal{GHS}_p(g,\alpha_s,\alpha_f) = \max_{\mathbf{u}} q(\mathbf{u})$. Hence, the cost $\mathcal{GHS}_p(g,\alpha_s,\alpha_f)$ can be expressed as

$$-\mathcal{GHS}_p(g,\alpha_s,\alpha_f) = \max_{\mathbf{u}} \min_{(\mathbf{N},\mathbf{p})\in\mathcal{B}_S\times\mathcal{B}_2} -\langle\mathbf{p}, \mathbf{d}*g-\mathbf{u}\rangle + \langle\overline{\mathbf{u}*\mathbf{d}^t}, -\mathbf{N}\rangle. \tag{70}$$

Next, we rewrite the maximization with respect to $\mathbf{u}$ as the minimization:

$$\mathcal{GHS}_p(g,\alpha_s,\alpha_f) = \min_{\mathbf{u}} \max_{(\mathbf{N},\mathbf{p})\in\mathcal{B}_S\times\mathcal{B}_2} \langle\mathbf{p}, \mathbf{d}*g-\mathbf{u}\rangle + \langle\overline{\mathbf{u}*\mathbf{d}^t}, \mathbf{N}\rangle. \tag{71}$$

As the part of the cost function with respect to $\mathbf{N}$ and $\mathbf{p}$ are separable, we can rewrite the respective maximizations independently as given below:

$$\mathcal{GHS}_p(g,\alpha_s,\alpha_f) = \min_{\mathbf{u}} \max_{\mathbf{N}\in\mathcal{B}_S} \langle\overline{\mathbf{u}*\mathbf{d}^t}, \mathbf{N}\rangle + \max_{\mathbf{p}\in\mathcal{B}_2}\langle\mathbf{p}, \mathbf{d}*g-\mathbf{u}\rangle. \tag{72}$$

Now we apply the definition of conjugate norm. By noting the fact that the conjugate (dual) norm for the norm $\|\cdot\|_{\infty,S(q)}$ is $\|\cdot\|_{1,S(p)}$ with $\frac{1}{p}+\frac{1}{q}=1$ [64], and for $\|\cdot\|_{\infty,2}$ is $\|\cdot\|_{1,2}$. Substituting this gives

$$\mathcal{GHS}_p(g,\alpha_s,\alpha_f) = \min_{\mathbf{u}} \alpha_s\|\overline{\mathbf{u}*\mathbf{d}^t}\|_{1,S(p)} + \alpha_f\|\mathbf{d}*g-\mathbf{u}\|_{1,2},$$

which completes the proof.

Proof of Proposition 2: First we note that $\mathbf{A}_f\mathbf{A}_f^t = \mathbf{I}$, and let $\mathbf{B}_f$ be matrix such that the augmented matrix $\mathbf{M} = \begin{bmatrix}\mathbf{A}_f^t & \mathbf{B}_f^t\end{bmatrix}^t$ satisfies $\mathbf{M}^t\mathbf{M} = \mathbf{I}$. Further, let $\hat{\mathbf{z}} = \mathbf{M}\mathbf{z}$, and $\hat{\mathbf{a}} = \mathbf{M}\mathbf{a}$, and let $\hat{\mathbf{z}} = \begin{bmatrix}\hat{\mathbf{z}}_1 \\ \hat{\mathbf{z}}_2\end{bmatrix}$ and $\hat{\mathbf{a}} = \begin{bmatrix}\hat{\mathbf{a}}_1 \\ \hat{\mathbf{a}}_2\end{bmatrix}$, where $\hat{\mathbf{z}}_1$ and $\hat{\mathbf{z}}_2$ are sub-vectors of $\hat{\mathbf{z}}$ of size 2×1, and similarly $\hat{\mathbf{a}}_1$ and $\hat{\mathbf{a}}_2$ sub-vectors of $\hat{\mathbf{a}}$ of size 2×1. Then the form of cost function that is easy to minimize can be obtained by substituting $\mathbf{z} = \mathbf{M}^t\hat{\mathbf{z}}$, $\mathbf{a} = \mathbf{M}^t\hat{\mathbf{a}}$, and $\hat{\mathbf{z}}_1 = \mathbf{A}_f\mathbf{z}$ in $L_f(\mathbf{z},\mathbf{a},\alpha_f)$. By doing this, we obtain the transformed function as given below:

$$\hat{L}(\hat{\mathbf{z}}_1,\hat{\mathbf{z}}_2) = L_f(\mathbf{M}^t\hat{\mathbf{z}}, \mathbf{M}^t\hat{\mathbf{a}}, \alpha_f) = \frac{\beta}{2}\|\hat{\mathbf{z}}_1-\hat{\mathbf{a}}_1\|_2^2 + \alpha_f\|\hat{\mathbf{z}}_1\|_2 + \frac{\beta}{2}\|\hat{\mathbf{z}}_2-\hat{\mathbf{a}}_2\|_2^2. \tag{73}$$

Let $(\hat{\mathbf{z}}_1^*, \hat{\mathbf{z}}_2^*)$ denote the minimum of $\hat{L}(\cdot,\cdot)$. Clearly, $\hat{\mathbf{z}}_2^* = \hat{\mathbf{a}}_2$. Next, $\hat{\mathbf{z}}_1^*$ is the well-known proximal solution of l_2 norm [56], and it is given by $\hat{\mathbf{z}}_1^* = \frac{max(\|\hat{\mathbf{a}}_1\|_2-t,0)}{\|\hat{\mathbf{a}}_1\|_2}\hat{\mathbf{a}}_1$, where $t = \alpha_f/\beta$. From these, the minimum of $L_f(\mathbf{z},\mathbf{a},\alpha_f)$, denoted by $\mathbf{z}^*$ can be expressed as $\mathbf{z}^* = \mathbf{M}^t\begin{bmatrix}\hat{\mathbf{z}}_1^* \\ \hat{\mathbf{z}}_2^*\end{bmatrix} = \mathbf{M}^t\begin{bmatrix}\hat{\mathbf{z}}_1^* \\ \hat{\mathbf{a}}_2\end{bmatrix}$. By replacing $\hat{\mathbf{z}}_1^*$

by $\hat{\mathbf{a}}_1 - (\hat{\mathbf{a}}_1 - \hat{\mathbf{z}}_1^*)$, we get $\mathbf{z}^* = \mathbf{M}^t \begin{bmatrix} \hat{\mathbf{a}}_1 \\ \hat{\mathbf{a}}_2 \end{bmatrix} - \mathbf{M}^t \begin{bmatrix} \hat{\mathbf{a}}_1 - \hat{\mathbf{z}}_1 \\ \mathbf{0} \end{bmatrix} = \mathbf{M}^t \begin{bmatrix} \hat{\mathbf{a}}_1 \\ \hat{\mathbf{a}}_2 \end{bmatrix} - \mathbf{A}_f^t(\hat{\mathbf{a}}_1 - \hat{\mathbf{z}}_1^*)$. Next, from the expression for $\hat{\mathbf{z}}_1^*$, we can deduce the following on the difference $(\hat{\mathbf{a}}_1 - \hat{\mathbf{z}}_1^*)$:

$$\begin{aligned}(\hat{\mathbf{a}}_1 - \hat{\mathbf{z}}_1^*) &= \hat{\mathbf{a}}_1 - \frac{max(\|\hat{\mathbf{a}}_1\|_2 - t, 0)}{\|\hat{\mathbf{a}}_1\|_2}\hat{\mathbf{a}}_1 \\ &= \frac{1}{\|\hat{\mathbf{a}}_1\|}\hat{\mathbf{a}}_1(\|\hat{\mathbf{a}}_1\|_2 - max(\|\hat{\mathbf{a}}_1\|_2 - t, 0)) \\ &= \frac{1}{\|\hat{\mathbf{a}}_1\|}\hat{\mathbf{a}}_1 min(\|\hat{\mathbf{a}}_1\|_2, t)\end{aligned}$$

By using the above relation and by using the fact that $\mathbf{M}^t \begin{bmatrix} \hat{\mathbf{a}}_1 \\ \hat{\mathbf{a}}_2 \end{bmatrix} = \mathbf{a}$, we get

$$\mathbf{z}^* = \mathbf{a} - \frac{1}{\|\hat{\mathbf{a}}_1\|} min(\|\hat{\mathbf{a}}_1\|_2, t)\mathbf{A}_f^t \hat{\mathbf{a}}_1.$$

Substituting $\hat{\mathbf{a}}_1 = \mathbf{A}_f \mathbf{a}$ in the above expressing gives the final expression.

Proof of Proposition 3: From the definitions of $\mathbf{K}$ and $\mathbf{L}$, we first note that they can be expressed as

$$\mathbf{K} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0.5 & 0.5 & 0 \\ 0 & 0.5 & 0.5 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad \mathbf{L} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0.5 & -0.5 & 0 \\ 0 & -0.5 & 0.5 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \tag{74}$$

From the form given above, we observe that $\mathbf{K}^t = \mathbf{K}, \mathbf{L} = \mathbf{L}^t$, and $\mathbf{LK} = \mathbf{KL} = \mathbf{0}$. Hence, $\frac{1}{2}\|\mathbf{z} - \mathbf{a}\|_2^2$ can be written as $\frac{1}{2}\|\mathbf{z} - \mathbf{a}\|_2^2 = \frac{1}{2}\|\mathbf{Kz} - \mathbf{Ka}\|_2^2 + \frac{1}{2}\|\mathbf{Lz} - \mathbf{La}\|_2^2$. As a result, the minimization problem becomes,

$$\mathbf{z}^* = \underset{\mathbf{z}\in\mathbb{R}^4}{argmin} \ \frac{1}{2}\|\mathbf{Kz} - \mathbf{Ka}\|_2^2 + \frac{1}{2}\|\mathbf{Lz} - \mathbf{La}\|_2^2 + t\|\mathbf{Kz}\|_{S(p)}.$$

As an additional property, we also observe that $\mathbf{K} + \mathbf{L} = \mathbf{I}$. Hence, we have $\mathcal{R}(\mathbf{K}) \oplus \mathcal{R}(\mathbf{L}) = \mathbb{R}^4$. This means that the minimum of $L_s(\mathbf{z}, \mathbf{a}, \alpha_s)$, denoted by $\mathbf{z}^*$, can be written as $\mathbf{z}^* = \mathbf{z}_k^* + \mathbf{z}_l^*$, where

$$\begin{aligned}(\mathbf{z}_k^*, \mathbf{z}_l^*) = \underset{\mathbf{z}_k\in\mathcal{R}(\mathbf{K}),\ \mathbf{z}_l\in\mathcal{R}(\mathbf{L})}{argmin} &\frac{1}{2}\|\mathbf{K}(\mathbf{z}_k + \mathbf{z}_l) - \mathbf{K}a\|_2^2 \\ &+ \frac{1}{2}\|\mathbf{L}(\mathbf{z}_k + \mathbf{z}_l) - \mathbf{L}a\|_2^2 + t\|\mathbf{K}(\mathbf{z}_k + \mathbf{z}_l)\|_{S(p)}.\end{aligned}$$

Next, because of the relation $\mathbf{LK} = \mathbf{KL} = \mathbf{0}$, we have $\mathbf{Kz}_l = \mathbf{Lz}_k = \mathbf{0}$. This means that the minimization subproblem can be separated as

$$\mathbf{z}_k^* = \underset{\mathbf{z}_k\in\mathcal{R}(\mathbf{K})}{argmin}\frac{1}{2}\|\mathbf{Kz}_k - \mathbf{Ka}\|_2^2 + t\|\mathbf{Kz}_k\|_{S(p)} \tag{75}$$

$$\mathbf{z}_l^* = \underset{\mathbf{z}_l\in\mathcal{R}(\mathbf{L})}{argmin}\frac{1}{2}\|\mathbf{Lz}_l - \mathbf{La}\|_2^2 \tag{76}$$

Next, we observe that $\mathbf{K}^2 = \mathbf{K}$ and $\mathbf{L}^2 = \mathbf{L}$, this means $\mathbf{K}$ and $\mathbf{L}$ are orthogonal projections on range spaces of $\mathbf{K}$ and $\mathbf{L}$ respectively, which implies that $\mathbf{Kz}_k = \mathbf{z}_k$ and $\mathbf{Lz}_l = \mathbf{z}_l$. Hence the above minimization problems can be written as

$$\mathbf{z}_k^* = \underset{\mathbf{z}_k\in\mathcal{R}(\mathbf{K})}{argmin}\frac{1}{2}\|\mathbf{z}_k - \mathbf{Ka}\|_2^2 + t\|\mathbf{z}_k\|_{S(p)} \tag{77}$$

$$\mathbf{z}_l^* = \underset{\mathbf{z}_l\in\mathcal{R}(\mathbf{L})}{argmin}\frac{1}{2}\|\mathbf{z}_l - \mathbf{La}\|_2^2 \tag{78}$$

From the above forms of minimization sub-problems, it is clear that $\mathbf{z}_l^* = \mathbf{La}$. We claim that $\mathbf{z}_k^* = \mathcal{P}_s(\mathbf{Ka}, t)$ where $\mathcal{P}_s(\mathbf{Ka}, t)$ is the proximal of the Schatten-Norm applied on $\mathbf{Ka}$ with threshold t. This is because $\mathcal{P}_s(\mathbf{Ka}, t)$ is the global unconstrained minimizer of the cost $\frac{1}{2}\|z_k - \mathbf{K}a\|_2^2 + t\|\mathbf{z}_k\|_{S(p)}$, also by definition of $\mathcal{P}_s(\mathbf{Ka}, t)$ [46] (section III.E), it can be seen that $\mathcal{P}_s(\mathbf{Ka}, t) \in \mathcal{R}(\mathbf{K})$. From the above two statements it can be concluded that $\mathcal{P}_s(\mathbf{Ka}, t)$ is the required minimizer. Hence the required solution becomes $\mathbf{z}^* = \mathbf{La} + \mathcal{P}_s(\mathbf{Ka}, t)$.

REFERENCES

[1] M. R. Banham and A. K. Katsaggelos, "Digital image restoration," *IEEE Signal Process. Mag.*, vol. 14, no. 2, pp. 24–41, Mar. 1997.
[2] R. Deriche, P. Kornprobst, and M. Nikolova, "Half-quadratic regularization for MRI image restoration," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, vol. 6, Apr. 2003, p. 585.
[3] S. Ramani, Z. Liu, J. Rosen, J.-F. Nielsen, and J. A. Fessler, "Regularization parameter selection for nonlinear iterative image restoration and MRI reconstruction using GCV and sure-based methods," *IEEE Trans. Image Process.*, vol. 21, no. 8, pp. 3659–3672, Aug. 2012.
[4] S. Rathee, Z. J. Koles, and T. R. Overton, "Image restoration in computed tomography: Estimation of the spatially variant point spread function," *IEEE Trans. Med. Imag.*, vol. 11, no. 4, pp. 539–545, Dec. 1992.
[5] J. Ma, J. Huang, Q. Feng, H. Zhang, H. Lu, Z. Liang, and W. Chen, "Low-dose computed tomography image restoration using previous normal-dose scan," *Med. Phys.*, vol. 38, no. 10, pp. 5713–5731, 2011.
[6] N. Dey, L. Blanc-Féraud, C. Zimmer, Z. Kam, J.-C. Olivo-Marin, and J. Zerubia, "A deconvolution method for confocal microscopy with total variation regularization," in *Proc. 2nd IEEE Int. Symp. Biomed. Imag., Macro Nano*, Apr. 2004, pp. 1223–1226.
[7] M. Arigovindan, J. C. Fung, D. Elnatan, V. Mennella, Y.-H. M. Chan, M. Pollard, E. Branlund, J. W. Sedat, and D. A. Agard, "High-resolution restoration of 3D structures from widefield images with extreme low signal-to-noise-ratio," *Proc. Nat. Acad. Sci. USA*, vol. 110, no. 43, pp. 17344–17349, Oct. 2013.
[8] P. C. Hansen, *Discrete Inverse Problems: Insight and Algorithms*. Philadelphia, PA, USA: SIAM, 2010.
[9] M. G. Kang and A. K. Katsaggelos, "General choice of the regularization functional in regularized image restoration," *IEEE Trans. Image Process.*, vol. 4, no. 5, pp. 594–602, May 1995.
[10] J. Ker, L. Wang, J. Rao, and T. Lim, "Deep learning applications in medical image analysis," *IEEE Access*, vol. 6, pp. 9375–9389, 2017.
[11] G. Yang, S. Yu, H. Dong, G. Slabaugh, P. L. Dragotti, X. Ye, F. Liu, S. Arridge, J. Keegan, Y. Guo, and D. Firmin, "DAGAN: Deep de-aliasing generative adversarial networks for fast compressed sensing MRI reconstruction," *IEEE Trans. Med. Imag.*, vol. 37, no. 6, pp. 1310–1321, Jun. 2018.
[12] J. Schlemper, J. Caballero, J. V. Hajnal, A. N. Price, and D. Rueckert, "A deep cascade of convolutional neural networks for dynamic mr image reconstruction," *IEEE Trans. Med. Imag.*, vol. 37, no. 2, pp. 491–503, Feb. 2017.
[13] K. H. Jin, M. T. McCann, E. Froustey, and M. Unser, "Deep convolutional neural network for inverse problems in imaging," *IEEE Trans. Image Process.*, vol. 26, no. 9, pp. 4509–4522, Sep. 2017.
[14] L. Xu, J. S. Ren, C. Liu, and J. Jia, "Deep convolutional neural network for image deconvolution," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 27, 2014, pp. 1790–1798.

[15] R. Ahmad, C. A. Bouman, G. T. Buzzard, S. Chan, S. Liu, E. T. Reehorst, and P. Schniter, "Plug-and-play methods for magnetic resonance imaging: Using denoisers for image recovery," *IEEE Signal Process. Mag.*, vol. 37, no. 1, pp. 105–116, Jan. 2020.

[16] E. Ryu, J. Liu, S. Wang, X. Chen, Z. Wang, and W. Yin, "Plug-and-play methods provably converge with properly trained denoisers," in *Proc. Int. Conf. Mach. Learn.*, vol. 97, K. Chaudhuri and R. Salakhutdinov, Eds. Long Beach, CA, USA: PMLR, Jun. 2019, pp. 5546–5557. [Online]. Available: http://proceedings.mlr.press/v97/ryu19a.html

[17] V. Antun, F. Renna, C. Poon, B. Adcock, and A. C. Hansen, "On instabilities of deep learning in image reconstruction and the potential costs of AI," *Proc. Nat. Acad. Sci. USA*, vol. 117, no. 48, pp. 30088–30095, Dec. 2020. [Online]. Available: https://www.pnas.org/content/early/2020/05/08/1907377117

[18] A. N. Tikhonov and V. Y. Arsenin, "Solutions of ill-posed problems," *SIAM Rev.*, vol. 21, no. 2, pp. 266–267.

[19] L. Ying, D. Xu, and Z.-P. Liang, "On Tikhonov regularization for image reconstruction in parallel MRI," in *Proc. 26th Annu. Int. Conf. IEEE Eng. Med. Biol. Soc.*, vol. 1, Sep. 2004, pp. 1056–1059.

[20] S. Ramani, C. Vonesch, and M. Unser, "Deconvolution of 3D fluorescence micrographs with automatic risk minimization," in *Proc. 5th IEEE Int. Symp. Biomed. Imag., From Nano Macro*, May 2008, pp. 732–735.

[21] A. Chambolle, R. A. De Vore, N.-Y. Lee, and B. J. Lucier, "Nonlinear wavelet image processing: Variational problems, compression, and noise removal through wavelet shrinkage," *IEEE Trans. Image Process.*, vol. 7, no. 3, pp. 319–335, Mar. 1998.

[22] M. A. T. Figueiredo and R. D. Nowak, "An EM algorithm for wavelet-based image restoration," *IEEE Trans. Image Process.*, vol. 12, no. 8, pp. 906–916, Aug. 2003.

[23] I. Daubechies, M. Defrise, and C. De Mol, "An iterative thresholding algorithm for linear inverse problems with a sparsity constraint," *Commun. Pure Appl. Math.*, vol. 57, no. 11, pp. 1413–1457, Nov. 2004.

[24] A. Beck and M. Teboulle, "A fast iterative shrinkage-thresholding algorithm for linear inverse problems," *SIAM J. Imag. Sci.*, vol. 2, no. 1, pp. 183–202, 2009.

[25] J. M. Bioucas-Dias and M. A. T. Figueiredo, "A new TwIST: Two-step iterative shrinkage/thresholding algorithms for image restoration," *IEEE Trans. Image Process.*, vol. 16, no. 12, pp. 2992–3004, Dec. 2007.

[26] S. Ravishankar and Y. Bresler, "MR image reconstruction from highly undersampled k-space data by dictionary learning," *IEEE Trans. Med. Imag.*, vol. 30, no. 5, pp. 1028–1041, May 2010.

[27] L. I. Rudin, S. Osher, and E. Fatemi, "Nonlinear total variation based noise removal algorithms," *Phys. D, Nonlinear Phenomena*, vol. 60, nos. 1–4, pp. 259–268, 1992.

[28] J. A. Fessler, "Optimization methods for MR image reconstruction (long version)," 2019, *arXiv:1903.03510*.

[29] C. Poon, "On the role of total variation in compressed sensing," *SIAM J. Imag. Sci.*, vol. 8, no. 1, pp. 682–720, 2015.

[30] D. Needell and R. Ward, "Stable image reconstruction using total variation minimization," *SIAM J. Imag. Sci.*, vol. 6, no. 2, pp. 1035–1058, Mar. 2013.

[31] A. Chambolle, V. Caselles, D. Cremers, M. Novaga, and T. Pock, "An introduction to total variation for image analysis," *Theor. Found. Numer. Methods Sparse Recovery*, vol. 9, nos. 263–340, p. 227, 2010.

[32] T. F. Chan and C.-K. Wong, "Total variation blind deconvolution," *IEEE Trans. Image Process.*, vol. 7, no. 3, pp. 370–375, Mar. 1998.

[33] I. Yanovsky, B. H. Lambrigtsen, A. B. Tanner, and L. A. Vese, "Efficient deconvolution and super-resolution methods in microwave imagery," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 8, no. 9, pp. 4273–4283, Sep. 2015.

[34] G. Tang and J. Ma, "Application of total-variation-based curvelet shrinkage for three-dimensional seismic data denoising," *IEEE Geosci. Remote Sens. Lett.*, vol. 8, no. 1, pp. 103–107, Jan. 2011.

[35] S. Durand and J. Froment, "Reconstruction of wavelet coefficients using total variation minimization," *SIAM J. Sci. Comput.*, vol. 24, no. 5, pp. 1754–1767, Jan. 2003.

[36] T. F. Chan, J. Shen, and H.-M. Zhou, "Total variation wavelet inpainting," *J. Math. Imag. Vis.*, vol. 25, no. 1, pp. 107–125, Jul. 2006, doi: 10.1007/s10851-006-5257-3.

[37] T. Chan, A. Marquina, and P. Mulet, "High-order total variation-based image restoration," *SIAM J. Sci. Comput.*, vol. 22, no. 2, pp. 503–516, 2000.

[38] Z. Dogan, S. Lefkimmiatis, A. Bourquard, and M. Unser, "A second-order extension of TV regularization for image deblurring," in *Proc. 18th IEEE Int. Conf. Image Process.*, Sep. 2011, pp. 705–708.

[39] O. Scherzer, "Denoising with higher order derivatives of bounded variation and an application to parameter estimation," *Computing*, vol. 60, no. 1, pp. 1–27, Mar. 1998.

[40] G. Steidl, "A note on the dual treatment of higher-order regularization functionals," *Computing*, vol. 76, nos. 1–2, pp. 135–148, Jan. 2006.

[41] N. M. Hue, D. N. H. Thanh, L. T. Thanh, N. N. Hien, and V. B. S. Prasath, "Image denoising with overlapping group sparsity and second order total variation regularization," in *Proc. 6th NAFOSTED Conf. Inf. Comput. Sci. (NICS)*, Dec. 2019, pp. 370–374.

[42] D. N. H. Thanh, V. B. S. Prasath, and S. Dvoenko, "An adaptive image inpainting method based on Euler's elastica with adaptive parameters estimation and the discrete gradient method," *Signal Process.*, vol. 178, Jan. 2021, Art. no. 107797.

[43] D. N. H. Thanh, V. B. S. Prasath, L. M. Hieu, and S. Dvoenko, "An adaptive method for image restoration based on high-order total variation and inverse gradient," *Signal, Image Video Process.*, vol. 14, no. 6, pp. 1189–1197, Sep. 2020.

[44] S. Lefkimmiatis, J. P. Ward, and M. Unser, "Hessian Schatten-norm regularization for linear inverse problems," *IEEE Trans. Image Process.*, vol. 22, no. 5, pp. 1873–1888, May 2013.

[45] S. Lefkimmiatis and M. Unser, "3D Poisson microscopy deconvolution with Hessian Schatten-norm regularization," in *Proc. IEEE 10th Int. Symp. Biomed. Imag.*, Apr. 2013, pp. 161–164.

[46] S. Lefkimmiatis and M. Unser, "Poisson image reconstruction with Hessian Schatten-norm regularization," *IEEE Trans. Image Process.*, vol. 22, no. 11, pp. 4314–4327, Nov. 2013.

[47] L. Liu, X. Li, K. Xiang, J. Wang, and S. Tan, "Low-dose CBCT reconstruction using Hessian Schatten penalties," *IEEE Trans. Med. Imag.*, vol. 36, no. 12, pp. 2588–2599, Dec. 2017.

[48] M. Ghulyani and M. Arigovindan, "Fast roughness minimizing image restoration under mixed Poisson–Gaussian noise," *IEEE Trans. Image Process.*, vol. 30, pp. 134–149, 2021.

[49] K. Bredies, K. Kunisch, and T. Pock, "Total generalized variation," *SIAM J. Imag. Sci.*, vol. 3, no. 3, pp. 492–526, 2010.

[50] F. Knoll, K. Bredies, T. Pock, and R. Stollberger, "Second order total generalized variation (TGV) for MRI," *Magn. Reson. Med.*, vol. 65, no. 2, pp. 480–491, Feb. 2011.

[51] Y. Jiao, Q. Jin, X. Lu, and W. Wang, "Alternating direction method of multipliers for linear inverse problems," *SIAM J. Numer. Anal.*, vol. 54, no. 4, pp. 2114–2137, Jan. 2016.

[52] W. Lu, J. Duan, Z. Qiu, Z. Pan, R. W. Liu, and L. Bai, "Implementation of high-order variational models made easy for image processing," *Math. Methods Appl. Sci.*, vol. 39, no. 14, pp. 4208–4233, 2016.

[53] W. Guo, J. Qin, and W. Yin, "A new detail-preserving regularization scheme," *SIAM J. Imag. Sci.*, vol. 7, no. 2, pp. 1309–1334, 2014.

[54] S. Boyd, N. Parikh, and E. Chu, *Distributed Optimization and Statistical Learning via the Alternating Direction Method of Multipliers*. Norwell, MA, USA: Now Publishers, 2011.

[55] S. Boyd, S. P. Boyd, and L. Vandenberghe, *Convex Optimization*. Cambridge, U.K.: Cambridge Univ. Press, 2004.

[56] N. Parikh and S. Boyd, "Proximal algorithms," *Found. Trends Optim.*, vol. 1, no. 3, pp. 127–239, Jan. 2014.

[57] N. Chauffert, P. Ciuciu, J. Kahn, and P. Weiss, "Variable density sampling with continuous trajectories," *SIAM J. Imag. Sci.*, vol. 7, no. 4, pp. 1962–1992, Oct. 2014.

[58] S. H. Chan, X. Wang, and O. A. Elgendy, "Plug-and-play ADMM for image restoration: Fixed-point convergence and applications," *IEEE Trans. Comput. Imag.*, vol. 3, no. 1, pp. 84–98, Jan. 2017.

[59] *Some Spectacular SEM Images of the Microscopic World*, IFL Sci., 2019. [Online]. Available: https://www.iflscience.com/technology/some-spectacular-sem-images-microscopic-world/

[60] C. A. Moxon, G. E. Grau, and A. G. Craig, "Malaria: Modification of the red blood cell and consequences in the human host," *Brit. J. Haematol.*, vol. 154, no. 6, pp. 670–679, Sep. 2011.

[61] M. Benkovičová, K. Végsö, P. Šiffalovic, M. Jergel, E. Majková, S. Luby, and A. Šatka, "Preparation of sterically stabilized gold nanoparticles for plasmonic applications," *Chem. Papers*, vol. 67, no. 9, pp. 1225–1230, 2013.
[62] *3D-SEM Dataset*, Kaggle, 2019. [Online]. Available: https://www.kaggle.com/kmader/3dsem-a-dataset-for-3d-sem-surface-reconstruction
[63] D. Bertsekas, A. Nedic, and A. Ozdaglar, *Convex Analysis and Optimization* (Athena Scientific Optimization and Computation Series). Belmont, MA, USA: Athena Scientific, 2003.
[64] R. A. Horn and C. R. Johnson, *Matrix Analysis*. Cambridge, U.K.: Cambridge Univ. Press, 2012.

MANU GHULYANI was born in Jaipur, Rajasthan, India, in 1990. He received the B.E. degree (Hons.) in electronics and instrumentation from the Birla Institute of Technology and Science, Pilani, in 2012, and the M.Sc. (Engg.) degree in electrical engineering from the Indian Institute of Science, Bengaluru, in 2018, where he is currently pursuing the Ph.D. degree with the Department of Electrical Engineering.

From 2012 to 2015, he was an Operations Engineer at NTPC Ltd. His research interests include optimization, inverse problems, machine learning, and signal processing.

DEEPAK G. SKARIAH was born in Kerala, India. He received the B.Tech. degree from the University of Kerala, in 2011, and the M.Sc. (Engg.) degree from the Indian Institute of Science, Bengaluru, India, in 2015, where he is currently pursuing the Ph.D. degree. His research interests include convex optimization, signal processing, and machine learning.

MUTHUVEL ARIGOVINDAN received the B.Tech. degree from the Pondicherry Engineering College, in 1995, the M.Sc.Eng. degree from the Indian Institute of Science, Bengaluru, India, in 1999, and the Ph.D. degree from the Swiss Federal Institute of Technology, Lausanne, in 2005. He then received the Postdoctoral Training from the Department of Biochemistry and Biophysics, University of California at San Francisco. He is currently an Assistant Professor with the Department of Electrical Engineering, Indian Institute of Science.