

Improved Sparse Signal Recovery via Adaptive Correlated Noise Model

Nasser Eslahi  and Alessandro Foi 

Abstract—Sparse signal recovery consists of employing a sparsity promoting regularizer to estimate the underlying signal from an incomplete set of measurements. Typical recovery approaches involve an alternating procedure where the estimate of the signal is progressively refined through filtering its degraded observation by a denoiser. The filter acts, implicitly, as a regularizer for the estimate. Hence, the implicit regularization is determined by the signal model underlying the denoising filter, as well as by the model of effective noise (i.e. degradation to be filtered) adopted by the filter. We improve the recovery by an adaptive stationary correlated noise model and the corresponding denoiser in place of the traditional filters for uncorrelated white noise. The effective noise can vary as the recovery progresses and we track these variations by estimating the noise correlation at every iteration. Competitive inverse problems are considered as benchmarks, including compressive spectral/temporal imaging and 2D/3D tomography. Analysis of the effective noise within each application demonstrates that it features various forms of correlation, which if leveraged by a denoiser lead to a better and faster signal recovery.

Index Terms—Sparse recovery, stationary correlated noise, power spectral density, noise estimation, collaborative filtering.

I. INTRODUCTION

SPARSE signal recovery as one of the fundamental problems in computational imaging aims to recover an unknown signal vector \mathbf{x} from few non-adaptive, possibly noisy, linear measurements \mathbf{y} of the form

$$\mathbf{y} = \mathbf{M}\mathbf{x} + \boldsymbol{\epsilon}, \quad (1)$$

where $\mathbf{M} \in \mathbb{C}^{m \times n}$ is a measurement matrix with $m \ll n$, and $\boldsymbol{\epsilon}$ represents the measurement error [1], [2]. The underlying signal \mathbf{x} can be recovered using a nonlinear sparsity-promoting algorithm, under the assumption that \mathbf{x} is sparse or compressible with respect to a given basis or a redundant dictionary mutually incoherent with \mathbf{M} [3].

Typical sparse signal recovery approaches proceed iteratively, refining the estimate of \mathbf{x} by filtering its degraded observation [4]–[24]. These approaches are successfully used in various applications [25]–[29]. As we detail in Section II, a common feature of these approaches is to utilize an additive white Gaussian noise (AWGN) shrinkage or denoising to alleviate the degradations. In particular, utilizing an AWGN denoiser corresponds to modeling the noise to be filtered as AWGN; however, this filtered noise model can be different from that of the actual degradation (or *effective noise*), i.e. the difference between the degraded observation and the ground truth. The AWGN assumption of effective noise holds indeed

only under special conditions that are hardly met in practice, e.g., the measurement matrix \mathbf{M} being itself independent and identically distributed random Gaussian as required in approximate message passing [11], [16]. Nevertheless, AWGN denoisers have been legitimately or pragmatically applied even when the effective noise is not white, e.g., in order to guarantee convergence to a given penalized solution [17]–[21], or simply because of limitations of the denoiser itself.

A more general noise model, which we advocate in this paper¹, allows for *correlation* within the degradations. This correlation can be result of multiple contributors: the structure of \mathbf{M} , the statistics of $\boldsymbol{\epsilon}$, as well as their interaction with the structure of \mathbf{x} and the effect of denoisers during previous iterations. In contrast to AWGN, correlated noise corresponds to errors that are disproportionate across the data spectrum, to an extent that AWGN denoisers may not effectively discern between the true signal and noise in regularization via shrinkage. Hence, ignoring such correlation in the denoising step can lead to ineffective filtering and also distortion to the underlying signal, thus impairing the accurate high-quality recovery of \mathbf{x} . Moreover, we have the denoising tools that allow to deal with correlated noise efficiently.

This work develops around the following five contributions:

1. (*Section III*) We treat the degradations in sparse signal recovery as *additive* (signal-independent) *stationary correlated noise* that can vary at each iteration.
2. (*Section IV*) Since degradations are unknown and always embedded within the degraded observations, we approximate their varying statistics at every iteration from alternative signals dubbed *surrogate* and *latent* noise, using approximations designed to curtail the signal contamination.
3. (*Sections V and IV*) At every iteration, the statistics of the degradations are represented as a noise *power spectral density* (PSD) that is adopted by the denoiser when filtering the degraded observations.
4. (*Section VI*) For various applications of sparse multidimensional and multispectral signal recovery, we analyze the effective noise and show that it is not white but it

¹Preliminary material was presented in [30], [31]. Here, we introduce a more sophisticated and accurate estimation of the power spectrum of the effective noise. Unlike prior works, we model both surrogate and latent noise terms, estimating the noise PSDs in local Fourier domain, hence supporting any filter for correlated noise. These methodological advances lead to superior quality of recovery. We further study how the tuning of the filter strength crucially depends on the model assumed for the effective noise, which we analyze in detail for a wider range of sparse signal recovery applications, under both ADMM and ISTA.

rather features significant forms of correlation, even when the recovery is carried out by an AWGN denoiser.

5. (*Section VI*) For all these applications, we compare the recovery under the modeling assumption of effective noise as AWGN versus the proposed adaptive correlated noise model. Experiments show advantage of the latter in terms of both objective and subjective visual quality as well as faster recovery.

II. ITERATIVE SPARSE SIGNAL RECOVERY

The estimation of \mathbf{x} in (1) is an ill-posed linear inverse problem which can be obtained through a regularization-based optimization problem of the form

$$\hat{\mathbf{x}} = \underset{\mathbf{v} \in \mathbb{R}^n}{\operatorname{argmin}} Q_{\mathbf{y}}(\mathbf{v}) + \gamma \mathcal{R}(\mathbf{v}), \quad (2)$$

where $Q_{\mathbf{y}}$ is the data-fidelity term enforcing the consistency of $\hat{\mathbf{x}}$ with the measurements \mathbf{y} , \mathcal{R} is the regularization term representing the prior model of \mathbf{x} , and $\gamma > 0$ is the regularization parameter balancing the contribution of both terms. Many optimization algorithms have been proposed to solve (2) [4]–[10], with \mathcal{R} explicitly modeling the sparsity of \mathbf{x} with respect to a transform.

In the remainder of this section we briefly review some widely used iterative approaches for sparse signal recovery. While this selection has no pretension of being exhaustive, we argue that other approaches (e.g., [7]–[9], [12]–[14], [17]–[24], [32]), modulo technical adjustments, can be seen as instances of the following cases.

The *alternating direction method of multipliers* (ADMM) [10] is widely used to tackle large-scale inverse problems of the form (1). The idea is to convert the unconstrained optimization problem (2) into its equivalent constrained form which is then decoupled into two separate proximal optimizations with respect to its augmented Lagrangian function:

$$\mathbf{u}_k = \operatorname{prox}_{\gamma \rho_{k-1} \mathcal{R}} \left(\overbrace{\mathbf{x}_{k-1} + \mathbf{b}_{k-1}}^{\mathbf{z}_k} \right), \quad (3a)$$

$$\mathbf{x}_k = \operatorname{prox}_{\rho_{k-1} Q_{\mathbf{y}}}(\mathbf{u}_k - \mathbf{b}_{k-1}), \quad (3b)$$

$$\mathbf{b}_k = \mathbf{b}_{k-1} + \mathbf{x}_k - \mathbf{u}_k, \quad (3c)$$

where \mathbf{u}_k and \mathbf{b}_k are respectively the auxiliary variable and the scaled Lagrange multiplier, $k \geq 1$, $\mathbf{b}_0 = \mathbf{0}$,

$$\operatorname{prox}_{\gamma \rho_{k-1} \mathcal{R}}(\cdot) \stackrel{\text{def}}{=} \underset{\mathbf{u} \in \mathbb{R}^n}{\operatorname{argmin}} \gamma \rho_{k-1} \mathcal{R}(\mathbf{u}) + \frac{1}{2} \|\mathbf{u} - \cdot\|_2^2 \quad (4)$$

is the proximal operator of $\gamma \rho_{k-1} \mathcal{R}$, and $\rho_{k-1} > 0$ is the step size. The \mathbf{x} -subproblem (3b) can be interpreted as an inversion step since it only relies on the choice of the forward model.

The *forward-backward splitting* (FBS) [5] (a.k.a. *proximal gradient*) method is another proximal splitting approach to solve (2). It follows an iterative procedure of the form

$$\mathbf{u}_k = \operatorname{prox}_{\gamma \rho_{k-1} \mathcal{R}} \left(\overbrace{\mathbf{x}_{k-1} + \mathbf{b}_{k-1}}^{\mathbf{z}_k} \right), \quad (5a)$$

$$\mathbf{x}_k = \mathbf{u}_k + \nu_k (\mathbf{u}_k - \mathbf{u}_{k-1}), \quad (5b)$$

$$\mathbf{b}_k = -\rho_k \nabla Q_{\mathbf{y}}(\mathbf{x}_k), \quad (5c)$$

where ∇ is the gradient operator, $\mathbf{b}_0 = \mathbf{0}$, $\mathbf{x}_0 = \mathbf{u}_0 \in \mathbb{R}^n$, $\nu_k \in [0, 1)$ is the relaxation parameter, and $\rho_{k-1} > 0$ is again the step size. For $\nu_k = 0 \forall k \geq 1$, the FBS (5a)–(5c) is known as ISTA (iterative shrinkage/thresholding algorithm) [4], while for $\nu_k = (t_{k-1} - 2)/t_k$ with $t_k = 1 + \sqrt{1 + t_{k-1}^2}$ and $t_0 = 2$ we have its accelerated version known as FISTA (fast ISTA) [6].

Typical examples of $Q_{\mathbf{y}}(\mathbf{v})$ in (2) are $\frac{1}{2} \|\mathbf{y} - \mathbf{M}\mathbf{v}\|_2^2$ and $\frac{1}{2} \|\mathbf{M}^\dagger(\mathbf{y} - \mathbf{M}\mathbf{v})\|_2^2$, where \dagger denotes the pseudoinverse. For $\mathcal{R}(\mathbf{v})$ one commonly uses $\|\mathcal{T}(\mathbf{v})\|_p$ for $p \in \{0, 1\}$ and a sparsifying transform \mathcal{T} . If \mathcal{T} is orthonormal then the proximal of \mathbf{z}_k (3a) and (5a) reduces to the shrinkage of $\mathcal{T}(\mathbf{z}_k)$ with hard threshold $\sqrt{2\gamma\rho_{k-1}}$ for $p=0$, or with soft threshold $\gamma\rho_{k-1}$ for $p=1$ [33]. Similar results hold also for other basic regularizers \mathcal{R} [34].

In (3a) and (5a), the action of the proximal operator (4) on

$$\mathbf{z}_k = \mathbf{x}_{k-1} + \mathbf{b}_{k-1}, \quad (6)$$

can be regarded as a *denoiser* seeking to recover \mathbf{x} from

$$\mathbf{z}_k = \mathbf{x} + \mathbf{e}_k, \quad (7)$$

where \mathbf{z}_k and \mathbf{e}_k respectively represent the *noisy signal* to be filtered and the *effective noise* (or degradation) at each iteration of the algorithm. Thus (3a) and (5a) are denoising steps dependent on the prior model.

Because natural signals enjoy more elaborate forms of regularity than are often captured by a simple \mathcal{R} such as in the above example, the recovery of \mathbf{x} in (1) can be significantly improved by using advanced denoising filters as implicit prior models instead of proximal (4). This pragmatic approach is known as *plug-and-play prior* (PnP) framework [15], [17]–[22], [25], [27], [35]. In PnP, (3a) and (5a) are replaced by

$$\mathbf{u}_k = \Phi(\mathbf{z}_k, \lambda, \Theta_{\mathbf{e}_k}), \quad (8)$$

where Φ denotes a denoiser, the parameter $\lambda > 0$ controls the filtering strength, and $\Theta_{\mathbf{e}_k}$ is a characterization of noise model in \mathbf{z}_k assumed by Φ . For an AWGN denoiser, $\Theta_{\mathbf{e}_k}$ is the noise variance $\sigma_{\mathbf{e}_k}^2$, whereas it can be the noise PSD $\Psi_{\mathbf{e}_k}$ when Φ is a correlated noise denoiser. The choice of Φ , λ and $\Theta_{\mathbf{e}_k}$ characterizes how signal and noise are modeled for regularization in PnP. Despite \mathbf{e}_k may not be AWGN [16], [29], [32], [36]–[39] and although various denoisers can be used within PnP, nearly all works have so far adopted AWGN filters, owing to popularity, vast off-the-shelf availability, ease of use, and the fact that simplest priors like those illustrated above lead to shrinkage with constant threshold, which is a prototype of AWGN denoising. We take a radical departure from these conventional models.

III. STATIONARY NOISE MODELING IN SPARSE SIGNAL RECOVERY

In the generic form (7), we explicitly model \mathbf{e}_k as a zero-mean additive stationary correlated noise [40], i.e.

$$\mathbf{e}_k = \boldsymbol{\omega}_k \circledast \mathbf{g}_k, \quad (9)$$

where $\boldsymbol{\omega}_k$ is standard white Gaussian noise, \circledast denotes convolution, and \mathbf{g}_k is a kernel which specifies the correla-

tion in \mathbf{e}_k . Let \mathcal{F} denote the global Fourier transform², then $\mathcal{F}(\mathbf{e}_k) = \mathcal{F}(\mathbf{g}_k) \odot \mathcal{F}(\boldsymbol{\omega}_k)$ where \odot denotes elementwise product, and the Fourier PSD of \mathbf{e}_k is

$$\Psi_{\mathbf{e}_k} = \text{var}\{\mathcal{F}(\mathbf{e}_k)\} = n|\mathcal{F}(\mathbf{g}_k)|^2, \quad (10)$$

describing how the noise power is distributed over frequencies. Accordingly, $\Psi_{\mathbf{e}_k}^{1/2} = \text{std}\{\mathcal{F}(\mathbf{e}_k)\}$ represents the Fourier root-PSD of \mathbf{e}_k . In particular, if \mathbf{g}_k in (9) is a scaled Dirac delta with mass $\sigma_{\mathbf{e}_k}$, then \mathbf{e}_k is AWGN with variance $\sigma_{\mathbf{e}_k}^2$, i.e. $\mathbf{e}_k \sim \mathcal{N}(\mathbf{0}, \sigma_{\mathbf{e}_k}^2 \mathbf{I}_n)$ where \mathbf{I}_n denotes the $n \times n$ identity matrix; this white noise features a flat PSD $\Psi_{\mathbf{e}_k} \equiv n\sigma_{\mathbf{e}_k}^2$ which is identically equal to n times the variance.

The subindex k in $\Psi_{\mathbf{e}_k}$ emphasizes that these PSDs may vary at each iteration of sparse signal recovery.

IV. NOISE PSD ESTIMATION IN SPARSE SIGNAL RECOVERY

Before addressing the proposed strategy for estimating $\Psi_{\mathbf{e}_k}$ under the correlated noise modeling of the effective noise \mathbf{e}_k (9), we first discuss two alternative procedures to set $\sigma_{\mathbf{e}_k}^2$ under the AWGN modeling of \mathbf{e}_k , by which $\Psi_{\mathbf{e}_k} \equiv n\sigma_{\mathbf{e}_k}^2$.

A. Non-adaptive Variance under AWGN Modeling

The first alternative is a non-adaptive procedure shared by many works, e.g., [15], [17]–[20], [22], [25], [27], [32], [35], where an “estimate” of $\sigma_{\mathbf{e}_k}$ used for a given PnP denoiser is determined directly by the choice of the parameters λ , γ and ρ_{k-1} and of the regularizer \mathcal{R} in the definition of the proximal operator (4). For instance, let $\mathcal{R}(\mathbf{u}) = \|\mathcal{T}(\mathbf{u})\|_0$ in (4) for orthonormal \mathcal{T} . Then an estimate of $\sigma_{\mathbf{e}_k}$ can be defined as

$$\hat{\sigma}_{\mathbf{e}_k} = \lambda^{-1} \sqrt{2\gamma\rho_{k-1}}. \quad (11)$$

Analogously, $\hat{\sigma}_{\mathbf{e}_k} = \lambda^{-1}\gamma\rho_{k-1}$ for $\mathcal{R}(\mathbf{u}) = \|\mathcal{T}(\mathbf{u})\|_1$.

B. Adaptive AWGN Variance Estimation

The second alternative instead estimates the AWGN variance adaptively from the highest-frequencies of the noisy image \mathbf{z}_k (6), which are likely dominated by noise. This is most commonly done by applying a robust estimator such as the sample median absolute deviation (MAD) [41] over the high-passed \mathbf{z}_k to discard the influence of outliers due to sharp features such as edges and singularities. We can formulate it as

$$\hat{\sigma}_{\mathbf{e}_k} = 1.4826 \|\varphi\|_2^{-1} \text{MAD}\{\mathbf{z}_k \otimes \varphi\}, \quad (12)$$

where φ is a high-pass filter and the factor 1.4826 calibrates for a normal distribution, which can be assumed regardless of the distribution of \mathbf{e}_k , since convolving with φ provides approximate Gaussianization by the central-limit theorem [42].

²Here \mathcal{F} and \otimes are multidimensional operators matching the dimension of the signals they are applied to (e.g., 3D for a grayscale video). Throughout the paper, vectorized signals are tacitly reshaped to their native dimensionality when subject to multidimensional operators like convolution, transforms, filters, etc.; the output of these operators is returned to a vector.

C. Approximation of the PSD of \mathbf{e}_k

Estimating $\Psi_{\mathbf{e}_k}$ under the correlated noise modeling of \mathbf{e}_k is far more involved. Unlike the flat PSD of a white noise, estimating $\Psi_{\mathbf{e}_k}$ from the spectrum of \mathbf{z}_k can lead to significant overestimation, because the spectrum of \mathbf{x} can be large and dominating over frequencies other than the highest ones.

The identity between (6) and (7) yields the effective noise $\mathbf{e}_k = \mathbf{b}_{k-1} + \mathbf{x}_{k-1} - \mathbf{x}$ which can be decomposed as

$$\mathbf{e}_k = \underbrace{\mathbf{b}_{k-1} + \mathbf{M}^\dagger \mathbf{M}(\mathbf{x}_{k-1} - \mathbf{x})}_{\mathbf{s}_k} + \underbrace{\mathbf{M}_\perp(\mathbf{x}_{k-1} - \mathbf{x})}_{\mathbf{l}_k}, \quad (13)$$

where $\mathbf{M}_\perp = \mathbf{I}_n - \mathbf{M}^\dagger \mathbf{M}$ is the orthogonal projection on the null space of \mathbf{M} . We dub \mathbf{s}_k the *surrogate noise*, and \mathbf{l}_k the *latent noise*. The decomposition of \mathbf{e}_k (13) allows us to replace $\mathbf{M}\mathbf{x}$ with $\mathbf{y} - \boldsymbol{\epsilon}$ through the forward model (1), where by this substitution the surrogate noise \mathbf{s}_k will not require the direct access to \mathbf{x} .

From (13), the global Fourier PSD of \mathbf{e}_k can be written as $\Psi_{\mathbf{e}_k} = \Psi_{\mathbf{s}_k} + \Psi_{\mathbf{l}_k} + 2\text{cov}\{\mathcal{F}(\mathbf{s}_k), \mathcal{F}(\mathbf{l}_k)\}$, where $\Psi_{\mathbf{s}_k}$ and $\Psi_{\mathbf{l}_k}$ are respectively the global Fourier PSDs of \mathbf{s}_k and \mathbf{l}_k , and $\text{cov}\{\cdot, \cdot\}$ is the covariance between its inputs. A series of pragmatic and simplifying assumptions are made in order to approximate $\Psi_{\mathbf{e}_k}$. First, we assume $\text{cov}\{\mathcal{F}(\mathbf{s}_k), \mathcal{F}(\mathbf{l}_k)\} = \mathbf{0}$, i.e. ignore any dependence between \mathbf{s}_k and \mathbf{l}_k , which results in³

$$\Psi_{\mathbf{e}_k} \approx \Psi_{\mathbf{s}_k} + \Psi_{\mathbf{l}_k}. \quad (14)$$

The two addends are addressed in the following subsections.

1) *Approximation of the PSD of Surrogate Noise \mathbf{s}_k* : Upon defining the measurement residual as

$$\mathbf{r}_k = \mathbf{y} - \mathbf{M}\mathbf{x}_k \quad (15)$$

and by simple substitutions, we can rewrite $\mathbf{M}^\dagger \mathbf{M}(\mathbf{x}_{k-1} - \mathbf{x})$ as $\mathbf{M}^\dagger(\boldsymbol{\epsilon} - \mathbf{r}_{k-1})$. Thus, we can further decompose \mathbf{s}_k in (13) as

$$\mathbf{s}_k = \underbrace{\mathbf{b}_{k-1} - \mathbf{M}^\dagger \mathbf{r}_{k-1}}_{\mathbf{w}_k} + \mathbf{M}^\dagger \boldsymbol{\epsilon}, \quad (16)$$

which gives $\Psi_{\mathbf{s}_k} = \Psi_{\mathbf{w}_k} + \Psi_{\mathbf{M}^\dagger \boldsymbol{\epsilon}} + 2\text{cov}\{\mathcal{F}(\mathbf{w}_k), \mathcal{F}(\mathbf{M}^\dagger \boldsymbol{\epsilon})\}$. Although the particular realization of $\boldsymbol{\epsilon}$ is unknown, its statistics are either known or can be estimated upon using the prior information on acquisition, and we can thus compute $\Psi_{\mathbf{M}^\dagger \boldsymbol{\epsilon}}$. Next, for the simplicity of computation, we assume that \mathbf{w}_k and $\mathbf{M}^\dagger \boldsymbol{\epsilon}$ have perfect correlation, i.e. $\text{cov}\{\mathcal{F}(\mathbf{w}_k), \mathcal{F}(\mathbf{M}^\dagger \boldsymbol{\epsilon})\} = \Psi_{\mathbf{w}_k}^{1/2} \odot \Psi_{\mathbf{M}^\dagger \boldsymbol{\epsilon}}^{1/2}$, which results in

$$\Psi_{\mathbf{s}_k} \approx (\Psi_{\mathbf{w}_k}^{1/2} + \Psi_{\mathbf{M}^\dagger \boldsymbol{\epsilon}}^{1/2})^2. \quad (17)$$

2) *Approximation of the PSD of Latent Noise \mathbf{l}_k* : Unlike \mathbf{s}_k , \mathbf{l}_k cannot be rewritten easily without involving \mathbf{x} , as $\mathbf{M}_\perp \mathbf{x}$ is not available.

For an arbitrary square matrix $\mathbf{A} \in \mathbb{C}^{n \times n}$, we can define

$$\mathbf{f}_{\mathbf{A}}[\xi] = \frac{1}{n} \|\phi_{\mathcal{F}}^{\xi \top} \mathbf{A}\|_2^2, \quad (18)$$

³Our preliminary works [30] and [31] utilized $\Psi_{\mathbf{e}_k} = \Psi_{\mathbf{b}_{k-1}}$. Hence, we emphasize that (14) is a compromise and alternative definitions are possible. We discuss some of them in Section VII-C.

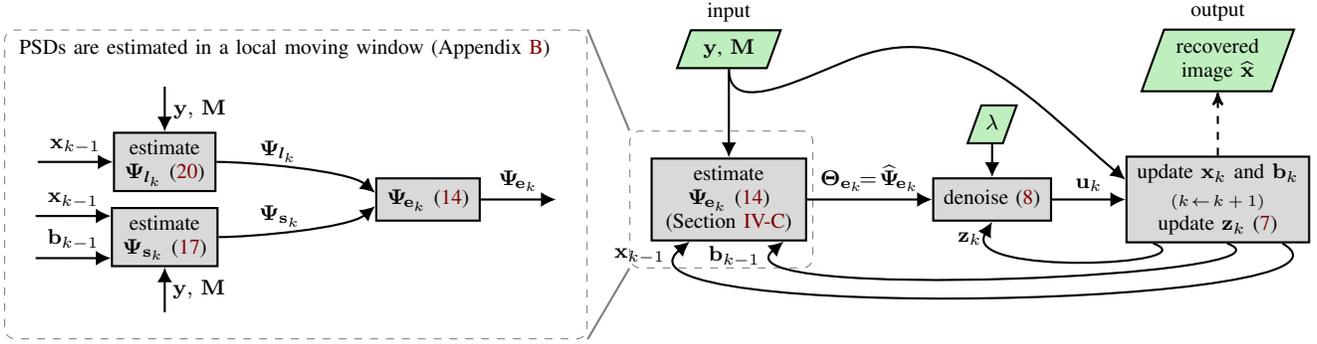


Figure 1. The flowchart of the proposed sparse signal recovery via adaptive correlated noise model.

where $\phi_{\mathcal{F}}^{\xi}$ is the ξ -th basis vector of \mathcal{F} , and the superscript \top denotes the Hermitian transpose. Note that for any AWGN η with flat PSD Ψ_{η} , $\Psi_{A\eta} = \mathbf{f}_A \odot \Psi_{\eta}$ and, by substituting $\Psi_{\eta} = \Psi_{M^{\dagger}M\eta}/\mathbf{f}_{M^{\dagger}M}$ into $\Psi_{M\perp\eta} = \mathbf{f}_{M\perp} \odot \Psi_{\eta}$, we have

$$\Psi_{M\perp\eta}[\xi] = \frac{\mathbf{f}_{M\perp}[\xi]}{\mathbf{f}_{M^{\dagger}M}[\xi]} \Psi_{M^{\dagger}M\eta}[\xi] \quad \forall \xi \text{ s.t. } \mathbf{f}_{M^{\dagger}M}[\xi] \neq 0. \quad (19)$$

Noting next that $\Psi_{M^{\dagger}M(x_{k-1}-x)} = \Psi_{M^{\dagger}(\epsilon - \mathbf{r}_{k-1})}$, and assuming $\text{cov}\{\mathcal{F}(M^{\dagger}\mathbf{r}_{k-1}), \mathcal{F}(M^{\dagger}\epsilon)\} = \Psi_{M^{\dagger}\mathbf{r}_{k-1}}^{1/2} \odot \Psi_{M^{\dagger}\epsilon}^{1/2}$ (i.e. perfect correlation between $M^{\dagger}\mathbf{r}_{k-1}$ and $M^{\dagger}\epsilon$, occurring when $M\mathbf{x}_{k-1} = M\mathbf{x}$), we have $\Psi_{M^{\dagger}M(x_{k-1}-x)} = (\Psi_{M^{\dagger}\mathbf{r}_{k-1}}^{1/2} - \Psi_{M^{\dagger}\epsilon}^{1/2})^2$. By further replacing η in (19) with $x_{k-1} - x$ (even if this might violate the AWGN condition needed for (19) to hold as an equality) we approximate Ψ_{I_k} as

$$\Psi_{I_k}[\xi] \approx \begin{cases} \frac{\mathbf{f}_{M\perp}[\xi]}{\mathbf{f}_{M^{\dagger}M}[\xi]} (\Psi_{M^{\dagger}\mathbf{r}_{k-1}}^{1/2} - \Psi_{M^{\dagger}\epsilon}^{1/2})^2[\xi] & \text{if } \mathbf{f}_{M^{\dagger}M}[\xi] \neq 0, \\ 0 & \text{otherwise.} \end{cases} \quad (20)$$

3) *Estimation of Ψ_{e_k} in Local Moving Window*: Since sample estimates of Ψ_{s_k} (17) and Ψ_{I_k} (20) cannot be directly computed from a single observation of s_k and I_k , we estimate them with respect to a local Fourier transform of smaller size on a moving window, where different window positions correspond to having several instances of a stationary process. These small-size PSDs are then summed according to (14) and upsampled, providing an estimate $\hat{\Psi}_{e_k}$ of Ψ_{e_k} . These operations are detailed in Appendix B. The overall proposed sparse signal recovery pipeline is illustrated in Figure 1.

V. STATIONARY NOISE MODELING IN TRANSFORM-DOMAIN AND BMxD FILTERS

Various types of denoisers Φ (8) have been adopted for sparse signal recovery such as total-variation regularization, nonlocal means, and neural networks. In this work, we nevertheless focus on transform-domain filters as: 1) they have been extensively used in sparse signal recovery (see, e.g., [11]–[17], [19], [22], [24]–[27], [32], [35]); 2) their shrinkage core can be easily linked to explicit regularizers \mathcal{R} (e.g., ℓ_0 or ℓ_1 norms) for which exist theoretical guarantees of convergence [17], [19]; and, most important for this work, 3) they allow for a relatively effortless and informative use and exploration of correlated noise models.

Transform-domain filters attenuate noise through a sparsity-promoting thresholding of the noisy spectrum. The core operation inside such filters can be formulated for (7) as

$$\mathcal{T}^{-1}(\Upsilon(\mathcal{T}(z_k), \tau_k)), \quad (21)$$

where \mathcal{T} is a chosen decorrelating/sparsifying transform, and Υ is a shrinkage operator with the threshold τ_k . In (21), $\mathcal{T}(z_k)[\xi] = \langle z_k, \phi_{\mathcal{T}}^{\xi} \rangle = \phi_{\mathcal{T}}^{\xi \top} z_k$, where $\langle \cdot, \cdot \rangle$ denotes the inner product, and $\phi_{\mathcal{T}}^{\xi}$ is the ξ -th basis vector of \mathcal{T} . The threshold τ_k acts as a gauge of the magnitude of each transform coefficient against that of the corrupting noise. When denoising stationary correlated noise, τ_k can be set as

$$\tau_k[\xi] = \lambda \sigma_{\mathcal{T}(e_k)}[\xi], \quad (22)$$

where $\sigma_{\mathcal{T}(e_k)} = \text{std}\{\mathcal{T}(e_k)\} = \{\text{std}\{\langle e_k, \phi_{\mathcal{T}}^{\xi} \rangle\}\}_{\xi}$ is the noise \mathcal{T} -root-PSD, i.e. the standard deviation of the noise e_k in \mathcal{T} domain, and $\lambda > 0$ is a parameter scaling the $\sigma_{\mathcal{T}(e_k)}$ to modulate the filtering strength (22). The noise \mathcal{T} -PSD (i.e. the variance of the noise e_k in \mathcal{T} domain) is

$$\begin{aligned} \sigma_{\mathcal{T}(e_k)}^2[\xi] &= \text{var}\{\langle e_k, \phi_{\mathcal{T}}^{\xi} \rangle\} = \\ &= \|\mathbf{g}_k \otimes \phi_{\mathcal{T}}^{\xi}\|_2^2 = \frac{1}{n^2} \|\Psi_{e_k} \odot \mathcal{F}^2(\phi_{\mathcal{T}}^{\xi})\|_1. \end{aligned} \quad (23)$$

A. Simplification under AWGN Modeling of e_k

For AWGN $e_k \sim \mathcal{N}(0, \sigma_{e_k}^2 \mathbf{I}_n)$, (23) simplifies to

$$\sigma_{\mathcal{T}(e_k)}^2[\xi] = \sigma_{e_k}^2 \|\phi_{\mathcal{T}}^{\xi}\|_2^2; \quad (24)$$

provided $\|\phi_{\mathcal{T}}^{\xi}\|_2 = 1$, i.e. normalization of the atoms of \mathcal{T} , (23) further reduces to a flat \mathcal{T} -PSD equal to the noise variance and consequently the filtering strength τ_k (22) is constant across ξ :

$$\tau_k[\xi] = \lambda \sigma_{e_k}, \quad \sigma_{\mathcal{T}(e_k)}^2 \equiv \sigma_{e_k}^2. \quad (25)$$

B. BMxD Filters

In this work, we leverage the family of BMxD filters in the denoising step of iterative sparse signal recovery; specifically, we adopt BM3D [43], BM4D [26], and RF3D [44]⁴.

The choice of the filter within the family depends on the data structures as well as the type of noise modeling.

⁴BMxD filters are used widely in PnP recovery [13], [15]–[17], [19], [20], [22], [25], [26], [32], [35]

For instance, BM3D [43] captures local spatial regularity within images through 2D blocks whereas BM4D [26] captures local 3D spatial/spatospectral regularity within volumetric/multispectral images using 3D cubes. Both methods exploit nonlocal self-similarity by stacking several mutually similar 2D blocks or 3D cubes into, respectively, 3D or 4D arrays called “groups” that are jointly filtered by a 3D or 4D transform-domain shrinkage. In what follows, we denote the transformation applied to groups by \mathcal{T}_{3D} or \mathcal{T}_{4D} . Similarly, RF3D [44] captures local and temporal regularity within a video by stacking 2D blocks along motion trajectories into 3D spatiotemporal volumes. Such volumes are akin to 3D arrays in BM3D, because blocks along a motion trajectory are very similar to each other; thus volumes are filtered by shrinkage in \mathcal{T}_{3D} domain.

Even though the BMxD filters can operate shrinkage as two successive stages of hard-thresholding and Wiener filtering, for PnP recovery we employ the hard-thresholding stage only, which can be formalized as an instance of (21)-(22).

Note that BM3D, as well as other BMxD filters, operates on a *multitude* of different groups extracted from \mathbf{z}_k , which are separately processed and aggregated. Therefore, the transform \mathcal{T} in (21) does not represent a single instance of \mathcal{T}_{3D} , but rather it stands for a highly *overcomplete collection* of groupwise transforms. Likewise, the \mathcal{T} -PSD (23) stands for a collection of groupwise PSDs $\sigma_{\mathcal{T}_{3D}(\mathbf{e}_k)}^2$.

Despite these general common features, these three BMxD filters differ significantly in the way noise is modeled and in the efficient computation of the \mathcal{T} -PSD (23), as we further detail in Appendix A.

C. Selection of the Threshold Parameter λ

The choice of the value of λ corresponds to choosing prior parameters of the denoising problem. Besides manual tuning, there are several established strategies for setting the value of λ . The most notable are the so-called “universal” threshold [33], which for an array of size $b \times b \times d$ is $\lambda = \sqrt{2 \log(b^2 d)}$, and the data-driven empirical thresholds based on Stein’s Unbiased Risk Estimate (SURE) [45], [46] or on the False Discovery Rate [47]. These thresholding schemes are shown to satisfy certain minimax optimality properties under white and, to some degree, also for stationary correlated noise [33], [46]–[49] when used in conjunction with basic orthonormal wavelets and other redundant wavelet or multiscale sparsifying transforms. Although sophisticated denoisers such as those considered in the present work leverage highly adaptive transform-domain representations which are significantly more involved than classical wavelets, they nevertheless adopt values of λ that are often quantitatively close to the values suggested by these established schemes. For instance, when working with 8×8 blocks, BM3D adopts by default a value of λ equal to 2.7, which is only marginally smaller than the value 2.9 suggested by the universal threshold for image patches of comparable size.

In (22) or (25), a large λ or overestimation of either $\sigma_{\mathcal{T}(\mathbf{e}_k)}$ or $\sigma_{\mathbf{e}_k}$ causes oversmoothing, i.e. excessive loss of features of \mathbf{x} in the filtered image. Conversely, a small λ or underestimation of either $\sigma_{\mathcal{T}(\mathbf{e}_k)}$ or $\sigma_{\mathbf{e}_k}$ leads to undersmoothing, i.e. a

significant portion of \mathbf{e}_k is left in the filtered image [33]. While ad-hoc tuning of λ can help to mitigate a systematic over- or under-estimation of the noise PSD, one cannot deal with cases where part of the spectrum is overestimated while the rest is underestimated (such as when approximating correlated noise by an AWGN model) simply by tuning the scalar λ (see, e.g., the extensive analysis in [44]).

The knowledge of $\sigma_{\mathcal{T}(\mathbf{e}_k)}$ or $\sigma_{\mathbf{e}_k}$ and a proper selection of λ are crucial for effective filtering as the main stage of PnP.

VI. EXPERIMENTS AND ANALYSIS

We compare the recovery results obtained by our proposed adaptive correlated noise model against those of AWGN models on several applications of sparse signal recovery via PnP-ADMM and PnP-ISTA: compressive spectral imaging (Section VI-A), compressive temporal imaging (Sections VI-B), 2D tomography (Section VI-C) and 3D tomography (Section VI-D). Experiments are carried out for both noise-free ($\epsilon = \mathbf{0}$) and noisy (AWGN ϵ) measurements, with the exception of Section VI-B where we model ϵ as AWGN and we estimate its variance σ_ϵ^2 .

Objective quality of the recovered \mathbf{x}_k is measured as the peak SNR (PSNR) in dB: $20 \log_{10}(\sqrt{n} \max(\mathbf{x}) \|\mathbf{x} - \mathbf{x}_k\|_2^{-1})$; consequently, we use $\mathcal{Q}_y(\mathbf{v}) = \frac{1}{2} \|\mathbf{M}^\dagger(\mathbf{y} - \mathbf{M}\mathbf{v})\|_2^2$ in (2), which complies with the mean squared-error in image domain.

In all experiments, we set $\mathbf{x}_0 = \mathbf{M}^\dagger \mathbf{y}$, and we limit the iterations to $k_{\text{final}} = 1000$, which is large enough to ensure significant recovery.

We consider the following strategies to set λ : 1) the default value λ^{dft} used by the filter in a stand-alone denoising task; and 2) a value λ^* , common to all k ⁵, tuned for the recovery task so to maximize the PSNR at k_{final} . It is thus extremely interesting and informative to compare the value of λ^* with that of λ^{dft} , as it highlights how the filtering step in sparse signal recovery differs from that of a conventional stand-alone denoising application.

The progress of PnP iterations depends on the step size ρ_k ; for simplicity we fix $\rho_k = \rho > 0$ as commonly done, e.g., [15], [19]–[22], [32], [35]. Both ρ and γ are tuned so to maximize the PSNR at k_{final} , and thence denoted by a superscript $*$, i.e. ρ^* , γ^* . The tuning of ρ^* , γ^* , as well as of λ^* , is made separately for each recovery task and each measurement noise level. When experiments are carried out over multiple images, a common set of parameters is used for the different images, maximizing the average PSNR at k_{final} over the images⁶.

Under the AWGN modeling of \mathbf{e}_k , we follow the approaches discussed in Sections IV-A and IV-B to set τ_k (25); whereas for the correlated noise modeling of \mathbf{e}_k , we set τ_k (22)

⁵In principle one could get better results by tuning λ for different k , [31] but for the sake of simplicity we use the same value of λ across all iterations.

⁶For the tuning, we have tested several different combinations of parameters over reasonably fine grids, selecting the combinations that lead to the highest PSNR at k_{final} . Whenever the optimum values are not strictly enclosed within the grid, this is extended accordingly.

according to the strategies discussed in Section IV-C. We thus have the following settings for τ_k for $k > 1$ ⁷:

$$\left. \begin{aligned} W_1: \tau_k &\equiv \lambda^{\text{dft}} \hat{\sigma}_{\mathbf{e}_k} \\ W_1^*: \tau_k &\equiv \lambda^* \hat{\sigma}_{\mathbf{e}_k} \\ W_2^*: \tau_k &\equiv \lambda^{\text{dft}} \sqrt{2\gamma^* \rho^*} \\ C: \tau_k &= \lambda^{\text{dft}} \hat{\sigma}_{\mathcal{T}(\mathbf{e}_k)} \\ C^*: \tau_k &= \lambda^* \hat{\sigma}_{\mathcal{T}(\mathbf{e}_k)} \end{aligned} \right\} \begin{array}{l} \text{from (25) with } \sigma_{\mathbf{e}_k} \text{ estimated by (12)} \\ \text{from (25) with } \sigma_{\mathbf{e}_k} \text{ "estimated" by (11)} \\ \text{from (22) and (23) with } \Psi_{\mathbf{e}_k} \\ \text{estimated as in Section IV-C} \end{array}$$

As hinted by notation, settings W_1, W_1^*, W_2^* are all based on a white noise model for \mathbf{e}_k , whereas both C and C^* assume that \mathbf{e}_k is correlated noise. The superscript $*$ denotes settings where the filtering strength τ_k is subject to scaling optimization, in the following sense. None of the parameters optimized for W_1 and C directly controls τ_k , which is adaptively set based on either a variance estimate $\hat{\sigma}_{\mathbf{e}_k}^2$ (12) or a PSD estimate $\hat{\Psi}_{\mathbf{e}_k}$ (providing via (23) the estimate $\hat{\sigma}_{\mathcal{T}(\mathbf{e}_k)}^2$ of the \mathcal{T} -PSD), with the PnP denoiser Φ (8) employed as in a stand-alone denoising task with default λ^{dft} . By optimizing λ^* in W_1^* and C^* , and γ^* in W_2^* , we can directly control τ_k in order to maximize the final recovery quality, operating Φ in PnP with a different strength than one would use in a stand-alone denoising task.

To evaluate the recovery performance under correlated vs. AWGN modeling of \mathbf{e}_k , it is crucial to compare the settings where the denoising step is treated as a stand-alone denoising problem with τ_k determined directly by the model assumed for \mathbf{e}_k , separately from those that feature an extra tuning to directly aid the recovery. Namely, C shall be compared with W_1 , whereas C^* with W_1^*, W_2^* .

Technical details of each experiment are given in the Appendix C. The MATLAB code used for the experiments of this section is available at <https://webpages.tuni.fi/foi/varikas/>.

A. Compressive Spectral Imaging

We begin with multispectral image recovery using the central 256×256 -pixel portion of three multispectral images (*chart_and_stuffed_toy*, *flowers*, *feathers*) from the CAVE dataset [50] with 16 spectral bands, from 400 nm to 700 nm with a 20 nm interval. *Single-shot* spectrally compressed measurements are acquired by the simulated CASSI (coded aperture snapshot spectral imager) [51]: each band is masked by a

⁷For white as well as for correlated noise modeling of \mathbf{e}_k , the initial \mathbf{e}_1 is always assumed to be AWGN with $\sigma_{\mathbf{e}_1} = 0.2 (\max(\mathbf{z}_1) - \min(\mathbf{z}_1))$, and $\tau_1 \equiv \lambda^{\text{dft}} \sigma_{\mathbf{e}_1}$. This aggressive initialization allows us to process major distortions in \mathbf{z}_1 as noise, helping to escape from a local-minimum region.

⁸We omit the combination $\tau_k \equiv \lambda^* \sqrt{2\gamma^* \rho^*}$, since tuning γ while $\lambda = \lambda^{\text{dft}}$ already provides direct scaling of τ_k , making it superfluous to tune λ in W_2^* . Note that γ is never used by recovery algorithms with W_1, W_1^*, C , and C^* .

Table I
COMPRESSIVE SPECTRAL IMAGING (SECTION VI-A): AVERAGE PSNR (dB) OVER THE RECOVERED MULTISPECTRAL IMAGES FROM NOISE-FREE (SNR= ∞) AND NOISY (AWGN ϵ , SNR=25 AND 15 dB) SINGLE-SHOT SPECTRALLY COMPRESSED MEASUREMENTS, AT k_{final} .

SNR (dB) \rightarrow		λ			PSNR (dB)			
		∞	25	15	∞	25	15	
BM4D	PnP-ADMM	W_1	3.7	3.7	3.7	23.6	23.3	20.3
		C	3.7	3.7	3.7	36.5	34.1	30.0
		W_1^*	9.5	8.0	7.8	34.5	33.1	29.9
		W_2^*	3.7	3.7	3.7	35.0	33.0	29.9
		C^*	3.7	3.7	3.7	36.5	34.1	30.0
BM4D	PnP-ISTA	W_1	3.7	3.7	3.7	19.2	19.3	18.7
		C	3.7	3.7	3.7	31.6	33.9	29.9
		W_1^*	18.0	10.3	7.8	33.0	32.7	29.8
		W_2^*	3.7	3.7	3.7	33.2	32.7	29.8
		C^*	4.9	4.5	4.3	32.4	34.0	30.0
GPSR-BB [7]		n/a	n/a	n/a	26.5	25.9	24.9	
TwIST [8]		n/a	n/a	n/a	28.6	27.6	26.4	

shifting pseudo-random binary mask with 25% transmittance; the 16 masked bands are then summed into a single 256×256 measurement array.

The recovery is possible thanks to the regularity and smoothness among adjacent pixels and across spectral bands of the underlying multispectral image. We capture this regularity via BM4D using groups of elongated cubes of size $8 \times 8 \times 16$ and set $\lambda^{\text{dft}} = 3.7 = \sqrt{2 \log(8^2 \cdot 16)}$ according to the ‘‘universal’’ threshold [33] (as opposed to the typical isotropic volumetric imaging configuration of this filter, which adopts much smaller $4 \times 4 \times 4$ cubes and a consequently smaller $\lambda^{\text{dft}} = 2.7$).

Table I reports PSNR results with their corresponding λ values, demonstrating improvement by modeling \mathbf{e}_k as correlated rather than white noise. The improvement is especially dramatic without separate tuning of τ_k , and most noticeable for lower SNR of the input. Having $\lambda^* = \lambda^{\text{dft}}$ for C^* in PnP-ADMM suggests that we are able to operate Φ effectively as if it were a stand-alone denoising task; whereas W_1^* demands much larger λ^* , showing the inadequacy of the AWGN modeling of \mathbf{e}_k . Although PnP-ISTA performs worse than PnP-ADMM when SNR= ∞ , likely due to falling into local minima, the results by the C setting without tuning of τ_k are still competitive, also in comparison with the Barzilai-Borwein gradient projection for sparse reconstruction (GPSR-BB) [7] and two-step ISTA (TwIST) [8] algorithms, both widely used for compressive spectral recovery [52] (details

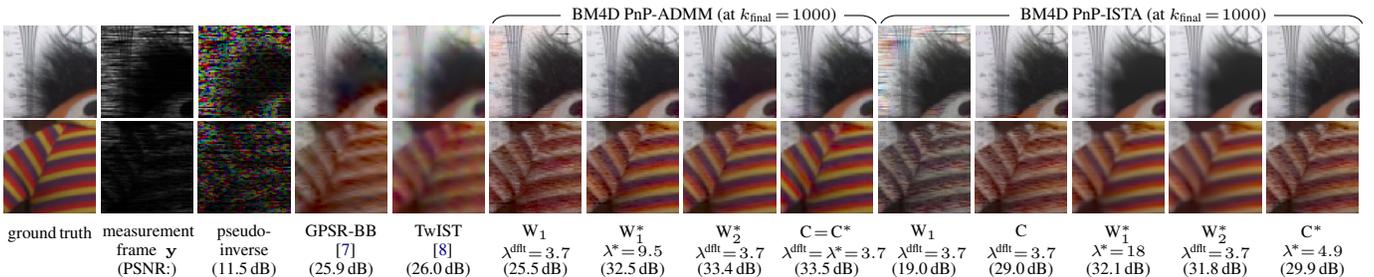


Figure 2. Compressive Spectral Imaging (Section VI-A). Visual comparison of the recovered multispectral *chart_and_stuffed_toy* image from a single-shot noise-free (i.e. $\epsilon = 0$, SNR= ∞) spectrally compressed measurements. The RGB representation is obtained by fusing channels #4, #5, #7, ..., #9, #13, ..., #16.

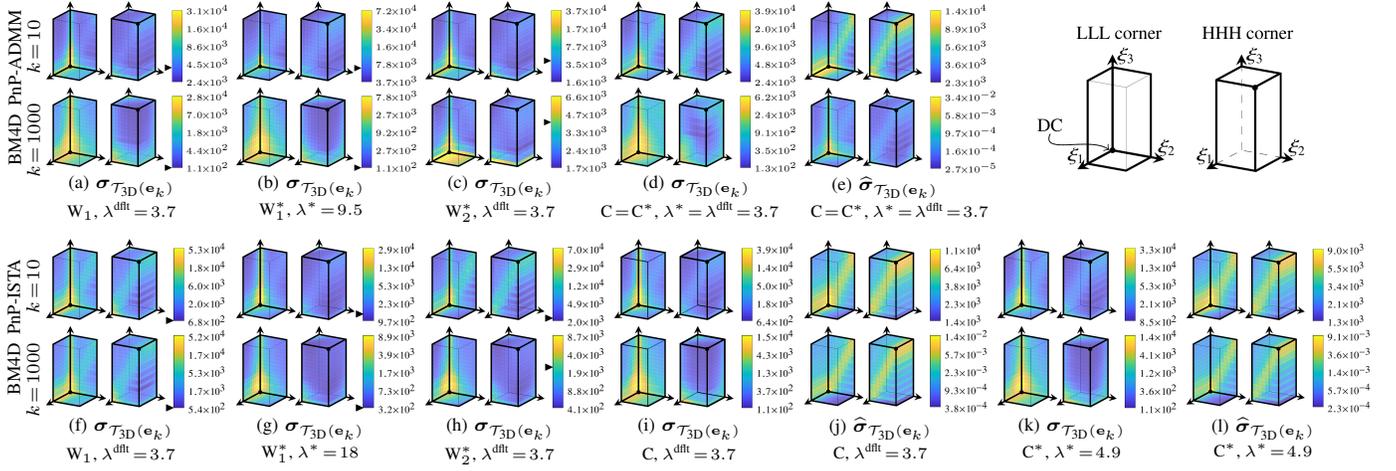


Figure 3. Compressive Spectral Imaging (Section VI-A). Spatsiospectral correlation in the effective noise e_k of the recovered multispectral *chart_and_stuffed_toy* image from noise-free (i.e. $\epsilon=0$, $\text{SNR}=\infty$) spectrally compressed measurements, visualized as the $8 \times 8 \times 16$ DCT root-PSDs. The root-PSDs denoted by $\sigma_{\mathcal{T}_{3D}}(e_k)$ and $\hat{\sigma}_{\mathcal{T}_{3D}}(e_k)$ are computed from (23) given respectively an "oracle" Ψ_{e_k} and an estimate $\hat{\Psi}_{e_k}$ that are obtained as described in Appendix B. We can show only the outer skin of these 3D objects, and separately display the three lowest-frequency faces that stem from the DC (LLL corner, at the back) and the three highest-frequency faces (HHH corner, diametrically opposed to the DC), as illustrated by the black and white sketch. The 1st and 2nd (resp. 3rd and 4th) rows correspond to the recovery via PnP-ADMM (resp. PnP-ISTA) framework with BM4D as the denoiser at $k=10$ and $k=1000$, respectively. The $\hat{\sigma}_{e_{10}}$ and $\hat{\sigma}_{e_{1000}}$ adopted by the AWGN models are indicated by \blacktriangleright in the colorbars of (a)–(c) and (f)–(h).

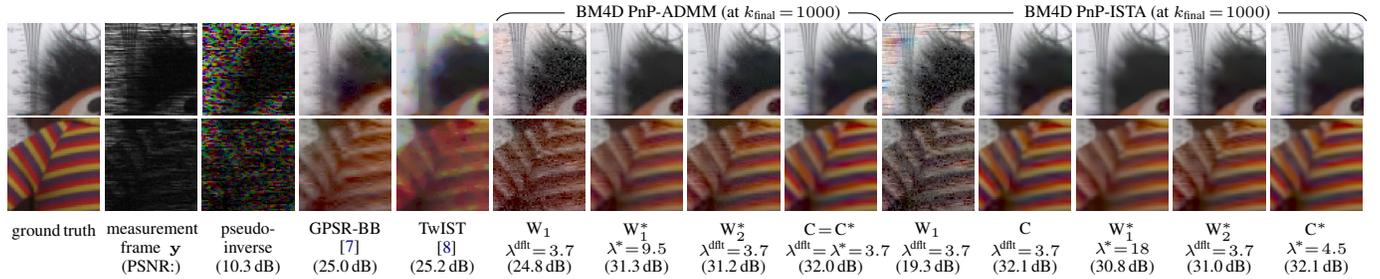


Figure 4. Compressive Spectral Imaging (Section VI-A). Recovered images as in Figure 2 but for noisy input measurements (AWGN ϵ , $\text{SNR}=25$ dB).

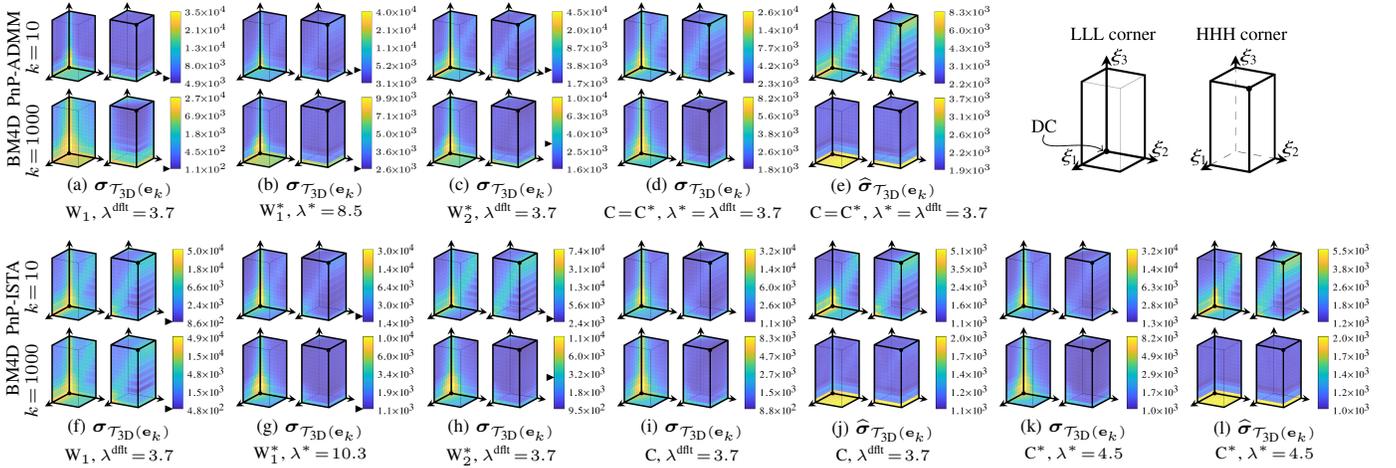


Figure 5. Compressive Spectral Imaging (Section VI-A). Root-PSD of e_k as in Figure 3, but for noisy input measurements (AWGN ϵ , $\text{SNR}=25$ dB). Please refer to the caption of Figure 3 for guidance.

on both algorithms in Appendix C-A.)

The recovered *chart_and_stuffed_toy* image from noise-free (resp. noisy) measurements and the spectra of e_k are respectively illustrated in Figures 2 and 3 (resp. Figures 4 and 5). The results of W_1 are quite poor, because its estimated $\hat{\sigma}_{e_k}$ value (denoted by \blacktriangleright in the colorbar) is smaller than most of the

"oracle"⁹ $\sigma_{\mathcal{T}_{3D}}(e_k)$, as seen in Figures 3(a,f) and 5(a,f), which makes $\tau_k \equiv \lambda^{\text{dft}} \hat{\sigma}_{e_k}$ to significantly underestimate the ideal

⁹From the ground-truth image x we can only obtain a *single* oracle realization of the effective noise $e_k = z_k - x$, but we do not have access to a genuine oracle PSD of the random process realized by e_k . We thus use quotation marks for such computed "oracle" PSDs.

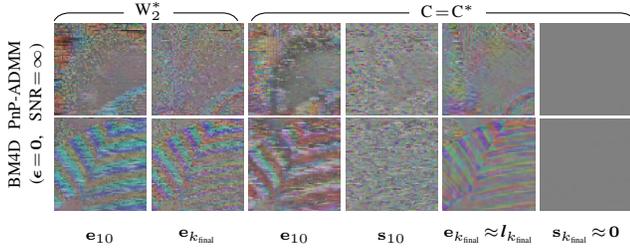


Figure 6. Compressive Spectral Imaging (Section VI-A). Visual comparison of e_k and s_k at $k=10$ and $k=1000$ corresponding to the results of Figure 2 and Figure 3(c)–(e). Nonstationarities in the noise can be noticed here.

strength $\lambda^{\text{dft}} \sigma_{\mathcal{T}_{3D}(e_k)}$. This is compensated by $\lambda^* \gg \lambda^{\text{dft}}$ in W_1^* . Indeed, in W_2^* , $\hat{\sigma}_{e_k} = \sqrt{2\gamma^* \rho^*} / \lambda^{\text{dft}}$ (11) with optimized γ^* is larger than most of the respective $\sigma_{\mathcal{T}_{3D}(e_k)}$, as seen in Figures 3(c,h) and 5(c,h). For neither W_1 , W_1^* , nor W_2^* , a constant σ_{e_k} can reasonably approximate $\sigma_{\mathcal{T}_{3D}(e_k)}$, which features order-of-magnitude differences between lowest and highest frequencies, and noticeable anisotropy. Hence, under AWGN modeling, the filter Φ must balance a compromise between systematical oversmoothing of fine details and leaving coarser structures unresolved, preventing an efficient recovery.

Figure 6 confirms that e_k contains highly correlated stationary features, but also nonstationary structures; such nonstationarities are stronger in l_k as opposed to in s_k , which we speculate is due to the direct presence of the difference image $x_{k-1} - x$ in l_k (13). To justify our speculation, we observe that in Figure 3(e) at $k=1000$, $\hat{\sigma}_{\mathcal{T}_{3D}(e_k)} \approx 0$ by which the denoiser and consequently the proximal operator in (3b) are approximately the identity operator, the algorithm reaches a steady state and $Mx_{k-1} \approx Mx$, which for $\epsilon=0$ implies $s_k \approx 0$ and $l_k \approx x_{k-1} - x$. Note that due to our approximation of Ψl_k (20), at $k=1000$ in this experiment $\hat{\Psi} l_k \approx 0$, hence $\hat{\sigma}_{\mathcal{T}_{3D}(e_k)}$ in Figure 3(e) and the “oracle” $\sigma_{\mathcal{T}_{3D}(e_k)}$ in Figure 3(d) are different. Also note that as the recovery progresses, e_k gets weaker and thus harder to estimate its PSD accurately.

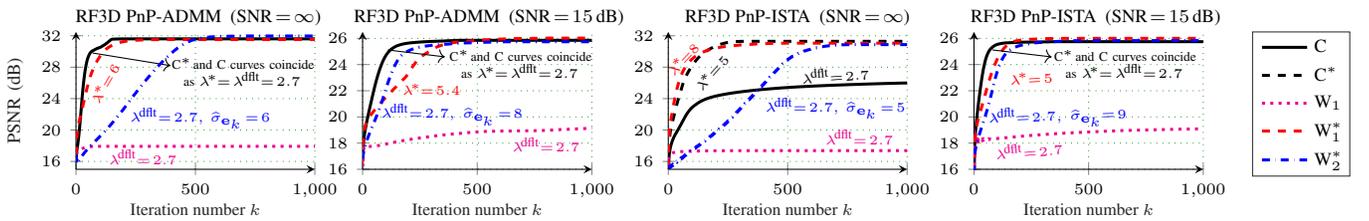


Figure 7. Compressive Temporal Imaging (Section VI-B). PSNR progression versus iteration number k for the *NBA* video recovery.

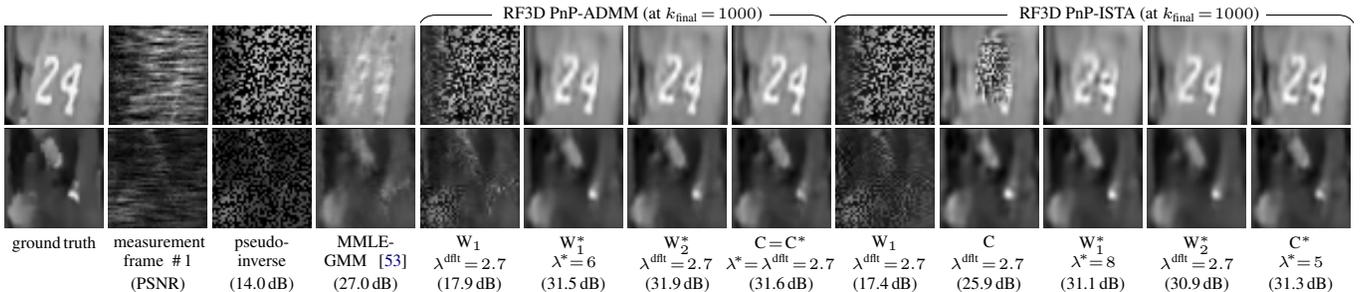


Figure 8. Compressive Temporal Imaging (Section VI-B). The recovered 6th frame of *NBA* video from noise-free ($\text{SNR}=\infty$) temporally encoded measurements.

We further note that unless the noise e_k is stationary like in (9), the PSD (even a genuine oracle PSD) does not characterize the statistics of the noise. Therefore, one should not necessarily expect improvement by the “oracle” PSD. It is then interesting to conduct an additional set of experiments with PnP-ADMM under C and C^* setting adopting an “oracle” root-PSD $\sigma_{\mathcal{T}_{3D}(e_k)}$ in the threshold (22). For the set of $\text{SNR} = \{\infty, 25, 15\}$ dB, these experiments under C and C^* respectively obtain $\{1.1, 1.0, 0.8\}$ dB and $\{0.5, 0.9, 0.8\}$ dB lower PSNR than the corresponding results in Table I, while $\lambda^* = \{3.3, 3.3, 3.7\}$ in this C^* setting. The shortcomings of the “oracle” PSD are confirmed by these inferior results, and also by $\lambda^* < \lambda^{\text{dft}}$ for noise-free and high-SNR measurements, likely to offset the nonstationary features in e_k that inflate the “oracle” PSD.

B. Compressive Temporal Imaging

We consider the recovery of the $256 \times 256 \times 32$ *NBA* video from the $256 \times 256 \times 4$ temporally compressed measurements used in [53], acquired by a simulated CACTI (coded aperture compressive temporal imager) [54]: the video is divided into 4 consecutive chunks, where each of the 8 frames in a chunk is masked by a shifting pseudo-random binary mask with 50% transmittance; the 8 masked frames are then summed into a single 256×256 measurement array. Although spatiotemporal regularity of natural scenes makes the recovery possible, the aggressive compressive capture of this highly dynamic scene with complex motions of non-rigid bodies makes this test challenging even for sophisticated methods like MMLE-GMM [53]. We capture this regularity via RF3D as the denoiser Φ .

As shown in Figures 7 and 8, the setting C provides better recovery than W_1 , while the setting C^* provides comparable results to those of W_1^* and W_2^* but with a faster recovery. Most interesting, the λ^* values for C^* are closer to λ^{dft} than those optimized for W_1^* . Figure 9 gives evidence of significant

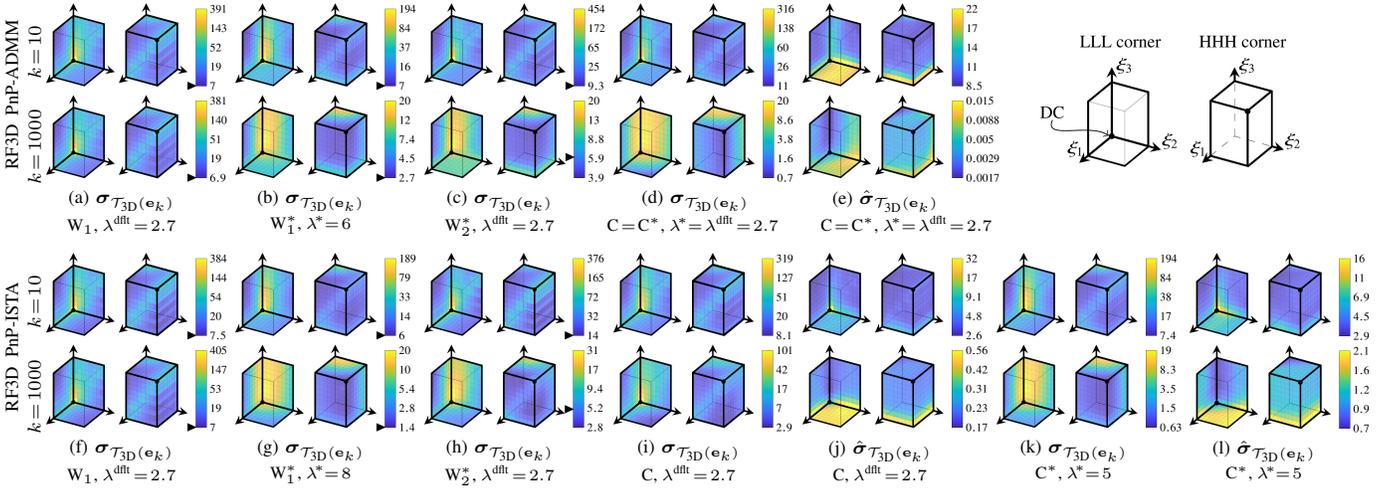


Figure 9. Compressive Temporal Imaging (Section VI-B). Spatiotemporal correlation in the effective noise of the recovered NBA video from noise-free (i.e. $\epsilon = \mathbf{0}$, $\text{SNR} = \infty$) temporally encoded measurements, visualized as the $8 \times 8 \times 9$ DCT root-PSD. The root-PSDs denoted by $\sigma_{\mathcal{T}_{3D}}(\mathbf{e}_k)$ and $\hat{\sigma}_{\mathcal{T}_{3D}}(\mathbf{e}_k)$ are computed from (23) given respectively an “oracle” $\Psi_{\mathbf{e}_k}$ and an estimate $\hat{\Psi}_{\mathbf{e}_k}$ that are obtained as described in Appendix B. The $\hat{\sigma}_{\mathbf{e}_{10}}$ and $\hat{\sigma}_{\mathbf{e}_{1000}}$ adopted by the AWGN models are indicated by \blacktriangleright in the colorbars of (a)–(c) and (f)–(h).

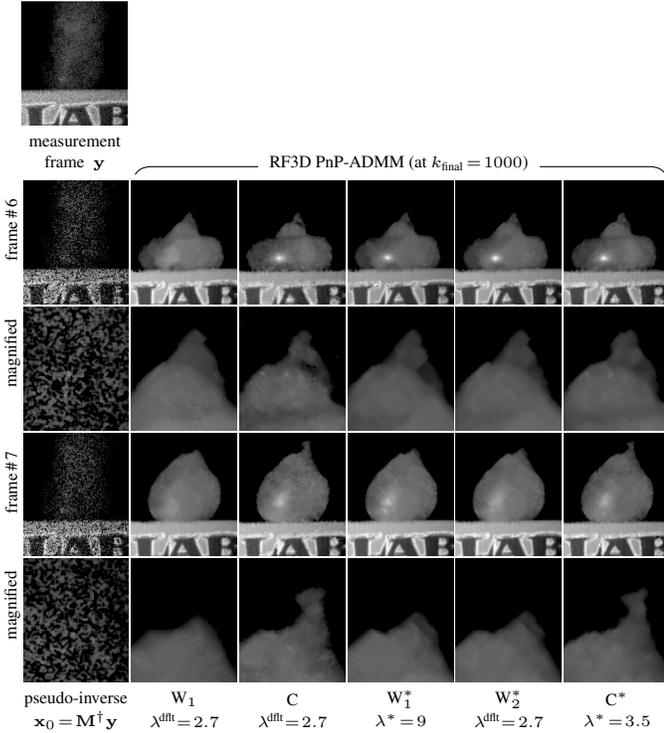


Figure 10. Compressive Video Recovery (Section VI-B). The recovered 6th and 7th frames of $512 \times 512 \times 10$ *water balloon* video from real compressive measurements. Observe how important details like the inflation outlet are recovered only under the C and C^* settings.

magnitude variations and anisotropy in $\sigma_{\mathcal{T}_{3D}}(\mathbf{e}_k)$ and $\hat{\sigma}_{\mathcal{T}_{3D}}(\mathbf{e}_k)$, consistent with the analysis in Section VI-A.

We also consider the recovery of the $512 \times 512 \times 10$ *water balloon* video from a *real* 512×512 measurement array, acquired by a snapshot compressive imaging camera [55]; each frame is masked by a distinct non-binary mask controlled by a digital micro-mirror device; the 10 masked frames are

Table II
2D TOMOGRAPHY (SECTION VI-C). AVERAGE PSNR (dB) OVER THE RECOVERED MR IMAGES FROM INCOMPLETE PSEUDO-RADIAL NOISE-FREE ($\text{SNR} = \infty$) AND NOISY (AWGN ϵ , $\text{SNR} = 20$ dB) MEASUREMENTS USING PnP-ADMM AND PnP-ISTA, AT $k_{\text{final}} = 1000$.

		15 ($m/n = 0.074$)				30 ($m/n = 0.144$)			
		λ		PSNR (dB)		λ		PSNR (dB)	
SNR (dB) \rightarrow		∞	20	∞	20	∞	20	∞	20
$\mathbf{x}_0 = \mathbf{M}^\dagger \mathbf{y}$		n/a	n/a	15.7	15.6	n/a	n/a	17.8	17.6
BM3D PnP-ADMM	W_1	2.7	2.7	18.1	16.9	2.7	2.7	24.9	20.2
	C	2.7	2.7	20.3	18.7	2.7	2.7	27.4	23.1
	\bar{W}_1^*	6.6	3.2	19.9	18.8	6.0	3.2	28.1	23.2
	W_2^*	2.7	2.7	20.3	18.7	2.7	2.7	28.4	23.2
	C^*	3.6	2.8	20.6	18.9	4.6	2.9	28.3	23.2
BM3D PnP-ISTA	W_1	2.7	2.7	18.1	16.9	2.7	2.7	24.9	20.2
	C	2.7	2.7	20.2	18.1	2.7	2.7	27.2	22.8
	\bar{W}_1^*	10.4	4.6	20.0	18.7	6.4	4.0	28.4	23.4
	W_2^*	2.7	2.7	19.9	18.9	2.7	2.7	28.4	23.3
	C^*	3.6	3.5	20.4	18.9	4.8	3.4	28.3	23.4

summed into a single temporally compressed measurement array to deliver an equivalent 500 frames-per-second video. The acquisition is subject to unknown camera shot noise ϵ that we model as AWGN for simplicity; we estimate its standard deviation similar to (12) as $\hat{\sigma}_\epsilon = 1.4826 \|\varphi\|_2^{-1} \text{MAD}\{\mathbf{y} \otimes \varphi\}$. We again employ RF3D as the denoiser Φ within PnP. As can be seen in Figure 10, C (resp. C^*) can recover more details than W_1 (resp. W_1^* and W_2^*), attesting the improvement of recovery by the proposed correlated noise modeling.

C. 2D Tomography

We consider the recovery of ten 217×181 -pixel magnitude slices (5 *transverse* and 5 *sagittal* cross-sections) of the Brain-Web Magnetic Resonance (MR) phantom [56] from incomplete radial sampling of their 2D FFT spectra. This problem is widely studied, e.g., [13], [20], [26], [27], [57]. We employ BM3D as the denoiser Φ within PnP frameworks.

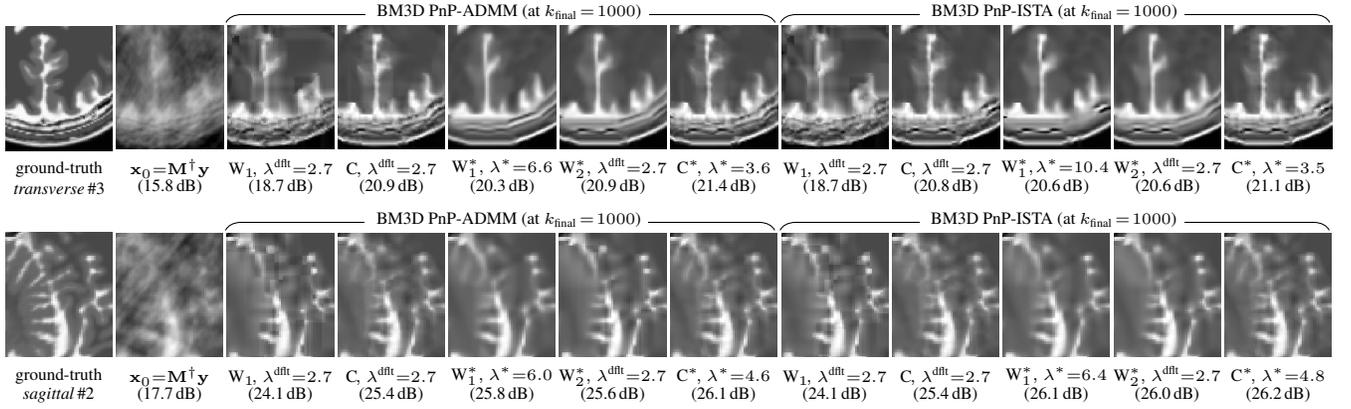


Figure 11. 2D Tomography (Section VI-C). The recovered T2 BrainWeb *transverse* and *sagittal* MR images from noise-free (i.e. $\epsilon = \mathbf{0}$, SNR= ∞) measurements acquired, respectively, by 15 and 30 radial lines of the 2D FFT spectrum at $k_{\text{final}} = 1000$. PSNR (dB) values are reported within parentheses.

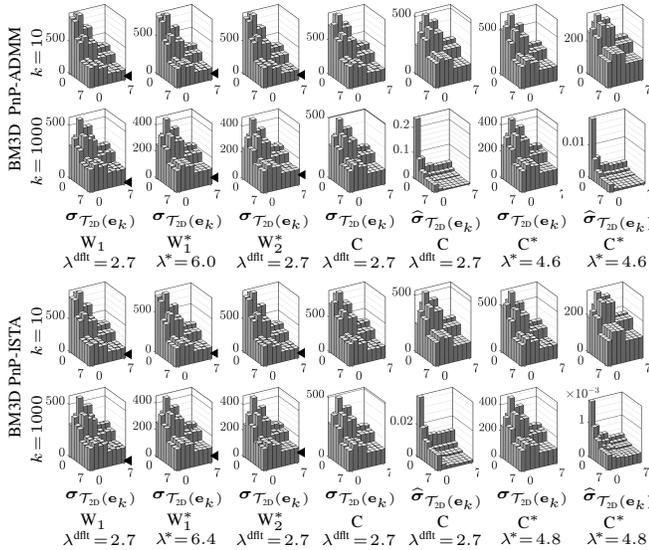


Figure 12. 2D Tomography (Section VI-C). Spatial correlation in \mathbf{e}_k of the recovered BrainWeb *sagittal* MR images (30 radial lines) corresponding to Figure 11, visualized as 8×8 Bior1.5 wavelet root-PSDs at $k=10$ and $k=1000$. The DC coefficient and the highest frequency coefficient are diametrically opposite, in $(0, 0)$ and $(7, 7)$, respectively. The root-PSDs denoted by $\sigma_{\mathcal{T}_{2D}}(\mathbf{e}_k)$ and $\hat{\sigma}_{\mathcal{T}_{2D}}(\mathbf{e}_k)$ are computed from (23) given respectively an “oracle” $\Psi_{\mathbf{e}_k}$ and an estimate $\hat{\Psi}_{\mathbf{e}_k}$ that are obtained as described in Appendix B. The $\hat{\sigma}_{\mathbf{e}_k}$ of AWGN models is indicated by \blacktriangleleft in the corresponding $\sigma_{\mathcal{T}_{2D}}(\mathbf{e}_k)$.

Table II reports the λ values and the average PSNR¹⁰ of recovered slices at k_{final} under 15 and 30 radial lines sampling, without and with noise (zero-mean complex AWGN, SNR=20 dB). As can be seen, C significantly outperforms W_1 . The performance of C^* is comparable to that of W_1^* or of W_2^* . The values of λ^* are closer to λ^{dft} under C^* than under W_1^* . Figure 11 illustrates two sets of recovered images, showing a better recovery of edges and fine details under correlated noise modeling even when the PSNR is comparable to those of AWGN model. Figure 12 gives evidence of significant variations and nonuniformity in $\sigma_{\mathcal{T}_{2D}}(\mathbf{e}_k)$ and $\hat{\sigma}_{\mathcal{T}_{2D}}(\mathbf{e}_k)$, again consistent with the analysis in Section VI-A.

¹⁰We compute PSNR only where $\mathbf{x} > 0.04 \max(\mathbf{x})$, to exclude sizable uninformative background from the phantoms used in Sections VI-C and VI-D.

Table III
3D TOMOGRAPHY (SECTION VI-D): PSNR (dB) OF THE RECOVERED 3D BrainWeb MR PHANTOM FROM INCOMPLETE PSEUDO-RADIAL NOISE-FREE (SNR= ∞) AND NOISY (AWGN ϵ , SNR=20 dB) MEASUREMENTS USING PnP-ADMM AND PnP-ISTA, AT $k_{\text{final}} = 1000$.

		$m/n = 0.136$				$m/n = 0.277$			
		λ		PSNR (dB)		λ		PSNR (dB)	
SNR (dB) \rightarrow		∞	20	∞	20	∞	20	∞	20
$\mathbf{x}_0 = \mathbf{M}^\dagger \mathbf{y}$		n/a	n/a	22.5	22.3	n/a	n/a	26.8	26.1
BM4D PnP-ADMM	W_1	2.7	2.7	32.7	25.4	2.7	2.7	39.9	30.9
	C	2.7	2.7	35.5	28.0	2.7	2.7	41.9	31.1
	W_1^*	8.0	3.2	35.3	28.2	6.0	3.4	42.5	31.2
	W_2^*	2.7	2.7	35.3	28.3	2.7	2.7	42.4	31.2
BM4D PnP-ISTA	C^*	4.2	2.7	35.9	28.0	4.8	2.7	43.1	31.1
	W_1	2.7	2.7	32.5	25.3	2.7	2.7	39.6	29.2
	C	2.7	2.7	35.4	28.1	2.7	2.7	41.9	31.2
	W_1^*	10	4.6	35.3	28.5	7.5	4.4	42.1	31.5
	W_2^*	2.7	2.7	35.2	28.4	2.7	2.7	42.2	31.4
	C^*	4.1	3.0	35.9	28.2	5.3	2.7	42.8	31.2

D. 3D Tomography

We consider the recovery of a $64 \times 64 \times 64$ -voxel magnitude of the BrainWeb MR phantom [56] from 13.6% and 27.7% 3D radial sampling of its FFT spectrum. We employ BM4D as the denoiser Φ within PnP frameworks.

Table III reports the λ values and the PSNR (dB) of the recovered images, demonstrating that the recovery under C setting significantly outperforms W_1 , while yielding comparable results to those of W_1^* and W_2^* settings which require one extra parameter to be tuned. The λ^* in C^* setting is closer to λ^{dft} than is that in W_1^* . The 3D cross-sections of the recovered magnitude are illustrated in Figure 13. Insight about these results can be gained from inspecting the spectra of \mathbf{e}_k in Figure 13, showing nonuniform root-PSDs.

VII. DISCUSSION

To the best of our knowledge, besides our preliminary works [30], [31], a correlated noise model in PnP recovery has been used only by the method [36], in a simplified setting limited to estimating the standard deviation of few individual scales of a wavelet decomposition for FISTA in radio-interferometry, by [32], estimating the variance of individual curvelet subbands

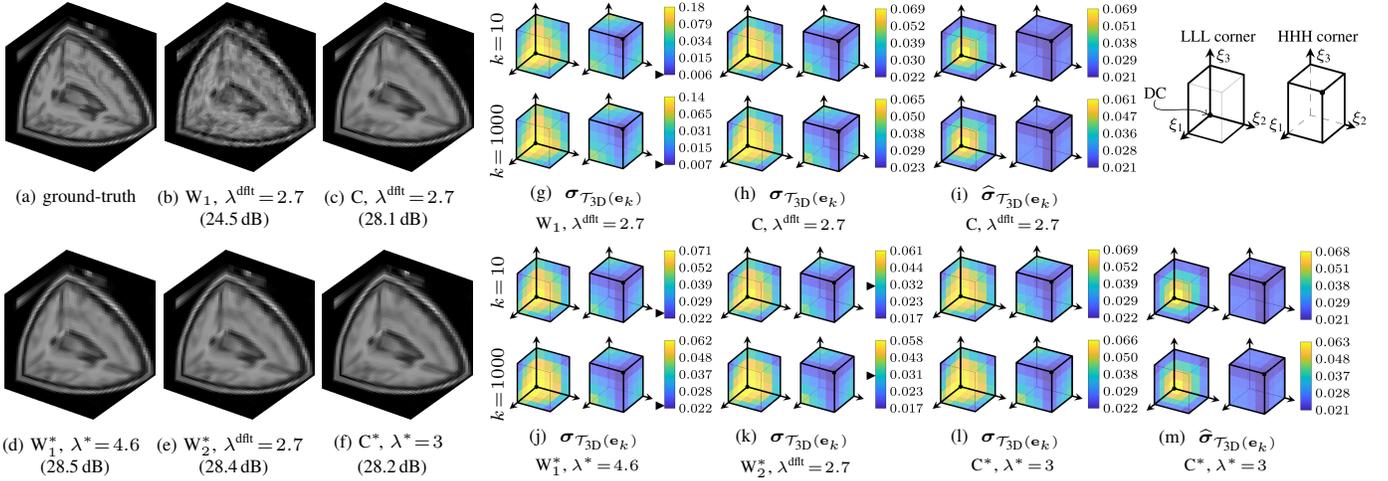


Figure 13. 3D Tomography (Section VI-D). (a)–(f): 3D cross-sections of the recovered BrainWeb phantom from 13.6% 3D radial sampling of its FFT noisy spectrum (zero-mean complex AWGN ϵ , SNR=20 dB) via BM4D PnP-ISTA at $k_{\text{final}}=1000$. (g)–(m): spatial correlation in \mathbf{e}_k of the recovered BrainWeb phantom (see Figures 13(a)–(f)), visualized as the $4 \times 4 \times 4$ 3D DCT root-PSD at iteration $k=10$ and $k=1000$. The root-PSDs denoted by $\sigma_{\mathcal{T}_{3D}(\mathbf{e}_k)}$ and $\hat{\sigma}_{\mathcal{T}_{3D}(\mathbf{e}_k)}$ are computed from (23) given respectively an “oracle” $\Psi_{\mathbf{e}_k}$ and an estimate $\hat{\Psi}_{\mathbf{e}_k}$ that are obtained as described in Appendix B. Only the lowest- and highest-frequency faces of each 3D root-PSD cube are shown. The $\hat{\sigma}_{\mathbf{e}_{10}}$ and $\hat{\sigma}_{\mathbf{e}_{100}}$ of the AWGN model are indicated by \blacktriangleright in the colorbars of (a).

within ADMM for image recovery from random projections, and later by [37]–[39], extending AMP [11], [16] in the special case of Fourier-domain variable-density sampling operator \mathbf{M} and again representing the correlation through the variance of a few wavelets subbands. Our method and framework developed in the present paper improves and generalizes fundamentally over all these prior works. Firstly, we explicitly decouple the effective noise PSD estimation in Fourier domain from the PnP denoising step, obtaining an explicit and rich representation of the Fourier PSD $\Psi_{\mathbf{e}_k}$ and thus enabling the use of any denoiser Φ for correlated noise through (8). Furthermore, we here encompass both ADMM and FBS (ISTA, FISTA, etc.) without resorting to ad-hoc tweaks, and include both surrogate and latent noise terms in the PSD estimation, which allows us to apply this unified framework to a wide range of recovery problems, including cases where \mathbf{M} is neither variable-density nor convolutional, as well as handling unusual situations featuring highly anisotropic degradation. We note that although \mathbf{e}_k may often deviate from being a purely stationary noise (as shown in Figure 6) and the estimation of $\Psi_{\mathbf{e}_k}$ be imprecise, the proposed noise model (9) offers an approximation of the correlation in \mathbf{e}_k that is nevertheless more accurate and effective than the naive AWGN model. Strong nonstationarities, and \mathbf{x} itself, make it hard to obtain a usable $\Psi_{\mathbf{e}_k}$ estimate directly from \mathbf{z}_k , which is why in this work we opt instead to estimate the noise PSD from \mathbf{s}_k and \mathbf{l}_k . Beside the technical detail given in Section IV-C, let us give an intuition of why this helps against overestimation. The significant part of the structured errors and signal \mathbf{x} leaked into \mathbf{e}_k is typically concentrated in the null space of \mathbf{M} , where it cannot be anchored to \mathbf{y} through the fidelity. Due to the decomposition of \mathbf{e}_k as (13), those errors end up in the latent noise component, hence \mathbf{s}_k is typically not structured (as can be seen in Figure 6, too). The approximation (20) used for estimating the PSD of the latent noise projects the structured errors that were in the null space of \mathbf{M} onto its row space,

thus effectively attenuating them.

A. Correlated Noise Filtering as Explicit Regularization

One can interpret correlated noise filtering in sparse signal recovery as equivalent to using a different regularizer in (2). For instance, let $\mathcal{R}(\mathbf{v}) = \|\mathcal{T}(\mathbf{v})\|_1 = \mathbf{1}_{n \times 1}^\top |\mathcal{T}(\mathbf{v})|$ be the basic regularizer promoting ℓ_1 sparsity with respect to an orthogonal \mathcal{T} . Upon replacing the constant vector $\mathbf{1}_{n \times 1}$ by an arbitrary nonnegative vector $\boldsymbol{\mu} \in \mathbb{R}^n$, (4) becomes a soft-shrinkage with thresholds $\tau_k[\xi] = \rho_k \gamma \boldsymbol{\mu}[\xi] \|\phi_{\mathcal{T}}^\xi\|_2^2$, which through (22) corresponds to denoising with a non-flat PSD. Further, allowing $\boldsymbol{\mu}$ to change at every iteration based on the current \mathbf{z}_k , i.e. $\boldsymbol{\mu}(\mathbf{z}_k)$, is akin to a reweighted ℓ_1 minimization where $\mathcal{R}(\mathbf{v}) = \boldsymbol{\mu}^\top(\mathbf{v}) |\mathcal{T}(\mathbf{v})|$ is the explicit form of the regularizer for (2) resulting in soft-shrinkage under a data-driven non-flat PSD. Specifically, we want $\boldsymbol{\mu}(\mathbf{z}_k)$ to depend on the stationary random structures within \mathbf{z}_k and hence, as discussed in Section IV-C, in our algorithm we use \mathbf{e}_k instead of \mathbf{z}_k and separate the noise estimation step from the denoising. Analogous observations hold also for $\|\mathcal{T}(\mathbf{v})\|_0 = \mathbf{1}_{n \times 1}^\top \mathbb{1}_{|\mathcal{T}(\mathbf{v})| > 0}$, where $\mathbb{1}_{(\cdot)}$ denotes the indicator of (\cdot) , with $\tau_k[\xi] = \sqrt{2\rho_k \gamma \boldsymbol{\mu}[\xi]} \|\phi_{\mathcal{T}}^\xi\|_2$.

B. Deep Learning Methods for Sparse Signal Recovery

As Φ (8) one can also use a denoiser based on deep learning. However, this filter should have been trained so to be able to deal with the type of correlation met during the recovery. The recovery methods in [38], [39] show that these filters perform well when trained for relatively narrow classes of PSDs (e.g., 13 degrees of freedom). However, the retraining gets more involved and the performance substantially decreased when dealing with much broader families of PSDs or when the filter is meant to be operated as a blind denoiser for correlated noise (see, e.g., the analysis in [58]). In this regard, transform-domain filters in general, and

Table IV
A FEW ALTERNATIVE APPROXIMATIONS OF $\Psi_{\mathbf{e}_k}$

case	approximated $\Psi_{\mathbf{e}_k} \approx$	approximations of other terms	H1	H2
A ₀	$\Psi_{\mathbf{s}_k} + \Psi_{I_k}$ as (14)	$\Psi_{\mathbf{s}_k}$ as (17), Ψ_{I_k} as (20)	exact	exact
A ₁	$\Psi_{\mathbf{s}_k}$	$\Psi_{\mathbf{s}_k}$ as (17)	exact	exact
A ₂	$\Psi_{\mathbf{b}_{k-1}}$ as in [30], [31], [36]	n/a	exact	inexact
A ₃	$\begin{cases} \frac{\Psi_{\mathbf{M}^\dagger \mathbf{M} \mathbf{e}_k^{[\xi]}}}{\mathbf{f}_{\mathbf{M}^\dagger \mathbf{M}^{[\xi]}}} & \text{if } \mathbf{f}_{\mathbf{M}^\dagger \mathbf{M}^{[\xi]}} \neq 0, \\ 0 & \text{otherwise.} \end{cases}$	$\Psi_{\mathbf{M}^\dagger \mathbf{M} \mathbf{e}_k} \approx$ $\Psi_{\mathbf{M}^\dagger (\mathbf{M} \mathbf{z}_k - \mathbf{y})} + \Psi_{\mathbf{M}^\dagger \boldsymbol{\epsilon}}$	inexact	exact
A ₄		$\Psi_{\mathbf{M}^\dagger \mathbf{M} \mathbf{e}_k} \approx$ $(\Psi_{\mathbf{M}^\dagger (\mathbf{M} \mathbf{z}_k - \mathbf{y})}^{1/2} + \Psi_{\mathbf{M}^\dagger \boldsymbol{\epsilon}}^{1/2})^2$	inexact	exact
A ₅	$\Psi_{\mathbf{b}_{k-1}} + \Psi_{\mathbf{x}_{k-1-\mathbf{x}}}$	$\Psi_{\mathbf{M}^\dagger \mathbf{M} (\mathbf{x}_{k-1-\mathbf{x}})} \approx$	exact	inexact
A ₆	$(\Psi_{\mathbf{b}_{k-1}}^{1/2} + \Psi_{\mathbf{x}_{k-1-\mathbf{x}}}^{1/2})^2$	$ \Psi_{\mathbf{M}^\dagger \mathbf{r}_{k-1}}^{1/2} - \Psi_{\mathbf{M}^\dagger \boldsymbol{\epsilon}}^{1/2} $	exact	inexact

specifically the BMxD employed in our manuscript, can work out-of-the-box for arbitrary PSDs and are thus well suited for dealing with the unknown and varying noise correlation that is encountered during the iterative recovery. As an alternative to using a learned deep denoiser within a PnP framework, it is sometimes preferred to employ unrolled algorithms such as the learned ISTA (LISTA) (e.g., [59] for a review) that solve the inverse problem through a neural network where each subsequent layer corresponds to a further iteration of the recovery algorithm. We speculate that their good performance is partly due to the fact that the parameters learned from end-to-end training do implicitly model the filtering of correlated noise across the various iterations, instead of simple AWGN residuals.

C. Alternative Approximations of Effective Noise PSD $\Psi_{\mathbf{e}_k}$

In this paper we use (14), (17) and (20) to approximate $\Psi_{\mathbf{e}_k}$. However, there are alternative ways to approximate $\Psi_{\mathbf{e}_k}$ based on different simplifying assumptions; some of these alternatives are summarized in Table IV. Although these are appealing due to their simplicity, we favor approximations that are exact for $\mathcal{Q}_y(\mathbf{v}) = \frac{1}{2} \|\mathbf{M}^\dagger (\mathbf{y} - \mathbf{M}\mathbf{v})\|_2^2$ in (2) used in the experiments, under any of these two idealistic hypotheses:

H1: $\mathbf{x}_{k-1} = \mathbf{x}$ (ideal recovery),

H2: $\mathbf{y} = \mathbf{x}$ (complete direct measurements, noise-free).

It can be shown that only A₀ and A₁ are exact under each hypothesis. Despite its defects, A₀ leads to better results in most cases and is hence adopted in this work.

D. Estimating Effective Noise PSD $\Psi_{\mathbf{e}_k}$ from \mathbf{z}_k

The above estimation strategies and approximations are all more involved than the direct estimation of $\Psi_{\mathbf{e}_k}$ from the noisy image \mathbf{z}_k . To justify the burden and demonstrate the overestimation occurring when the noise PSD is estimated directly from \mathbf{z}_k (7), we conduct the experiments of Section VI-A with BM4D PnP-ADMM under setting C and C* where $\sigma_{\mathcal{T}_{3D}(\mathbf{e}_k)}$ is now estimated from \mathbf{z}_k (7) using a robust estimator (see Appendix B) as a means to decrease the effect of signal contamination on the estimated PSD. For the set of SNR = {∞, 25, 15} dB, these experiments under C and C* respectively obtain {11.2, 8.6, 5.4} dB and {11.0, 6.0, 4.6} dB lower PSNR compared to their corresponding results in Table I, while $\lambda^* = \{3.1, 2.9, 3.1\}$ in this C* setting (smaller

than $\lambda^{\text{dft}} = 3.7$, revealing overestimation). These significantly inferior results confirm the importance of designing viable PSD estimation strategies for general PnP recovery problems.

E. Wiener filtering

In our experiments, the Wiener stage of BMxDs is not used since it depends more crucially on a precise noise model, and also it does not typically bring improvement to the recovery because it is rather weak as a sparsity-promoting regularizer.

F. Computational Complexity

Denosing correlated noise is slightly more expensive than denosing AWGN, with main differences consisting in the use of nonuniform shrinkage thresholds. For instance, with the BM3D MATLAB/C implementation, the denoising step of the experiments in Section VI-C, takes respectively 0.21 and 0.29 seconds per iteration (Intel Core i7-7700HQ 2.8-GHz CPU) for the $W_1/W_2/W_2^*$ and C/C^* settings, respectively. The adaptive estimation of the PSD of \mathbf{e}_k at every iteration (see Appendix B) has marginal complexity compared to BMxD denoising itself, taking 0.06 seconds (MATLAB).

VIII. CONCLUSIONS

We studied the effective noise \mathbf{e}_k in sparse signal recovery, modeling it as a generic stationary correlated noise, and introduced a comprehensive methodology for its estimation and effective attenuation within established PnP iterative approaches. Through extensive analysis, we have demonstrated that:

1. Significant correlation in \mathbf{e}_k is commonplace.
 2. This correlation can occur also when measurements \mathbf{y} are noise-free, i.e. $\boldsymbol{\epsilon} = \mathbf{0}$, and even under a PnP AWGN denoiser.
 3. Filters for correlated noise can use in PnP a strength parameter λ close to the default value for stand-alone denoising, thus allowing effective recovery without separate tuning of λ .
 4. Modeling \mathbf{e}_k as correlated noise improves recovery quality.
- Overall, we have shown the rationale, feasibility, and practical advantages supporting the adoption of a correlated noise model for \mathbf{e}_k over the common AWGN model.

APPENDIX A

NOISE MODELING IN BMxD FILTERS

A. Stationary Correlated Noise Modeling in BM3D/BM4D

BM3D [43] operates by stacking mutually similar $b \times b$ blocks into 3D arrays called groups. Groups are filtered collaboratively to obtain distinct estimates for each block. Let \mathbf{z}_k^c denote a block extracted from \mathbf{z}_k at position \mathbf{c} , and let \mathbf{Z} denote a generic group formed by stacking h blocks $\{\mathbf{z}_k^c\}_{i=1}^h$. The collaborative filtering of \mathbf{Z} can be defined analogous to (21)-(22) as $\tilde{\mathbf{Z}} = \mathcal{T}_{3D}^{-1}(\Upsilon(\mathcal{T}_{3D}(\mathbf{Z}), \lambda \sigma_{\mathcal{T}_{3D}(\mathbf{Z})}))$, where $\mathcal{T}_{3D} = \mathcal{T}_{2D} \otimes \mathcal{T}_{1D}$ is a separable 3D transform, \otimes denotes the tensor product, and $\sigma_{\mathcal{T}_{3D}(\mathbf{Z})}$ is the \mathcal{T}_{3D} noise root-PSD of \mathbf{Z} . Here \mathcal{T}_{2D} is a spatial transform for $b \times b$ blocks and \mathcal{T}_{1D} is a “nonlocal” 1D transform applied along the stacking dimension.

The 3D array $\tilde{\mathbf{Z}}$ is treated as a stack of estimates of the noise-free blocks, which are aggregated to their original positions $\{\mathbf{c}_i\}_{i=1}^h$ within the image using adaptive weights.

Under the simplifying assumption that noise in a block is independent from the noise in any other block¹¹, and due to the orthonormality of \mathcal{T}_{1D} along the third dimension, the \mathcal{T}_{3D} noise PSD of a group \mathbf{Z} is the stacked replica of the \mathcal{T}_{2D} noise PSD:

$$\begin{aligned} \sigma_{\mathcal{T}_{3D}(\mathbf{Z})}^2[\xi_1, \xi_2, \xi_3] &= \text{var} \{ \mathcal{T}_{3D}(\mathbf{Z}) \} [\xi_1, \xi_2, \xi_3] = \\ &= \text{var} \{ \langle \mathbf{Z}, \phi_{\mathcal{T}_{2D}}^{\xi_1, \xi_2} \otimes \phi_{\mathcal{T}_{1D}}^{\xi_3} \rangle \} = \\ &= \text{var} \{ \mathcal{T}_{2D}(\mathbf{z}_k^{c_i})[\xi_1, \xi_2] \} = \sigma_{\mathcal{T}_{2D}(\mathbf{e}_k)}^2[\xi_1, \xi_2], \end{aligned} \quad (26)$$

where ξ_1, ξ_2 and ξ_3 are, respectively, the two spatial and the nonlocal components in the \mathcal{T}_{3D} domain, $\mathbf{z}_k^{c_i}$ is any of the blocks stacked in \mathbf{Z} , and $\sigma_{\mathcal{T}_{2D}(\mathbf{e}_k)}^2$ denotes the \mathcal{T}_{2D} -PSD of \mathbf{e}_k ¹². The \mathcal{T}_{2D} -PSD $\sigma_{\mathcal{T}_{2D}(\mathbf{e}_k)}^2$ is computed according to (23) as $\sigma_{\mathcal{T}_{2D}(\mathbf{e}_k)}^2[\xi_1, \xi_2] = n^{-2} \|\Psi_{\mathbf{e}_k} \odot \mathcal{F}^2(\phi_{\mathcal{T}_{2D}}^{\xi_1, \xi_2})\|_1$.

BM4D [26] is the extension of BM3D to 3D data. It operates by grouping h mutually similar $b \times b \times d$ cubes of voxels into 4D arrays which are then filtered collaboratively. BM4D models \mathbf{e}_k as stationary noise correlated in all three dimensions, with $\sigma_{\mathcal{T}_{3D}(\mathbf{e}_k)}^2$ denoting the \mathcal{T}_{3D} -PSD of \mathbf{e}_k . Analogous to (26), the \mathcal{T}_{4D} noise PSD of a 4D group \mathbf{Z}

$$\begin{aligned} \sigma_{\mathcal{T}_{4D}(\mathbf{Z})}^2[\xi_1, \xi_2, \xi_3, \xi_4] &= \text{var} \{ \langle \mathbf{Z}, \phi_{\mathcal{T}_{3D}}^{\xi_1, \xi_2, \xi_3} \otimes \phi_{\mathcal{T}_{1D}}^{\xi_4} \rangle \} = \\ &= \sigma_{\mathcal{T}_{3D}(\mathbf{e}_k)}^2[\xi_1, \xi_2, \xi_3], \end{aligned} \quad (27)$$

where ξ_1, ξ_2, ξ_3 are the spatial components and ξ_4 is the non-local component in the separable $\mathcal{T}_{4D} = \mathcal{T}_{3D} \otimes \mathcal{T}_{1D}$ domain¹³.

B. Stationary Correlated Noise Modeling in RF3D

The video denoiser RF3D [44] aggregates a multitude of motion-compensated spatiotemporal volumes which are filtered in a transform domain. A generic spatiotemporal volume \mathbf{Z} is formed by concatenating $b \times b$ blocks extracted from h consecutive frames of \mathbf{z}_k into a $b \times b \times h$ array. The collaborative filtering of \mathbf{Z} via shrinkage in \mathcal{T}_{3D} domain and estimate aggregation are performed analogous to BM3D.

Specific to RF3D is a model of \mathbf{e}_k as a combination of two frame-wise components: a *random noise* $\mathbf{e}_{\text{RND}_k}$ that is independently realized at every frame, and a *fixed-pattern noise* (FPN) $\mathbf{e}_{\text{FPN}_k}$ that is constant in time. Thus, in a set of n_3 frames, there are n_3 independent realizations of $\mathbf{e}_{\text{RND}_k}$ and n_3 copies of a unique realization of $\mathbf{e}_{\text{FPN}_k}$. Both $\mathbf{e}_{\text{RND}_k}$ and $\mathbf{e}_{\text{FPN}_k}$ are zero-mean processes and each features its own spatial correlation with 2D PSDs Ψ_{RND_k} and Ψ_{FPN_k} , respectively. Overall this corresponds to a spatiotemporally correlated noise \mathbf{e}_k whose 3D PSD $\Psi_{\mathbf{e}_k}$ is formed by replicating the 2D Ψ_{RND_k} along the temporal frequencies plus $n_3 \Psi_{\text{FPN}_k}$ over the

¹¹An improved BM3D with exact calculation of $\sigma_{\mathcal{T}_{3D}(\mathbf{Z})}^2$ (i.e. without assuming noise independence from block to block) was recently introduced [60] but is not adopted here so to maintain consistent PSD modeling with BM4D and also enable direct comparison with earlier works.

¹²The slight abuse of notation for $\sigma_{\mathcal{T}_{2D}(\mathbf{e}_k)}^2$ (\mathcal{T}_{2D} is defined for blocks, not for the entire \mathbf{e}_k) follows from noting that, due to the assumed noise stationarity, any block extracted from \mathbf{e}_k is subject to same variances $\text{var} \{ \mathcal{T}_{2D}(\mathbf{z}_k^c) \}$, irrespective of the block position \mathbf{c} .

¹³For the experiments in Section VI-A, ξ_3 represents the spectral component (i.e. wavelengths) of the multispectral image.

temporal-DC plane. The \mathcal{T}_{3D} -PSD of noise in a spatiotemporal volume \mathbf{Z} formed by blocks extracted at positions $\{\mathbf{c}_i\}_{i=1}^h$ from h consecutive frames is then

$$\begin{aligned} \sigma_{\mathcal{T}_{3D}(\mathbf{Z})}^2[\xi_1, \xi_2, \xi_3] &= \sigma_{\text{RND}_k}^2[\xi_1, \xi_2] \\ &+ \alpha(\xi_3, \{\mathbf{c}_i\}_{i=1}^h) \sigma_{\text{FPN}_k}^2[\xi_1, \xi_2], \end{aligned} \quad (28)$$

where $\sigma_{\text{RND}_k}^2$ and $\sigma_{\text{FPN}_k}^2$ are the \mathcal{T}_{2D} -PSDs of $\mathbf{e}_{\text{RND}_k}$ and $\mathbf{e}_{\text{FPN}_k}$ respectively computed from Ψ_{RND_k} and Ψ_{FPN_k} according to (23), and the factor $\alpha > 0$ internally computed by RF3D depends on the temporal frequency ξ_3 and on the spatial alignment of the blocks within the frames [44, Section IV-D] and thus changes adaptively for each volume. Computing $\sigma_{\mathcal{T}_{3D}(\mathbf{Z})}^2$ via a pair of 2D PSDs as in (28) is more efficient than the formal computation with the 3D $\Psi_{\mathbf{e}_k}$ assumed by (23).

C. Simplification under AWGN Modeling in BMxD Filters

When \mathbf{e}_k is modeled as AWGN $\mathbf{e}_k \sim \mathcal{N}(\mathbf{0}, \sigma_{\mathbf{e}_k}^2 \mathbf{I}_n)$, (26) for BM3D and (27) for BM4D reduce respectively to

$$\sigma_{\mathcal{T}_{3D}(\mathbf{Z})}^2[\xi_1, \xi_2, \xi_3] = \sigma_{\mathbf{e}_k}^2 \|\phi_{\mathcal{T}_{3D}}^{\xi_1, \xi_2}\|_2^2, \quad (29)$$

$$\sigma_{\mathcal{T}_{4D}(\mathbf{Z})}^2[\xi_1, \xi_2, \xi_3, \xi_4] = \sigma_{\mathbf{e}_k}^2 \|\phi_{\mathcal{T}_{3D}}^{\xi_1, \xi_2, \xi_3}\|_2^2. \quad (30)$$

Under the AWGN model $\mathbf{e}_k \sim \mathcal{N}(\mathbf{0}, \sigma_{\mathbf{e}_k}^2 \mathbf{I}_n)$ there can be no FPN and thus for RF3D $\sigma_{\text{FPN}_k} = \mathbf{0}$ and (28) reduces to (29).

When $\|\phi_{\mathcal{T}}^{\xi}\|_2 = 1$, which holds for all BMxD transforms in this work, both (29) and (30) further simplify to a constant independent of ξ : $\sigma_{\mathcal{T}(\mathbf{Z})}^2 \equiv \sigma_{\mathbf{e}_k}^2$.

APPENDIX B ADAPTIVE NOISE PSD ESTIMATION IN SPARSE SIGNAL RECOVERY

The local FFT-PSD of a stationary noise can be estimated as the sample variance of its local FFT spectrum computed over a moving window, as each position of the window can be treated as a different realization of the noise. Specifically, we first separately estimate the local FFT-PSDs of \mathbf{w}_k and $\mathbf{M}^\dagger \epsilon$, which following (17) give the local FFT-PSD of \mathbf{s}_k . Even though $\mathbf{M}^\dagger \epsilon$ is unavailable, we can generate a noise $\hat{\epsilon}$ with the same statistics as ϵ , and then apply the moving window onto $\mathbf{M}^\dagger \hat{\epsilon}$ ¹⁴. To estimate the local FFT-PSD of \mathbf{l}_k , the local versions of $\mathbf{f}_{\mathbf{M}^\dagger \mathbf{M}}$ and $\mathbf{f}_{\mathbf{M}_\perp}$ should be computed. To this end, we can compute the local FFT-PSDs of $\mathbf{M}^\dagger \mathbf{M} \boldsymbol{\eta}$ and $\mathbf{M}_\perp \boldsymbol{\eta}$ for a standard AWGN $\boldsymbol{\eta}$, which upon division by the size of the window respectively coincide with local versions of $\mathbf{f}_{\mathbf{M}^\dagger \mathbf{M}}$ and $\mathbf{f}_{\mathbf{M}_\perp}$. These, together with the local FFT-PSDs of $\mathbf{M}^\dagger \mathbf{r}_{k-1}$ and $\mathbf{M}^\dagger \epsilon$, give the local FFT-PSD of \mathbf{l}_k as per (20). Finally, by (14), we obtain the local FFT-PSD of \mathbf{e}_k , which is then upsampled to a global FFT-PSD, constituting an estimate $\hat{\Psi}_{\mathbf{e}_k}$ of $\Psi_{\mathbf{e}_k}$. For the upsampling, we use the constrained nonnegative least-squares method [40, Section 6.3.2], enforcing smoothness and symmetries of Fourier PSDs.

¹⁴As an alternative to the moving window method, for an AWGN ϵ with variance σ_{ϵ}^2 , each entry of the local FFT-PSD of $\mathbf{M}^\dagger \epsilon$ can also be computed as $n^{-2} \|\Psi_{\mathbf{M}^\dagger \epsilon} \odot \mathcal{F}^2(\phi_{\mathcal{T}}^{\xi})\|_1$ with \mathcal{T} being the small size FFT and $\Psi_{\mathbf{M}^\dagger \epsilon}[\xi] = \text{var} \{ \langle \mathbf{M}^\dagger \epsilon, \phi_{\mathcal{T}}^{\xi} \rangle \} = \sigma_{\epsilon}^2 \|\phi_{\mathcal{T}}^{\xi}\|_2^2$. The local versions of $\mathbf{f}_{\mathbf{M}^\dagger \mathbf{M}}$ and $\mathbf{f}_{\mathbf{M}_\perp}$ can be computed similarly.

This approach is used directly for the experiments in Section VI-C with BM3D, where the 2D PSD $\Psi_{\mathbf{e}_k}$ is estimated starting from 2D moving-window analyses of 2D \mathbf{w}_k , $\mathbf{M}^\dagger \boldsymbol{\epsilon}$, $\mathbf{M}^\dagger \mathbf{r}_{k-1}$ and, likewise, for the experiment in Sections VI-A and VI-D with BM4D, where PSDs and moving windows are 3D. For the video-recovery experiments in Section VI-B with RF3D, we take advantage of the special structure assumed by the 3D PSD $\Psi_{\mathbf{e}_k}$ and estimate instead its frame-wise 2D components Ψ_{RND_k} and Ψ_{FPN_k} . Since $\mathbf{e}_{\text{FPN}_k}$ is temporally invariant and $\mathbf{e}_{\text{RND}_k}$ is zero-mean, we can estimate them as $\widehat{\mathbf{e}}_{\text{FPN}_k} = \overline{\mathbf{e}_k}$ and $\widehat{\mathbf{e}}_{\text{RND}_k} = \mathbf{e}_k - \widehat{\mathbf{e}}_{\text{FPN}_k}$, where the decoration $\overline{\cdot}$ stands for the temporal mean over all frames in the video. The same separation carries over to $\mathbf{M}^\dagger \boldsymbol{\epsilon}$, \mathbf{w}_k , and $\mathbf{M}^\dagger \mathbf{r}_{k-1}$, which can be split into their respective FPN subcomponents (i.e. $\overline{\mathbf{M}^\dagger \boldsymbol{\epsilon}}$, $\overline{\mathbf{w}_k}$, $\overline{\mathbf{M}^\dagger \mathbf{r}_{k-1}}$) and ‘‘RND’’ subcomponents (i.e. $\mathbf{M}^\dagger \boldsymbol{\epsilon} - \overline{\mathbf{M}^\dagger \boldsymbol{\epsilon}}$, $\mathbf{w}_k - \overline{\mathbf{w}_k}$, $\mathbf{M}^\dagger \mathbf{r}_{k-1} - \overline{\mathbf{M}^\dagger \mathbf{r}_{k-1}}$). We estimate the 2D FFT-PSDs of each of these six subcomponents via a local 2D FFT analysis as above, which upon combination like in (17), (20) and (14), and subsequent upsampling, gives $\widehat{\Psi}_{\text{RND}_k}$ and $\widehat{\Psi}_{\text{FPN}_k}$.

We note that using a small moving window can limit the frequency resolution of the PSD, while a bigger moving window yields fewer samples for PSD estimation. We used moving windows of size $31 \times 31 \times 13$ for the CAVE multispectral images in Section VI-A, 31×31 for the *NBA* and *water balloon* sequences in Section VI-B, 21×18 for the 2D *Brainweb* cross-sections in Section VI-C, and $15 \times 15 \times 15$ for the 3D *Brainweb* volume in Section VI-D. We remark that the local FFT-PSDs corresponding to these moving windows, while of course smaller than the global PSDs, have nevertheless orders of magnitude more degrees of freedom than the simple representations of noise correlation through the variances of individual wavelet subbands [32], [36]–[39].

The above local definition of $\mathbf{f}_{\mathbf{M}^\dagger \mathbf{M}}$ can also improve the conditioning in the definition (20). In particular, for the experiments with binary sampling of the Fourier spectrum in Sections VI-C and VI-D, the proportion of zeros of $\mathbf{f}_{\mathbf{M}^\dagger \mathbf{M}}$ is substantially lower for its local version, which leads to a better approximation of $\Psi_{\mathbf{e}_k}$.

For the comparative illustrations of root-PSDs in Figures 3, 5, 9, 12 and 13, we compute ‘‘oracle’’ $\Psi_{\mathbf{e}_k}$ by upsampling the local FFT-PSD of a single realization of the oracle effective noise $\mathbf{e}_k = \mathbf{z}_k - \mathbf{x}$, computed too over a moving window.

To estimate $\Psi_{\mathbf{e}_k}$ from the noisy image \mathbf{z}_k (8) in Section VII-D, we apply the sample MAD [41] over the local FFT-spectra of \mathbf{z}_k computed over a moving window, followed by upsampling.

APPENDIX C

TECHNICAL DETAILS OF THE EXPERIMENTS

A. Compressive Spectral Imaging (Section VI-A)

In (1), \mathbf{x} represents the multispectral image vectorized starting from the spectral dimension, \mathbf{M} is a $m \times 16m$ block-diagonal matrix, where each block is a binary row vector of length 16 (number of bands) extracted from the pseudo-random shifting mask in reverse direction to its shifting, and \mathbf{y} are spectrally compressed measurements.

For $\mathcal{Q}_{\mathbf{y}}(\mathbf{v}) = \frac{1}{2} \|\mathbf{M}^\dagger(\mathbf{y} - \mathbf{M}\mathbf{v})\|_2^2$, (3b) is solved as

$$\mathbf{x}_k = \frac{\rho_k}{\rho_k + 1} \mathbf{M}^\dagger \mathbf{y} + \left(\mathbf{I}_n - \frac{\rho_k}{\rho_k + 1} \mathbf{M}^\dagger \mathbf{M} \right) (\mathbf{u}_k - \mathbf{b}_{k-1}). \quad (31)$$

GPSR-BB solves (2) for an analysis-based ℓ_1 regularization with $\mathcal{Q}_{\mathbf{y}}(\mathbf{v}) = \frac{1}{2} \|\mathbf{y} - \mathbf{M}\mathbf{v}\|_2^2$ and $\mathcal{R}(\mathbf{v}) = \|\mathcal{T}(\mathbf{v})\|_1$, where \mathcal{T} is the 3D separable composition of 2D Symlet-8 wavelet transform for the spatial dimensions and 1D DCT along the spectral dimension [52]. TwIST is based on a similar formulation, with the gradient operator instead of \mathcal{T} , thus solving (2) for the total-variation regularizer. Similar to \mathbf{W}_2^* , GPSR-BB and TwIST include an explicit parameter γ from (2) that was separately optimized at $k_{\text{final}} = 3000$.

B. Compressive Temporal Imaging (Section VI-B)

Here \mathbf{x} is the concatenation of all consecutive video chunks, where each chunk is vectorized first in time and then in space, \mathbf{M} is a $m \times 8m$ block-diagonal matrix comprised of 4 (number of chunks) identical $\frac{m}{4} \times \frac{8m}{4}$ block-diagonal matrices, where each block is a binary row vector of length 8 (number of frames per chunk) extracted from the mask in reverse direction to its shifting, and \mathbf{y} are the temporally compressed measurements. The solution of (3b) is similar to (31).

MMLE-GMM [53] maximizes the likelihood of the Gaussian mixture model of \mathbf{x} given \mathbf{y} .

For the *water balloon* data, \mathbf{M} is formalized as a $m \times 10m$ block-diagonal matrix where each block is a row vector of length 10 (number of frames), providing a single snapshot measurement \mathbf{y} shown in Figure 10. Since the ground-truth video is unavailable, the tuning of parameters ρ^* , γ^* , and λ^* (where applicable) is made for the recovery of a test video from simulated acquisition by the same \mathbf{M} subject to synthetic AWGN $\boldsymbol{\epsilon}$ with variance $\widehat{\sigma}_{\boldsymbol{\epsilon}}^2$.

C. 2D Tomography (Section VI-C)

Let \mathbf{F} be the forward FFT matrix with $\mathbf{F}^\dagger \mathbf{F} \mathbf{x} = \mathbf{x}$, then in (1) $\mathbf{M} = \mathbf{D}\mathbf{F}$ where \mathbf{D} is a downsampling matrix composed of m rows of \mathbf{I}_n and \mathbf{y} are k -space measurements of the vectorized image \mathbf{x} . For $\mathcal{Q}_{\mathbf{y}}(\mathbf{v}) = \frac{1}{2} \|\mathbf{M}^\dagger(\mathbf{y} - \mathbf{M}\mathbf{v})\|_2^2$, (3b) is solved as

$$\mathbf{x}_k = \mathbf{F}^\dagger \left((\mathbf{D}^\dagger \mathbf{D} + \rho_k^{-1} \mathbf{I}_n)^{-1} (\mathbf{D}^\dagger \mathbf{y} + \rho_k^{-1} \mathbf{F}(\mathbf{u}_k - \mathbf{b}_{k-1})) \right).$$

Nonnegativity is then enforced by updating $\mathbf{x}_k \mapsto \max(\mathbf{x}_k, \mathbf{0})$.

D. 3D Tomography (Section VI-D)

Here (1) is analogous to the one above, only with \mathbf{x} a vectorized 3D volumetric image. We zero-pad the $181 \times 217 \times 181$ BrainWeb phantom [56] to $217 \times 217 \times 217$ and then trilinearly resize it into a $64 \times 64 \times 64$ magnitude.

ACKNOWLEDGMENT

We thank Mario Figueiredo, Brendt Wohlberg, and the late Jose Bioucas-Dias for the fruitful discussions and encouragement during the development of this work. We also thank the Associate Editor and the five anonymous reviewers for the many insightful and constructive suggestions that helped us improve the paper. This work was supported by the European Union (H2020-MSCA-ITN-2014, grant agreement no. 642685 ‘‘MacSeNet’’) and by the Academy of Finland (projects 310779 and 326439).

REFERENCES

- [1] D. L. Donoho, "Compressed sensing," *IEEE Trans. Inf. Theory*, vol. 52, no. 4, pp. 1289–1306, 2006.
- [2] E. J. Candès, J. K. Romberg, and T. Tao, "Stable signal recovery from incomplete and inaccurate measurements," *Comm. Pure Appl. Math.*, vol. 59, no. 8, pp. 1207–1223, 2006.
- [3] E. Candès and J. Romberg, "Sparsity and incoherence in compressive sampling," *Inv. Prob.*, vol. 23, no. 3, p. 969, 2007.
- [4] 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, 2004.
- [5] P. L. Combettes and V. R. Wajs, "Signal recovery by proximal forward-backward splitting," *Multiscale Model. Simul.*, vol. 4, no. 4, pp. 1168–1200, 2005.
- [6] 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.
- [7] M. A. Figueiredo, R. D. Nowak, and S. J. Wright, "Gradient projection for sparse reconstruction: Application to compressed sensing and other inverse problems," *IEEE J. Sel. Top. Sig. Process.*, vol. 1, no. 4, pp. 586–597, 2007.
- [8] J. M. Bioucas-Dias and M. A. Figueiredo, "A new TwIST: two-step iterative shrinkage/thresholding algorithms for image restoration," *IEEE Trans. Image Processing*, vol. 16, no. 12, pp. 2992–3004, 2007.
- [9] S. J. Wright, R. D. Nowak, and M. A. Figueiredo, "Sparse reconstruction by separable approximation," *IEEE Trans. Sig. Process.*, vol. 57, no. 7, pp. 2479–2493, 2009.
- [10] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, "Distributed optimization and statistical learning via the alternating direction method of multipliers," *Found. Trends Mach. Learn.*, vol. 3, no. 1, pp. 1–122, 2011.
- [11] D. L. Donoho, A. Maleki, and A. Montanari, "Message-passing algorithms for compressed sensing," *Proc. Nat. Acad. Sci.*, vol. 106, no. 45, pp. 18914–18919, 2009.
- [12] O. G. Guleryuz, "Nonlinear approximation based image recovery using adaptive sparse reconstructions and iterated denoising-part I: theory," *IEEE Trans. Image Process.*, vol. 15, no. 3, pp. 539–554, 2006.
- [13] K. Egiazarian, A. Foi, and V. Katkovnik, "Compressed sensing image reconstruction via recursive spatially adaptive filtering," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, 2007, pp. 549–552.
- [14] X. Li, "Image recovery via hybrid sparse representations: A deterministic annealing approach," *IEEE J. Sel. Top. Sig. Process.*, vol. 5, no. 5, pp. 953–962, 2011.
- [15] S. Venkatakrisnan, C. Bouman, and B. Wohlberg, "Plug-and-play priors for model based reconstruction," in *Proc. IEEE Global Conf. Sig. Inf. Process. (GlobalSIP)*, 2013, pp. 945–948.
- [16] C. Metzler, A. Maleki, and R. Baraniuk, "From denoising to compressed sensing," *IEEE Trans. Inf. Theory*, vol. 62, no. 9, pp. 5117–5144, 2016.
- [17] S. H. Chan, X. Wang, and O. A. Elgandy, "Plug-and-play ADMM for image restoration: fixed-point convergence and applications," *IEEE Trans. Comput. Imag.*, vol. 3, no. 1, pp. 84–98, 2017.
- [18] G. T. Buzzard, S. H. Chan, S. Sreehari, and C. A. Bouman, "Plug-and-play unplugged: Optimization-free reconstruction using consensus equilibrium," *SIAM J. Imaging Sci.*, vol. 11, no. 3, pp. 2001–2020, 2018.
- [19] Y. Sun, B. Wohlberg, and U. S. Kamilov, "An online plug-and-play algorithm for regularized image reconstruction," *IEEE Trans. Comput. Imag.*, vol. 5, no. 3, pp. 395–408, 2019.
- [20] E. Ryu *et al.*, "Plug-and-play methods provably converge with properly trained denoisers," in *Proc. Int. Conf. Mach. Learn. (ICML)*, 2019.
- [21] X. Xu, Y. Sun, J. Liu, B. Wohlberg, and U. S. Kamilov, "Provable convergence of plug-and-play priors with MMSE denoisers," *IEEE Sig. Process. Lett.*, vol. 27, pp. 1280–1284, 2020.
- [22] T. Tirer and R. Giryes, "Back-projection based fidelity term for ill-posed linear inverse problems," *IEEE Trans. Image Process.*, vol. 29, pp. 6164–6179, 2020.
- [23] Y. Romano, M. Elad, and P. Milanfar, "The little engine that could: Regularization by denoising (RED)," *SIAM J. Imaging Sci.*, vol. 10, no. 4, pp. 1804–1844, 2017.
- [24] B. Wen, Y. Li, and Y. Bresler, "Image recovery via transform learning and low-rank modeling: The power of complementary regularizers," *IEEE Trans. Image Process.*, vol. 29, pp. 5310–5323, 2020.
- [25] S. Sreehari *et al.*, "Plug-and-play priors for bright field electron tomography and sparse interpolation," *IEEE Trans. Comput. Imag.*, vol. 2, no. 4, pp. 408–423, 2016.
- [26] M. Maggioni, V. Katkovnik, K. Egiazarian, and A. Foi, "Nonlocal transform-domain filter for volumetric data denoising and reconstruction," *IEEE Trans. Image Process.*, vol. 22, no. 1, pp. 119–133, 2013.
- [27] R. Ahmad *et al.*, "Plug-and-play methods for magnetic resonance imaging: Using denoisers for image recovery," *IEEE Sig. Process. Mag.*, vol. 37, no. 1, pp. 105–116, 2020.
- [28] X. Yuan, D. J. Brady, and A. K. Katsaggelos, "Snapshot compressive imaging: Theory, algorithms, and applications," *IEEE Sig. Process. Mag.*, vol. 38, no. 2, pp. 65–88, 2021.
- [29] J. Tan, Y. Ma, H. Rueda, D. Baron, and G. R. Arce, "Compressive hyperspectral imaging via approximate message passing," *IEEE J. Sel. Top. Sig. Process.*, vol. 10, no. 2, pp. 389–401, 2016.
- [30] N. Eslahi, V. Ramakrishnan, K. Wiik, and A. Foi, "Sparse signal recovery via correlated degradation modeling," in *Proc. Sig. Process. Adapt. Sparse Struct. Repr. Workshop (SPARS)*, 2017.
- [31] N. Eslahi and A. Foi, "Anisotropic spatiotemporal regularization in compressive video recovery by adaptively modeling the residual errors as correlated noise," in *Proc. IEEE Image Video Multidim. Sig. Process. Workshop (IVMSP)*, 2018.
- [32] N. Eslahi and A. Aghagholzadeh, "Compressive sensing image restoration using adaptive curvelet thresholding and nonlocal sparse regularization," *IEEE Trans. Image Process.*, vol. 25, no. 7, pp. 3126–3140, 2016.
- [33] D. L. Donoho and I. M. Johnstone, "Ideal spatial adaptation by wavelet shrinkage," *Biometrika*, vol. 81, no. 3, pp. 425–455, 1994.
- [34] P. L. Combettes and J.-C. Pesquet, "Proximal splitting methods in signal processing," in *Fixed-point algorithms for inverse problems in science and engineering*. Springer, 2011, pp. 185–212.
- [35] U. S. Kamilov, H. Mansour, and B. Wohlberg, "A plug-and-play priors approach for solving nonlinear imaging inverse problems," *IEEE Sig. Process. Lett.*, vol. 24, no. 12, pp. 1872–1876, 2017.
- [36] H. Garsden *et al.*, "LOFAR sparse image reconstruction," *Astron. Astrophys.*, vol. 575, p. A90, 2015.
- [37] C. Millard, A. T. Hess, B. Mailhé, and J. Tanner, "Approximate message passing with a colored aliasing model for variable density Fourier sampled images," *IEEE Open J. Signal Process.*, vol. 1, pp. 146–158, 2020.
- [38] C. A. Metzler and G. Wetzstein, "D-VDAMP: Denoising-based approximate message passing for compressive MRI," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP)*, 2021, pp. 1410–1414.
- [39] S. K. Shastri, R. Ahmad, C. A. Metzler, and P. Schniter, "Denoising generalized expectation-consistent approximation for MR image recovery," *IEEE J. Sel. Areas Inf. Theory*, 2022.
- [40] L. Azzari, L. R. Borges, and A. Foi, "Modeling and estimation of signal-dependent and correlated noise," in *Denoising of Photographic Images and Video*, M. Bertalmío, Ed. Springer, 2018, pp. 1–36.
- [41] F. R. Hampel, "The influence curve and its role in robust estimation," *J. Am. Stat. Assoc.*, vol. 69, no. 346, pp. 383–393, 1974.
- [42] A. Foi, "Clipped noisy images: Heteroskedastic modeling and practical denoising," *Sig. Process.*, vol. 89, no. 12, pp. 2609–2629, 2009.
- [43] K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian, "Image denoising by sparse 3-D transform-domain collaborative filtering," *IEEE Trans. Image Process.*, vol. 16, no. 8, pp. 2080–2095, 2007.
- [44] M. Maggioni, E. Sánchez-Monge, and A. Foi, "Joint removal of random and fixed-pattern noise through spatiotemporal video filtering," *IEEE Trans. Image Process.*, vol. 23, no. 10, pp. 4282–4296, 2014.
- [45] D. L. Donoho and I. M. Johnstone, "Adapting to unknown smoothness via wavelet shrinkage," *J. Am. Stat. Assoc.*, vol. 90, no. 432, pp. 1200–1224, 1995.
- [46] I. M. Johnstone and B. W. Silverman, "Empirical Bayes selection of wavelet thresholds," *Ann. Stat.*, pp. 1700–1752, 2005.
- [47] F. Abramovich, Y. Benjamini, D. L. Donoho, and I. M. Johnstone, "Adapting to unknown sparsity by controlling the false discovery rate," *Ann. Stat.*, vol. 34, no. 2, pp. 584–653, 2006.
- [48] I. M. Johnstone and B. W. Silverman, "Wavelet threshold estimators for data with correlated noise," *J. Royal Stat. Soc. B (Meth.)*, vol. 59, no. 2, pp. 319–351, 1997.
- [49] D. L. Donoho and I. M. Johnstone, "Minimax estimation via wavelet shrinkage," *Ann. Stat.*, vol. 26, no. 3, pp. 879–921, 1998.
- [50] F. Yasuma, T. Mitsunaga, D. Iso, and S. K. Nayar, "CAVE multispectral image database," <http://www.cs.columbia.edu/CAVE/databases/multispectral/>, 2008.
- [51] A. Wagadarikar, R. John, R. Willett, and D. Brady, "Single disperser design for coded aperture snapshot spectral imaging," *Appl. Opt.*, vol. 47, no. 10, pp. B44–B51, 2008.
- [52] G. R. Arce, D. J. Brady, L. Carin, H. Arguello, and D. S. Kittle, "Compressive coded aperture spectral imaging: An introduction," *IEEE Sig. Process. Mag.*, vol. 31, no. 1, pp. 105–115, 2013.
- [53] J. Yang *et al.*, "Compressive sensing by learning a Gaussian mixture model from measurements," *IEEE Trans. Image Process.*,

- vol. 24, no. 1, pp. 106–119, 2015. [Online]. Available: <https://github.com/jianboyang/MMLE-GMM>
- [54] P. Llull *et al.*, “Coded aperture compressive temporal imaging,” *Opt. Expr.*, vol. 21, no. 9, pp. 10 526–10 545, 2013.
- [55] M. Qiao, Z. Meng, J. Ma, and X. Yuan, “Deep learning for video compressive sensing,” *Apl Photonics*, vol. 5, no. 3, p. 030801, 2020. [Online]. Available: <https://github.com/mq0829/DL-CACTI>
- [56] R. Vincent. (2006) Brainweb: Simulated brain database. [Online]. Available: <http://brainweb.bic.mni.mcgill.ca/brainweb/>
- [57] E. J. Candès, J. Romberg, and T. Tao, “Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information,” *IEEE Trans. Inf. Theory*, vol. 52, no. 2, pp. 489–509, 2006.
- [58] Y. Mäkinen, “Exact transform domain variances for collaborative filtering of correlated noise,” Ph.D. dissertation, Tampere University, 2021.
- [59] V. Monga, Y. Li, and Y. C. Eldar, “Algorithm unrolling: Interpretable, efficient deep learning for signal and image processing,” *IEEE Sig. Process. Mag.*, vol. 38, no. 2, pp. 18–44, 2021.
- [60] Y. Mäkinen, L. Azzari, and A. Foi, “Collaborative filtering of correlated noise: Exact transform-domain variance for improved shrinkage and patch matching,” *IEEE Trans. Image Process.*, vol. 29, pp. 8339–8354, 2020.