An Online Plug-and-Play Algorithm for Regularized Image Reconstruction

Yu Sun, *Student Member, IEEE*, Brendt Wohlberg, *Senior Member, IEEE*, and Ulugbek S. Kamilov, *Member, IEEE*

***Abstract*—Plug-and-play priors (PnP) is a powerful framework for regularizing imaging inverse problems by using advanced denoisers within an iterative algorithm. Recent experimental evidence suggests that PnP algorithms achieve state-of-the-art performance in a range of imaging applications. In this paper, we introduce a new online PnP algorithm based on the proximal gradient method (PGM). The proposed algorithm uses only a subset of measurements at every iteration, which makes it scalable to very large datasets. We present a new theoretical convergence analysis, for both batch and online variants of PnP-PGM, for denoisers that do not necessarily correspond to proximal operators. We also present simulations illustrating the applicability of the algorithm to image reconstruction in diffraction tomography. The results in this paper have the potential to expand the applicability of the PnP framework to very large datasets.**

***Index Terms*—Regularized image reconstruction, plug-and-play priors, regularization by denoising, iterative thresholding, alternating minimization, stochastic optimization.**

I. Introduction

The reconstruction of an unknown image $\boldsymbol{x} \in \mathbb{R}^n$ from a set of noisy measurements $\boldsymbol{y} \in \mathbb{R}^m$ is one of the most widely studied problems in computational imaging. The task is frequently formulated as an optimization problem

$$\widehat{\boldsymbol{x}} = \underset{\boldsymbol{x}\in\mathbb{R}^N}{\operatorname{arg\,min}} \{f(\boldsymbol{x})\} \quad \text{with} \quad f(\boldsymbol{x}) = d(\boldsymbol{x}) + r(\boldsymbol{x}), \tag{1}$$

where d is the data-fidelity term that penalizes the mismatch to the measurements and r is the regularizer that imposes prior knowledge regarding the unknown image. Some popular imaging priors include nonnegativity, transform-domain sparsity, and self-similarity [1]–[4].

Over the past two decades, a substantial effort has been devoted to combining the best regularizers with efficient optimization algorithms. The large dimensionality of the imaging data and nondifferentiability of many regularizers has led to the widespread adoption of proximal algorithms [5], such as variants of proximal gradient method (PGM) [6]–[9] and alternating direction method of multipliers (ADMM) [10]–[13]. These algorithms avoid differentiating the regularizer by using a mathematical concept known as the proximal operator, which is itself an optimization problem equivalent to regularized image denoising.

The mathematical equivalence of the proximal operator to denoising has recently inspired Venkatakrishnan *et al.* [14] to introduce the powerful plug-and-play priors (PnP) framework for image reconstruction. The key idea in PnP is to replace the proximal operator in an iterative algorithm with a state-of-the-art image denoiser, such as BM3D [15], WNNM [16], or TNRD [17], which does not necessarily have a corresponding regularization objective. This implies that PnP methods generally lose interpretability as optimization problems. Nonetheless, the framework has gained in popularity due to its effectiveness in a range of applications in the context of imaging inverse problems [18]–[26]. In particular, the effectiveness of PnP was demonstrated beyond the original ADMM formulation [14] to other proximal algorithms such as primal-dual splitting and PGM [24]–[26].

All current PnP algorithms are iterative *batch* procedures, which means that they use the full set of measurements at every iteration. This effectively precludes their application to very large datasets [27] common in three-dimensional (3D) imaging or in imaging of dynamic objects [28], [29]. In this paper, we address this limitation by proposing a new *online* extension called *plug-and-play stochastic proximal gradient method (PnP-SPGM)*. By using only a subset of the measurements at a time, the proposed algorithm scales to datasets that would otherwise be prohibitively large for batch processing. More specifically, our key contributions are as follows.

- We present a detailed theoretical convergence analysis of batch PnP-PGM under a set of explicit assumptions. Our analysis complements the recent theoretical results on PnP-ADMM by Sreehari *et al.* [18] and Chan *et al.* [19] in two major ways. We show that for PnP-PGM the symmetric gradient assumption from [18] is not necessary, while the bounded denoiser assumption from [19] is not sufficient to establish the convergence.
- We extend the traditional batch PnP framework with our novel online algorithm PnP-SPGM. We prove the theoretical convergence of the algorithm to the same set of fixed points as batch PnP-PGM and PnP-ADMM. This makes PnP-SPGM a powerful and theoretically sound alternative for large-scale image reconstruction. We also illustrate its applicability with several numerical simulations on image reconstruction problems encountered in diffraction tomography [30].

This material is based upon work supported by the National Science Foundation under Grant No. 1813910. *(Corresponding author: Ulugbek S. Kamilov.)*

Y. Sun is with the Department of Computer Science & Enginnering, Washington University in St. Louis, MO 63130, USA.

B. Wohlberg is with Theoretical Division, Los Alamos National Laboratory, Los Alamos, NM 87545 USA.

U. S. Kamilov (email: kamilov@wustl.edu) is with the Department of Computer Science & Engineering and the Department of Electrical & Systems Engineering, Washington University in St. Louis, MO 63130, USA.

Algorithm 1 PGM/APGM

input: $\boldsymbol{x}^0 = \boldsymbol{s}^0 \in \mathbb{R}^n$, $\gamma > 0$, and $\{q_k\}_{k\in\mathbb{N}}$
for $k = 1, 2, \ldots$ **do**
  $\boldsymbol{z}^k \leftarrow \boldsymbol{s}^{k-1} - \gamma \nabla d(\boldsymbol{s}^{k-1})$
  $\boldsymbol{x}^k \leftarrow \mathsf{prox}_{\gamma r}(\boldsymbol{z}^k)$
  $\boldsymbol{s}^k \leftarrow \boldsymbol{x}^k + ((q_{k-1}-1)/q_k)(\boldsymbol{x}^k - \boldsymbol{x}^{k-1})$
end for

Algorithm 2 ADMM

input: $\boldsymbol{x}^0 \in \mathbb{R}^n$, $\boldsymbol{s}^0 = \boldsymbol{0}$, and $\gamma > 0$
for $k = 1, 2, \ldots$ **do**
  $\boldsymbol{z}^k \leftarrow \mathsf{prox}_{\gamma d}(\boldsymbol{x}^{k-1} - \boldsymbol{s}^{k-1})$
  $\boldsymbol{x}^k \leftarrow \mathsf{prox}_{\gamma r}(\boldsymbol{z}^k + \boldsymbol{s}^{k-1})$
  $\boldsymbol{s}^k \leftarrow \boldsymbol{s}^{k-1} + (\boldsymbol{z}^k - \boldsymbol{x}^k)$
end for

II. Background

In this section, we provide the background material that forms the foundation to our contributions. We first review the problem of regularized image reconstruction and then introduce more recent results related to the PnP algorithms.

A. Inverse problems in imaging

Consider the linear inverse problem

$$\boldsymbol{y} = \boldsymbol{H}\boldsymbol{x} + \boldsymbol{e}, \tag{2}$$

where the goal is to recover $\boldsymbol{x} \in \mathbb{R}^n$ given the measurements $\boldsymbol{y} \in \mathbb{R}^m$. Here, the measurement matrix $\boldsymbol{H} \in \mathbb{R}^{m\times n}$ models the response of the imaging system and $\boldsymbol{e} \in \mathbb{R}^m$ represents the measurement noise, which is often assumed to be independent and identically distributed (i.i.d.) Gaussian. When the inverse problem is nonlinear, the measurement operator can be generalized to a more general mapping $\boldsymbol{H} : \mathbb{R}^n \to \mathbb{R}^m$ with $\boldsymbol{y} = \boldsymbol{H}(\boldsymbol{x}) + \boldsymbol{e}$.

Practical inverse problems are often ill-posed, which often leads to the formulation in (1). In such cases, one of the most popular data-fidelity terms is least-squares

$$d(\boldsymbol{x}) = \frac{1}{2}\|\boldsymbol{y} - \boldsymbol{H}\boldsymbol{x}\|_2^2, \tag{3}$$

which imposes an ℓ_2-penalty on data-fit. Similarly, two common regularizers for images include the spatial sparsity-promoting penalty $r(\boldsymbol{x}) \triangleq \lambda\|\boldsymbol{x}\|_1$ and total variation (TV) penalty $r(\boldsymbol{x}) \triangleq \lambda\|\boldsymbol{D}\boldsymbol{x}\|_1$, where $\lambda > 0$ is the regularization parameter and $\boldsymbol{D}$ is the discrete gradient operator [1], [31]–[33].

Many popular regularizers, such as the ones based on the ℓ_1-norm, are nondifferentiable. Two common algorithms for working with such regularizers are PGM and ADMM summarized in Algorithm 1 and 2, respectively (see Appendix B for a review). The key step for handling nonsmooth regularizers is the proximal operator [34]

$$\mathsf{prox}_{\gamma r}(\boldsymbol{z}) \triangleq \operatorname*{arg\,min}_{\boldsymbol{x}\in\mathbb{R}^n} \left\{ \frac{1}{2}\|\boldsymbol{x} - \boldsymbol{z}\|_2^2 + \gamma r(\boldsymbol{x}) \right\}. \tag{4}$$

According to definition (4), the proximal operator corresponds to an image denoiser formulated as regularized optimization. Note also that when the values for $\{q_k\}$ in Algorithm 1 are adapted as

$$q_k \leftarrow \frac{1}{2}\left(1 + \sqrt{1 + 4q_{k-1}^2}\right) \tag{5}$$

the algorithm corresponds to the accelerated variant of PGM, known as accelerated PGM (APGM) [35]. On the other hand, when $q_k = 1$ for all $k \geq 1$, then one recovers the traditional PGM. In this paper, we will use the sequence $\{q_k\}$ as a mechanism for switching between the methods.

A careful inspection of PGM and ADMM reveals a fundamental conceptual difference between the algorithms in their treatment of the data-fidelity. While PGM relies on the gradient ∇d, ADMM relies on the proximal operator $\mathsf{prox}_{\gamma d}$. For a large class of linear and nonlinear inverse problems, the gradient of the data-fidelity is significantly easier to evaluate than its proximal operator. As an example, for least-squares we have

$$\nabla d(\boldsymbol{x}) = \boldsymbol{H}^{\mathsf{T}}(\boldsymbol{H}\boldsymbol{x} - \boldsymbol{y}) \tag{6}$$

and

$$\mathsf{prox}_{\gamma d}(\boldsymbol{x}) = \operatorname*{arg\,min}_{\boldsymbol{z}\in\mathbb{R}^n} \left\{ \frac{1}{2}\|\boldsymbol{z} - \boldsymbol{x}\|_2^2 + \frac{\gamma}{2}\|\boldsymbol{H}\boldsymbol{z} - \boldsymbol{y}\|_2^2 \right\} \tag{7a}$$

$$= [\boldsymbol{I} + \gamma\boldsymbol{H}^{\mathsf{T}}\boldsymbol{H}]^{-1}(\boldsymbol{x} + \gamma\boldsymbol{H}^{\mathsf{T}}\boldsymbol{y}). \tag{7b}$$

The matrix inversion in (7) can make ADMM updates computationally expensive for problems where the measurement matrix is not easily invertible. On the other hand, ADMM is known to be fast for matrices that can be inverted efficiently [36]–[38].

The theoretical analysis in this paper is closely related to the convergence results established for first-order methods by Nesterov [39] and Beck and Teboulle [35]. In particular, our work is related to inexact proximal-gradient optimization that has been extensively investigated by several researchers [40]–[47]. We extend this prior work beyond traditional optimization, where denoising operators do not necessarily correspond to proximal operators of a given objective. To achieve this, we adopt the monotone operator theory [48], [49], which enables a unified analysis of PnP methods by expressing them as finding zeros of an operator.

B. Using denoisers as priors

Both PGM and ADMM have modular structures in the sense that the prior on the image is only imposed via the proximal operator. Additionally, since the proximal operator is mathematically equivalent to regularized image denoising, the powerful idea of Venkatakrishnan *et al.* [14] was to consider replacing it with a more general denoising operator $\mathsf{denoise}_\sigma(\cdot)$ of controllable strength $\sigma > 0$. For compatibility with the traditional optimization formulation, this strength parameter is often set as $\sigma = \sqrt{\gamma\lambda}$, where $\gamma > 0$ is the optimization algorithm step-size, and $\lambda > 0$ is the regularization parameter.

While the original formulation of PnP [14] relies on ADMM, it has recently been shown that it can be as effective

Algorithm 3 PnP-PGM/PnP-APGM

input: $\boldsymbol{x}^0 = \boldsymbol{s}^0 \in \mathbb{R}^n$, $\gamma > 0$, $\sigma > 0$, and $\{q_k\}_{k\in\mathbb{N}}$
for $k = 1, 2, \dots$ **do**
  $\boldsymbol{z}^k \leftarrow \boldsymbol{s}^{k-1} - \gamma \nabla d(\boldsymbol{s}^{k-1})$
  $\boldsymbol{x}^k \leftarrow \mathsf{denoise}_\sigma(\boldsymbol{z}^k)$
  $\boldsymbol{s}^k \leftarrow \boldsymbol{x}^k + ((q_{k-1} - 1)/q_k)(\boldsymbol{x}^k - \boldsymbol{x}^{k-1})$
end for

Algorithm 4 PnP-ADMM

input: $\boldsymbol{x}^0 \in \mathbb{R}^n$, $\boldsymbol{s}^0 = \boldsymbol{0}$, $\gamma > 0$, and $\sigma > 0$
for $k = 1, 2, \dots$ **do**
  $\boldsymbol{z}^k \leftarrow \mathsf{prox}_{\gamma d}(\boldsymbol{x}^{k-1} - \boldsymbol{s}^{k-1})$
  $\boldsymbol{x}^k \leftarrow \mathsf{denoise}_\sigma(\boldsymbol{z}^k + \boldsymbol{s}^{k-1})$
  $\boldsymbol{s}^k \leftarrow \boldsymbol{s}^{k-1} + (\boldsymbol{z}^k - \boldsymbol{x}^k)$
end for

when used with other proximal algorithms [24]–[26], or with another class of algorithms known as approximate message passing (AMP) [50]–[52]. AMP-based algorithms have been shown to be effective for problems where $\boldsymbol{H}$ is large and random [53], [54], but are also known to be unstable for general matrices $\boldsymbol{H}$ [55]–[57]. Therefore, in this paper, our focus will be exclusively on the variants of PnP based on PGM and ADMM, summarized in Algorithm 3 and 4, respectively.

Several recent publications have analyzed the theoretical convergence of PnP algorithms [18], [19], [23], [25]. Sreehari *et al.* [18] have established the convergence of PnP-ADMM to the global minimum of some implicitly defined objective function. Specifically, by building on the theoretical analysis by Moreau [34], they show that $\mathsf{denoise}_\sigma$ is a valid proximal operator of some implicit regularizer if it is nonexpansive and $\nabla \mathsf{denoise}_\sigma(\boldsymbol{x})$ is a symmetric matrix for all $\boldsymbol{x} \in \mathbb{R}^n$. Chan *et al.* [19] have proved a fixed-point convergence of PnP-ADMM for bounded denoisers, which are defined as denoisers satisfying

$$\frac{1}{n}\|\mathsf{denoise}_\sigma(\boldsymbol{x}) - \boldsymbol{x}\|_2^2 \leq \sigma^2\, c, \tag{8}$$

for any $\boldsymbol{x} \in \mathbb{R}^n$, where $c > 0$ is a constant independent of n and σ. Meinhardt *et al.* [25] have shown that for continuous denoisers several PnP algorithms admit an equivalent fixed-point iteration. More recently, Teodoro *et al.* [23] considered a special class of denoisers based on Gaussian mixture models (GMMs) and showed that PnP-ADMM converges when the GMM denoiser is simplified to be a linear function of its input.

A different but related approach to denoiser-driven regularization was recently proposed by Romano *et al.* [58]. They proposed the regularization by denoising (RED) framework, in which an explicit regularizer is constructed as

$$r(\boldsymbol{x}) = \frac{1}{2}\boldsymbol{x}^{\mathsf{T}}(\boldsymbol{x} - \mathsf{denoise}_\sigma(\boldsymbol{x})). \tag{9}$$

Remarkably, they also showed that under some conditions, the gradient of the regularizer has a very simple expression. More recently, Reehorst and Schniter [59] have provided additional insight into RED by establishing conditions for the existence of explicit regularizers based on denoising operators. The key difference between PnP and RED is that the former does not seek to define an explicit regularization functional, but relies on the fixed points of a given denoising operator for regularization. This generality of the PnP framework makes it widely applicable, but also substantially complicates its theoretical analysis.

Another recent related framework is the consensus equilibrium (CE) by Buzzard *et al.* [60]. Given multiple sources of information (defined via image denoisers or other similar mappings), CE proposes to fuse them by computing a specific equilibrium point. The CE framework extends the traditional consensus optimization [13] to operators that are not necessarily proximal operators and formulates a new variant of PnP that can handle multiple denoising functions. In this paper, we will restrict our attention to the traditional PnP formulation under PGM-based optimization.

III. Batch Algorithm

In this section, we present a detailed theoretical convergence analysis of batch PnP-PGM. The results are based on the fixed point analysis of Algorithm 3 and rely on basic convex and monotone analysis, summarized in Appendix A.

The central building block of PnP-PGM is the following denoiser-gradient operator

$$\mathsf{P}(\boldsymbol{x}) \triangleq \mathsf{denoise}_\sigma(\boldsymbol{x} - \gamma \nabla d(\boldsymbol{x})), \tag{10}$$

which first computes the gradient-step with respect to the function d and then denoises the result with a given denoiser. Throughout this paper, we assume that the function d is convex and has a Lipschitz continuous gradient with constant $L > 0$. We are interested in convergence of Algorithm 3 to the set of fixed points of the operator P

$$\mathsf{fix}(\mathsf{P}) \triangleq \{\boldsymbol{x} \in \mathbb{R}^n : \boldsymbol{x} = \mathsf{P}(\boldsymbol{x})\}. \tag{11}$$

Note that when $\mathsf{denoise}_\sigma$ is the proximal operator of a convex function, $\mathsf{fix}(\mathsf{P})$ coincides with the set of solutions of (1).

Proposition 1. *Let* $\mathsf{denoise}_\sigma(\cdot) = \mathsf{prox}_{\gamma r}(\cdot)$ *for* $\gamma, \sigma > 0$. *Then* $\boldsymbol{x}^* \in \mathsf{fix}(\mathsf{P})$ *if and only if it minimizes* $f = d + r$.

Proof. See Appendix C. □

Our central goal, however, is to generalize $\mathsf{denoise}_\sigma$ beyond proximal operators. The key assumption that we adopt for our analysis is that the denoiser is averaged (see Appendix A).

Definition 1. *Consider an operator* $\mathsf{denoise}_\sigma$ *and a constant* $\theta \in (0, 1)$. $\mathsf{denoise}_\sigma$ *is* θ*-averaged if and only if the operator* $(1 - 1/\theta)\mathsf{I} + (1/\theta)\mathsf{denoise}_\sigma$, *where* I *denotes the identity operator, is nonexpansive.*

The class of averaged operators is a superset of proximal operators and a subset of nonexpansive operators. In fact, the proximal operator is an averaged operator with $\theta = 1/2$. Note that given any nonexpansive denoiser, it is always possible to make it averaged by defining a damped operator $\mathsf{D} \triangleq (1 - \theta)\mathsf{I} + \theta\mathsf{denoise}_\sigma$, with $\theta \in (0, 1)$, which has the same set of fixed points as $\mathsf{denoise}_\sigma$ [5].

Assumption 1. *We analyze PnP-PGM under the following assumptions:*

(a) The function d is convex and differentiable with a Lipschitz continuous gradient of constant $L > 0$.
(b) $\mathsf{denoise}_\sigma$ is θ-averaged with $\theta \in (0, 1)$ for any $\sigma > 0$.
(c) There exists $\boldsymbol{x}^* \in \mathbb{R}^n$ such that $\boldsymbol{x}^* \in \mathsf{fix}(\mathsf{P})$.

We can then establish the following convergence result.

Proposition 2. *Run PnP-PGM for $t \geq 1$ iterations under Assumption 1 with step-size $\gamma \in (0, 1/L]$ and $q_k = 1$ for all $k \in \{1, \dots, t\}$. Then, for any $\boldsymbol{x}^* \in \mathsf{fix}(\mathsf{P})$, we have that*

$$\frac{1}{t}\sum_{k=1}^{t} \|\boldsymbol{x}^{k-1} - \mathsf{P}(\boldsymbol{x}^{k-1})\|_2^2 \leq \frac{2}{t}\left(\frac{1+\theta}{1-\theta}\right)\|\boldsymbol{x}^0 - \boldsymbol{x}^*\|_2^2.$$

Proof. See Appendix D. □

The direct consequence of Proposition 2 is that

$$\min_{k\in\{1,\dots,t\}} \left\{\|\boldsymbol{x}^{k-1} - \mathsf{P}(\boldsymbol{x}^{k-1})\|_2^2\right\} = O(1/t), \tag{12}$$

that is under Assumption 1, the iterates of PnP-PGM can get arbitrarily close to the set of fixed points $\mathsf{fix}(\mathsf{P})$ with rate $O(1/t)$. This result is different from the traditional monotonic $O(1/t)$ convergence of PGM to the minimum of an objective function [35]. The convergence in (12) is not monotonic and is expressed in terms of the smallest distance to $\boldsymbol{x} = \mathsf{P}(\boldsymbol{x})$ in the window of $t \geq 1$ previous iterations. This is because PnP-PGM is not necessarily minimizing any objective function. On the other hand, the result still guarantees that, given a sufficient number of $t \geq 1$ iterations, the iterates of PnP-PGM can get arbitrarily close to the set $\mathsf{fix}(\mathsf{P})$.

Recently, Meinhardt *et al.* [25] have showed that for continuous denoisers, the fixed-points of several PnP algorithms coincide. The following proposition is a minor variant of their result tailored for PnP-ADMM.

Proposition 3. *Under Assumption 1, the set of fixed-points of PnP-ADMM coincides with $\mathsf{fix}(\mathsf{P})$.*

Proof. See Appendix E. □

In the context of the work by Sreehari *et al.* [18], the propositions above indicate that the symmetric gradient assumption is not necessary for the convergence of PnP-PGM. Since the symmetry of $\nabla\mathsf{denoise}_\sigma(\boldsymbol{x})$ in [18] ensures that the denoiser is an implicitly defined proximal operator, the results here provide a generalization of the convergence beyond proximal operators. Moreover, both PnP-PGM and PnP-ADMM are equivalent in the sense that they have the same set of solutions specified by $\mathsf{fix}(\mathsf{P})$.

The bounded denoiser assumption (8) is a more relaxed assumption on the denoising operator and was used to analyze PnP-ADMM. However, we argue that it is not sufficient to guarantee the convergence of PnP-PGM. The following proposition builds on a specific counter example.

Proposition 4. *There exists a function d that is convex and has a Lipschitz continuous gradient of constant L, and a denoiser $\mathsf{denoise}_\sigma$ that satisfies (8), such that PnP-PGM with the step $\gamma \in (0, 1/L)$, $q_k = 1$ for all $k \in \mathbb{N}$, and $\sigma > \gamma/\sqrt{c}$ diverges.*

Algorithm 5 PnP-SPGM

input: $\boldsymbol{x}^0 = \boldsymbol{s}^0 \in \mathbb{R}^n$, $\gamma > 0$, $\sigma > 0$, $\{q_k\}$, and $B \geq 1$
for $k = 1, 2, \dots$ **do**
  $\hat{\nabla} d(\boldsymbol{s}^{k-1}) \leftarrow \mathsf{minibatchGradient}(\boldsymbol{s}^{k-1}, B)$
  $\boldsymbol{z}^k \leftarrow \boldsymbol{s}^{k-1} - \gamma\hat{\nabla} d(\boldsymbol{s}^{k-1})$
  $\boldsymbol{x}^k \leftarrow \mathsf{denoise}_\sigma(\boldsymbol{z}^k)$
  $\boldsymbol{s}^k \leftarrow \boldsymbol{x}^k + ((q_{k-1} - 1)/q_k)(\boldsymbol{x}^k - \boldsymbol{x}^{k-1})$
end for

Proof. See Appendix F. □

Definition 1 makes verifying that a denoiser is averaged equivalent to verifying nonexpansiveness of some operator. As was argued in several recent publications [18], [19], [23] the task is more difficult for some denoisers than it is for others and there exist denoisers for which this condition does not hold. However, all recently designed denoisers for PnP from [18], [23] satisfy our assumptions. In fact, the denoisers that satisfy conditions outlined in [18] correspond to implicit proximal operators, which implies that they are $\theta = 1/2$ averaged operators. For example, the modified nonlocal means (NLM) filter specifically designed in [18] is by definition an averaged operator.

IV. Online Algorithm

We now introduce our second key contribution: the new online variant of PnP-PGM called PnP-SPGM. We additionally prove its convergence for averaged denoisers.

In many imaging applications, the data-fidelity term d consists of a large number of component functions

$$d(\boldsymbol{x}) = \mathbb{E}[d_i(\boldsymbol{x})] = \frac{1}{I}\sum_{i=1}^{I} d_i(\boldsymbol{x}), \tag{13}$$

where each d_i typically depends only on the subset $\boldsymbol{y}_i$ of the measurements in $\boldsymbol{y}$. For example, in tomographic imaging each $\boldsymbol{y}_i$ corresponds to a single projection of an object along a specific angle [30]. Note that in equation (13), the expectation is taken over a uniformly distributed random variable $i \in \{1, \dots, I\}$. The computation of the gradient

$$\nabla d(\boldsymbol{x}) = \mathbb{E}[\nabla d_i(\boldsymbol{x})] = \frac{1}{I}\sum_{i=1}^{I} \nabla d_i(\boldsymbol{x}), \tag{14}$$

scales with the total number of components I, which means that when the latter is large, the memory requirements or computation time of the classical batch PnP algorithms may become impractical. The central idea of PnP-SPGM, summarized in Algorithm 5, is to approximate the gradient at every iteration with an average of $B \ll I$ component gradients

$$\hat{\nabla} d(\boldsymbol{x}) = \frac{1}{B}\sum_{b=1}^{B} \nabla d_{i_b}(\boldsymbol{x}), \tag{15}$$

where $i_1, \dots, i_B$ are independent random indices that are distributed uniformly over $\{1, \dots, I\}$. The minibatch size

parameter $B \geq 1$ controls the number of gradient components used at every iteration.

Assumption 2. *We analyze PnP-SPGM under the following assumptions:*

(a) The functions d_i are all convex and differentiable with the same Lipschitz constant $L > 0$.
(b) $\mathsf{denoise}_\sigma$ is θ-averaged with $\theta \in (0, 1)$ for any $\sigma > 0$.
(c) There exists $\boldsymbol{x}^* \in \mathbb{R}^n$ such that $\boldsymbol{x}^* \in \mathsf{fix}(\mathsf{P})$.
(d) At every iteration, the gradient estimate is unbiased and has a bounded variance:

$$\mathbb{E}[\hat{\nabla} d(\boldsymbol{x})] = \nabla d(\boldsymbol{x}) \quad \text{and} \quad \mathbb{E}[\|\nabla d(\boldsymbol{x}) - \hat{\nabla} d(\boldsymbol{x})\|_2^2] \leq \frac{\nu^2}{B},$$

for some constant $\nu > 0$.

Note that Assumption 2(a) implies that the complete data-fidelity term d is also convex and has a Lipschitz continuous gradient of constant L. The key difference between Assumption 1 and Assumption 2 is the last condition. The fact that the minibatch gradient is unbiased is the direct consequence of (15). The bounded variance assumption is a standard assumption used in the analysis of online and stochastic algorithms [46], [61], [62].

Proposition 5. *Run PnP-SPGM for $t \geq 1$ iterations under Assumption 2 with step-size $\gamma \in (0, 1/L]$ and $q_k = 1$ for all $k \in \{1, \dots, t\}$. Then, for any $\boldsymbol{x}^* \in \mathsf{fix}(\mathsf{P})$, we have that*

$$\mathbb{E}\left[\frac{1}{t}\sum_{k=1}^{t} \|\boldsymbol{x}^{k-1} - \mathsf{P}(\boldsymbol{x}^{k-1})\|_2^2\right] \leq 2\left(\frac{1+\theta}{1-\theta}\right)\left[\frac{\gamma^2\nu^2}{B} + \frac{2\gamma\nu}{\sqrt{B}}\|\boldsymbol{x}^0 - \boldsymbol{x}^*\|_2 + \frac{\|\boldsymbol{x}^0 - \boldsymbol{x}^*\|_2^2}{t}\right],$$

where $\mathsf{P}(\cdot)$ is given by (10)*.*

Proof. See Appendix G. □

This result shows that the convergence in expectation of PnP-SPGM to an element of $\mathsf{fix}(\mathsf{P})$ is proportional to the step-size γ and inversely proportional to the mini-batch size B. By controlling these two parameters, we can obtain the following convergence rates.

Corollary 1. *Consider Proposition 5 with the following fixed (i.e., independent of iteration k) parameters.*

(a) For $\gamma = 1/(L\sqrt{t})$ and $B = 1$, we have that

$$\mathbb{E}\left[\frac{1}{t}\sum_{k=1}^{t} \|\boldsymbol{x}^{k-1} - \mathsf{P}(\boldsymbol{x}^{k-1})\|_2^2\right] \leq \frac{A}{\sqrt{t}},$$

(b) For $\gamma = 1/L$ and $B = t$, we have that

$$\mathbb{E}\left[\frac{1}{t}\sum_{k=1}^{t} \|\boldsymbol{x}^{k-1} - \mathsf{P}(\boldsymbol{x}^{k-1})\|_2^2\right] \leq \frac{A}{\sqrt{t}},$$

(c) For $\gamma = 1/(L\sqrt{t})$ and $B = t$, we have that

$$\mathbb{E}\left[\frac{1}{t}\sum_{k=1}^{t} \|\boldsymbol{x}^{k-1} - \mathsf{P}(\boldsymbol{x}^{k-1})\|_2^2\right] \leq \frac{A}{t},$$

where

$$A \triangleq 2\left(\frac{1+\theta}{1-\theta}\right)\left(\|\boldsymbol{x}^0 - \boldsymbol{x}^*\|_2 + \frac{\nu}{L}\right)^2.$$

Fig. 1. Test images used. Top row from left to right: *Barbara*, *Boat*, *Foreman*, *House*. Bottom row from left to right: *Lenna*, *Monarch*, *Parrot*, *Peppers*.

Corollary 1(c) implies the worst-case convergence rate

$$\mathbb{E}\left[\min_{k\in\{1,\dots,t\}} \left\{\|\boldsymbol{x}^{k-1} - \mathsf{P}(\boldsymbol{x}^{k-1})\|_2^2\right\}\right] = O(1/t), \tag{16}$$

which means that under Assumption 2 and with a particular selection of parameters B and γ, the iterates of PnP-SPGM (in expectation) can get arbitrarily close to $\mathsf{fix}(\mathsf{P})$ as $O(1/t)$.

V. Numerical Simulations

We now empirically validate PnP-SPGM in the context of diffraction tomography (DT) using three popular denoisers: TV [1], BM3D [15], and TNRD [17]. Our goal is not to justify the PnP framework, as its benefits have been well illustrated in prior work [14], [18], [26], but to focus on the aspects that relate to online processing of data. Therefore, we first discuss empirical convergence of PnP-SPGM, and then highlight the benefit of using it for processing a large number of measurements.

A. Diffraction tomography

DT is a technique used to form an image of the distribution of dielectric permittivity within an object from multiple measurements of light it scatters [30], [63]. This problem is common in a number of applications—including ultrasound [64] and optical microscopy [65]—and is known to be highly data-intensive. A typical reconstruction task uses hundreds or thousands of measurements for forming a single image. As is common in DT, we adopt the first-Born approximation [63], which leads to the linear inverse problem formulation of image reconstruction.

Note that PnP-SPGM is applicable beyond DT and our choice of the latter is only due to the fact that image reconstruction in DT requires the processing of a large number of distinct measurements. Additionally, our focus is not on the experimental application of DT, but rather on the demonstration of our online algorithm for image reconstruction. Hence, we restrict our study here to image reconstruction from purely simulated DT data, which enables optimal parameter tuning and quantitative comparisons.

Consider an object with the permittivity distribution $\epsilon(\boldsymbol{r})$ within a bounded domain $\Omega \subseteq \mathbb{R}^2$ with a background medium of permittivity ϵ_b. The object is illuminated with a monochromatic and coherent incident electric field $u_{\text{in}}(\boldsymbol{r})$ emitted by

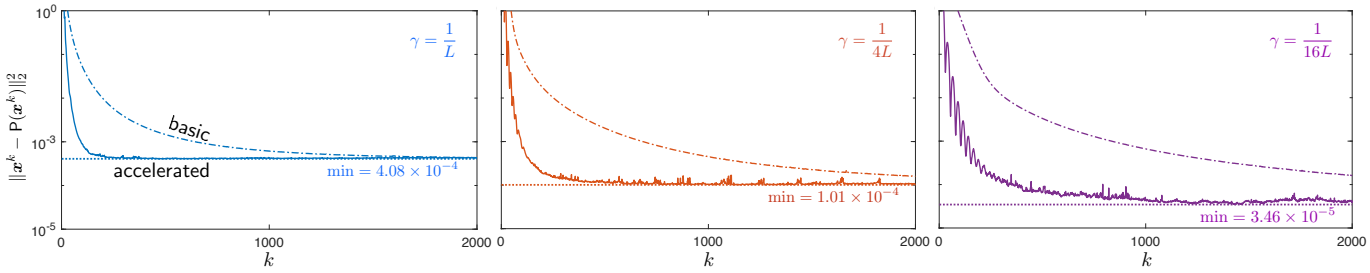

Fig. 2. Illustration of the influence of the step-size γ on the convergence of PnP-SPGM with BM3D as the denoiser. The distance to a fixed point is plotted against the iteration number for 3 distinct step-sizes for both accelerated (solid) and basic (dashed) variants of PnP-SPGM for $B = 30$. The dotted line at the bottom shows the minimal distance to a fixed point attained by the algorithm. This plot illustrates that the empirical performance of PnP-SPGM under BM3D is consistent with Proposition 5, where the accuracy improves with smaller γ.

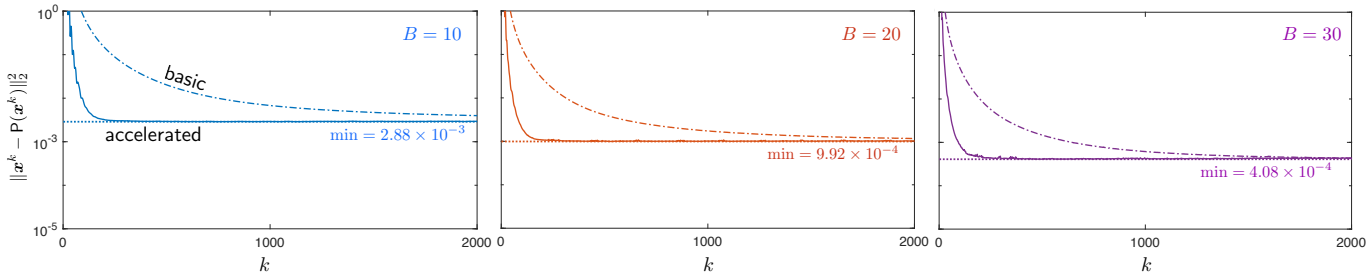

Fig. 3. Illustration of the influence of the minibatch size B on the convergence of PnP-SPGM with BM3D as a denoiser. The distance to a fixed point is plotted against the iteration number for 3 distinct minibatch sizes for both accelerated (solid) and basic (dashed) variants of PnP-SPGM for $\gamma = 1/L$. The dotted line at the bottom shows the minimal distance to a fixed point attained by the algorithm. This plot illustrates that the empirical performance of PnP-SPGM using BM3D is consistent with Proposition 5, where the accuracy improves with larger B.

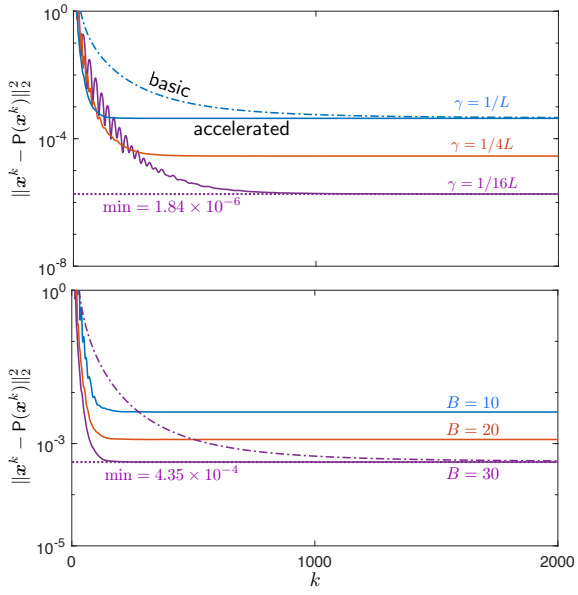

Fig. 4. Illustration of the influence of the step and minibatch sizes on the convergence of PnP-SPGM with TV as the denoiser. The dotted line at the bottom shows the minimal distance to a fixed point attained by the algorithm. A proximal operator is $(1/2)$-averaged, which means that it perfectly satisfies the assumptions of Proposition 5.

one of N transmitters. The incident field is assumed to be known both inside Ω and at the sensor domain $\Gamma \subseteq \mathbb{R}^2$. The measurements correspond to the field scattered by the object recorded by M receivers located within Γ. Under the first-Born approximation, the measurement matrix for a single illumination can be represented as $\boldsymbol{H} = \mathbf{S}\mathrm{diag}(\mathbf{u}_{\text{in}})$, where $\mathbf{u}_{\text{in}} \in \mathbb{C}^N$ is the input field u_{in} inside Ω, and $\mathbf{S} \in \mathbb{C}^{M \times N}$ is the discretization of the Green's function evaluated at Γ [66]. In practice, the image reconstruction relies on the set of illuminations $\{\mathbf{u}_{\text{in}}^i\}_{i \in \{1,\dots,I\}}$, with each individual illumination resulting in a measurement $\boldsymbol{y}^i \in \mathbb{C}^M$ and a distinct measurement matrix $\boldsymbol{H}_i$.

The objects we reconstruct correspond to the eight standard grayscale images shown Fig. 1. The physical size of an image is set to 18 cm $\times$ 18 cm, discretized to a grid of 256×256. The wavelength of the illumination was set to $\lambda = 0.84$ cm and the background medium was assumed to be air with $\epsilon_b = 1$. We additionally set the number of transmitters to $N = 60$, distributed uniformly along a circle of radius 1.6 meters, and for each illumination, the corresponding scattered field is measured by $M = 360$ receivers around the object. The simulated measurements were additionally corrupted by an additive white Gaussian noise (AWGN) corresponding to 40 dB of input signal-to-noise ratio (SNR). The quantitative evaluation of the experimental results is also provided in terms of SNR defined as

$$\text{SNR (dB)} \triangleq 10 \log_{10} \left(\frac{\|\boldsymbol{x}\|_2^2}{\|\widehat{\boldsymbol{x}} - \boldsymbol{x}\|_2^2} \right),$$

where $\widehat{\boldsymbol{x}}$ and $\boldsymbol{x}$ are the reconstructed and the ground truth images, respectively. We use the term *average SNR* to indicate the SNR averaged over all the test images. In each experiment, all algorithmic hyperparameters were optimized for the best SNR performance with respect to the ground truth test image.

TABLE I
MINIMAL DISTANCE AVERAGED OVER THE TEST IMAGE SET

Denoiser	**Step-size** (γ)			**Mini-batch size** (B)		
	$1/L$	$1/4L$	$1/16L$	10	20	30
TV	4.35e-4	2.86e-5	1.84e-6	4.14e-3	1.20e-3	4.35e-4
BM3D	4.08e-4	1.01e-4	3.46e-5	2.88e-3	9.92e-4	4.08e-4
TNRD	1.19e-1	2.20e-2	3.14e-3	7.50e-1	3.07e-1	1.19e-1

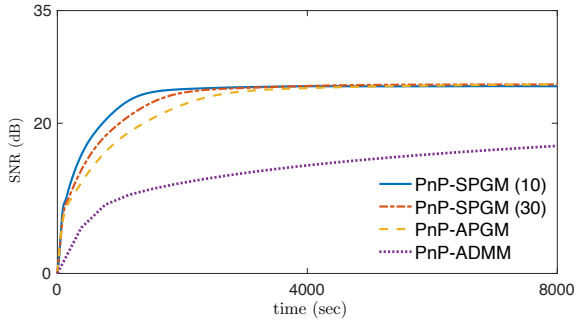

Fig. 5. Comparison between the batch and online PnP algorithms for a fixed reconstruction time. SNR (dB) is plotted against the time in seconds for three algorithms: PnP-SPGM, PnP-APGM, and PnP-ADMM. Both PnP-APGM and PnP-ADMM use the full set of 60 illuminations at every iteration, while PnP-SPGM uses a random subset of 10 or 30 illuminations. This lower per-iteration cost, leads to a substantially faster convergence of PnP-SPGM.

B. Convergence of PnP-SGD

One of the key conclusions of Proposition 5 is that the final accuracy of PnP-SPGM to a fixed point is proportional to the step-size and inversely proportional to the minibatch size. In order to numerically evaluate the convergence, we define the distance to $\mathsf{fix}(\mathsf{P})$ at the kth iteration as

$$\mathsf{dist}(\boldsymbol{x}^k) \triangleq \|\boldsymbol{x}^k - \mathsf{P}(\boldsymbol{x}^k)\|_2^2 , \tag{17}$$

where P is given by (10). As the sequence $\{\boldsymbol{x}^k\}$ approaches $\mathsf{fix}(\mathsf{P})$, $\mathsf{dist}(\boldsymbol{x}^k)$ approaches zero.

Fig. 2 and Fig. 3 empirically evaluate the evolution of the distance to a fixed point for different step and minibatch sizes, respectively. PnP-SPGM, with BM3D as a denoiser, is run until convergence with $\gamma \in \{1/L, 1/(4L), 1/(16L)\}$ and $B \in \{10, 20, 30\}$. Here, the quantity $L > 0$ denotes the Lipschitz constant, which, for linear inverse problems, corresponds to the squared largest singular value of the measurement matrix [35]. We show the performance of both basic and accelerated variants of PnP-SPGM, where the latter is obtained by setting $\{q_k\}$ as in (5). The plots clearly illustrate the improvement in final accuracy for smaller γ and larger B, which is consistent with Proposition 5. Additionally, they indicate that the convergence is significantly improved when using the accelerated variant of the algorithm. Note that our theoretical analysis does not predict monotonic reduction of the distance, which also seems to be consistent with the empirical performance of PnP-SPGM. In Fig. 4, we provide a reference plot showing the performance of PnP-SPGM under TV, which is a valid proximal operator and hence is known to be a $1/2$-averaged operator. We can again observe that the convergence behavior of PnP-SPGM is consistent with Proposition 5. Finally, the summary in Table I, highlights the same convergence trends for all three algorithms, where both γ and B control the accuracy of PnP-SPGM.

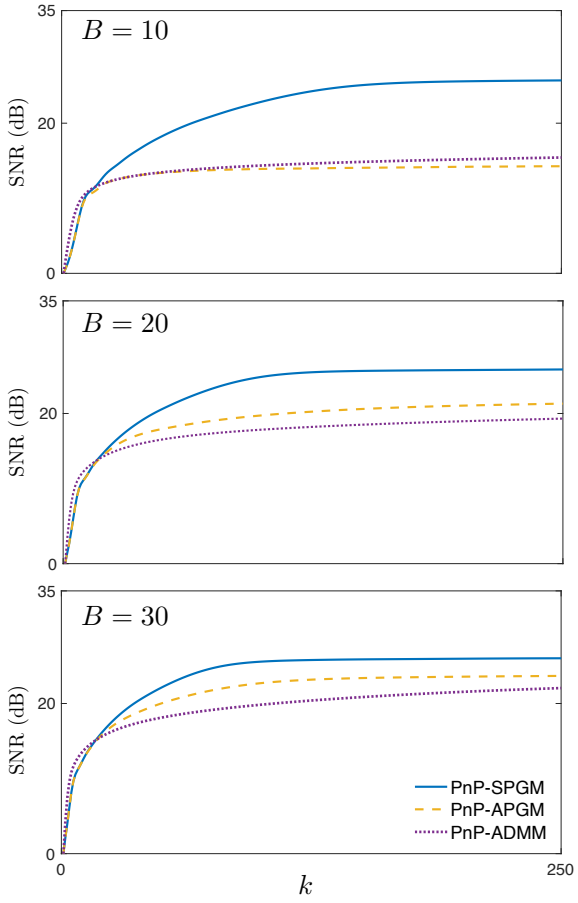

Fig. 6. Comparison between the batch and online PnP algorithms under a fixed measurement budget. SNR (dB) is plotted against the number of iterations for three algorithms: PnP-SPGM, PnP-APGM, and PnP-ADMM. From the top to the bottom, figures show the performance when the budget is 10, 20, and 30 illuminations, respectively. The plot illustrates that for the same per iteration cost, PnP-SPGM can significantly outperform its batch counterparts.

C. Benefits of online processing

We now highlight the higher efficiency of PnP-SPGM against PnP-PGM and PnP-ADMM for larger number of measurements. Specifically, we consider two scenarios where: (a) the total time budget is fixed; (b) the number of measurements is fixed. While we use BM3D as our plug-in operator of choice, we note that our observations here directly generalize to any other denoiser.

Fig. 5 compares the average reconstruction SNR of PnP-SPGM, PnP-APGM, and PnP-ADMM for a fixed runtime. The batch algorithms use the full 60 illuminations at every iteration, while PnP-SPGM uses 10 and 30 illuminations per iteration. This gives PnP-SPGM a significantly lower per iteration cost compared to the batch algorithms. Specifically, the average per iteration time for PnP-SPGM using $B = 10$, PnP-SPGM using $B = 30$, PnP-APGM, and PnP-ADMM was 8.86 seconds, 22.10 seconds, 44.94 seconds, and 382.83 seconds, respectively. The higher cost of PnP-ADMM is the result of the forward model inversion in (7). The figure

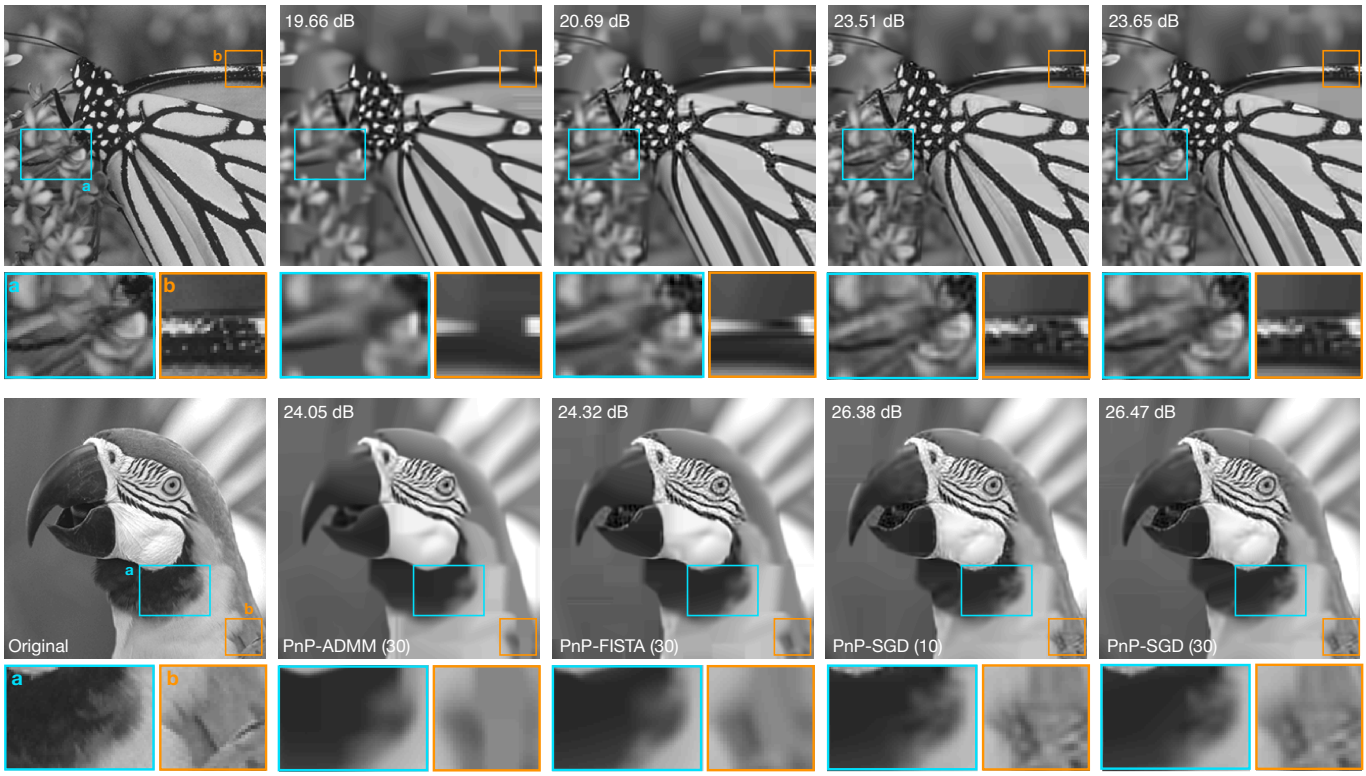

Fig. 7. Visual illustration of the reconstructed *Monarch* and *Parrot* images obtained using PnP-SPGM, PnP-APGM, and PnP-ADMM, all under BM3D. The original images are displayed in the first column. The second and the third columns show the results of PnP-APGM and PnP-ADMM with the budget of 30 illuminations, and the fourth and the fifth columns present the results of the PnP-SPGM with the budget of 10 and 30 illuminations. Visual differences are highlighted using the rectangles drawn inside the images. Each reconstruction is labeled with its SNR (dB) value with respect to the original image.

illustrates that, in practice, the solution of PnP-SPGM is close to that of the batch algorithm with the final SNRs for $B = 10$ and $B = 30$ being within 0.2 dB and 0.01 dB, respectively, from that of PnP-APGM. Additionally, PnP-SPGM achieves a significant speedup due to the reduction in per-iteration complexity. This indicates to the potential of the algorithm for efficient image reconstruction from a large number of measurements.

Fig. 6 compares the average reconstruction SNR of PnP-SPGM, PnP-APGM, and PnP-ADMM for a fixed per-iteration measurement budget. The batch algorithms are allowed to use only 10 (top figure), 20 (middle figure), or 30 (bottom figure) uniformly distributed illuminations. Convergence of each algorithm was observed in 1000 iterations, and the figure displays the average SNR within the first 250 iterations since PnP-SPGM has already converged in all three plots. Similarly, PnP-SPGM uses the same number of illuminations per iteration, but randomly cycles through all the measurements. This means that in each figure both PnP-SPGM and PnP-APGM have the same per-iteration computational complexity. The computational complexity of PnP-ADMM is higher due to the need to invert the measurement matrix. Table II shows the final SNR obtained by all three algorithms on each individual image in the dataset. Additionally, two visual illustrations on *Monarch* and *Parrot* are shown in Fig. 7. The two rectangles under each image show areas rich in texture that were selected to highlight the visual differences in the results. As expected, PnP-SPGM achieves dramatically higher SNR compared to batch algorithms, since it makes use of the full set of measurements. Additionally, we note the comparable final SNR performance of PnP-SPGM with $B = 10$ and $B = 30$, with the latter leading to a faster convergence speed. These results again highlight the potential of PnP-SPGM for large-scale PnP image reconstruction.

The simulations in this section highlight the benefit of PnP-SPGM in tomographic imaging, where each measurement contains information from a large portion of the object. PnP-SPGM leverages this setting to improve the computational and memory efficiency of processing a large number of measurements. Whether this benefit of using PnP-SPGM would persist in other imaging problems — such as inpainting or deblurring, where the information on the unknown is heavily localized in the measurements — is still an open question and a potential avenue of future research.

To conclude this section, let us put the results here in the context of our theoretical analysis. Proposition 5 reveals that PnP-SPGM converges to the same set of fixed points $\mathsf{fix}(\mathsf{P})$ as PnP-PGM and PnP-ADMM, up to a term that depends on the minibatch size $B \geq 1$. Larger B leads to a higher accuracy of PnP-SPGM with respect to $\mathsf{fix}(\mathsf{P})$, which was empirically confirmed in Fig. 3. The SNR results here additionally reveal that even with a relatively small B, PnP-SPGM is accurate in terms of image quality. For example, in Table II, we can observe that the average SNR difference between PnP-SPGM

TABLE II
INDIVIDUAL RECONSTRUCTION SNRS FOR EACH IMAGE.

Images	PnP-ADMM (10)	PnP-ADMM (20)	PnP-ADMM (30)	PnP-APGM (10)	PnP-APGM (20)	PnP-APGM (30)	PnP-SPGM (10)	PnP-SPGM (20)	PnP-SPGM (30)
Barbara	15.62	19.16	21.18	13.32	16.60	20.21	23.61	23.79	23.90
Boat	15.94	19.82	23.10	13.69	18.40	22.01	24.87	25.05	25.15
Foreman	23.10	27.27	29.19	18.46	26.64	28.61	29.61	29.91	29.80
House	19.23	23.46	26.43	15.68	25.36	26.79	28.29	27.84	28.41
Lenna	15.52	20.32	23.17	13.49	20.57	22.91	25.30	25.35	25.38
Monarch	11.46	15.82	19.66	8.80	17.38	20.69	23.51	23.50	23.65
Parrot	17.29	21.49	24.05	13.73	22.29	24.32	26.38	26.38	26.47
Pepper	15.49	20.46	22.90	11.67	20.89	22.96	24.92	24.82	25.15
Average	16.71	20.98	23.71	14.26	21.02	23.73	25.85	25.83	26.04

with $B = 10$ and $B = 30$ is within 0.2 dB of each other. Additionally, in Fig. 5, we observe that the batch and online algorithms approximately achieve the same final SNR performance. These observations suggest that while there is an order of magnitude difference in accuracy between $B = 10$ and $B = 30$ when measured in terms of the distance to a fixed point (see Fig. 3), the difference is relatively mild when measured in terms of image quality (see Fig. 7), with smaller B nearly matching the image quality of the batch algorithm.

VI. CONCLUSION

The online PnP algorithm developed in this paper is beneficial in the context of large-scale image reconstruction, when the amount of data is too large to be processed jointly. We have presented an in-depth theoretical convergence analysis for both batch and online variants of PnP-PGM. Our work represents a substantial extension of the current convergence theory of PnP-algorithms for image reconstruction. Related experiments are also presented to empirically confirm the proposed propositions and to elucidate the higher efficiency of PnP-SPGM in different representative situations. Future work will aim to apply the algorithm to other image reconstruction tasks, relax some of the assumptions, and extend the theoretical results in this paper to ADMM and APGM.

APPENDIX

A. Review of Averaged Operators

We start by reviewing the key concepts useful for our analysis. A more complete description of these ideas can be found in literature [5], [48], [49].

We will represent denoisers as functions $\mathsf{D}_\sigma : \mathbb{R}^n \to \mathbb{R}^n$ that depend on $\sigma > 0$. We will also use a shorthand notation $\mathsf{G}_\gamma \triangleq \mathsf{I} - \gamma\nabla d$ to denote the gradient-step operator, where I denotes the identity operator. We will assume that all operators are defined everywhere on $\mathbb{R}^n$.

Definition 2. *An operator F is Lipschitz continuous with a constant $L > 0$ if*

$$\|\mathsf{F}(\boldsymbol{x}) - \mathsf{F}(\boldsymbol{y})\|_2 \leq L\|\boldsymbol{x} - \boldsymbol{y}\|_2, \quad \forall \boldsymbol{x}, \boldsymbol{y} \in \mathbb{R}^n. \tag{18}$$

When $L = 1$, F is said to be nonexpansive.

It is straightforward to show that given two operators F_1 and F_2 with Lipschitz constants L_1 and L_2, respectively, the composition $\mathsf{F} \triangleq \mathsf{F}_2 \circ \mathsf{F}_1$ has Lipschitz constant $L = L_1L_2$. This means that the composition of two nonexpansive operators is also nonexpansive.

Definition 3. *We say that $\boldsymbol{x}^* \in \mathbb{R}^n$ is a fixed point of F if $\boldsymbol{x}^* = \mathsf{F}(\boldsymbol{x}^*)$. We denote the set of fixed points of an operator F as $\mathsf{fix}(\mathsf{F}) \triangleq \{\boldsymbol{x} \in \mathbb{R}^n : \boldsymbol{x} = \mathsf{F}(\boldsymbol{x})\}$.*

Note that the iteration of a nonexpansive operator does not necessarily converge. To see this consider a nonexpansive operator $\mathsf{F} = -\mathsf{I}$, where I is the identity. However, the Krasnosel'skii-Mann theorem (see Theorem 5.15 in [48]) states that the iteration of the damped operator $\mathsf{D} \triangleq (1-\alpha)\mathsf{I} + \alpha\mathsf{F}$, for $\alpha \in (0, 1)$, will converge to $\mathsf{fix}(\mathsf{F})$. This idea is further formalized with the definition of the following class of operators.

Definition 4. *For a constant $\alpha \in (0, 1)$, we say that the operator D is α-averaged, if there exists a nonexpansive operator F such that $\mathsf{D} = (1-\alpha)\mathsf{I} + \alpha\mathsf{F}$.*

An important result from convex analysis is that the proximal operator is $(1/2)$-averaged (see p. 132 in [5]). Similarly, when d is convex and has a Lipschitz continuous gradient of constant L, the gradient-step operator G_γ is $(\gamma L/2)$-averaged for any $\gamma \in (0, 2/L)$ (see p. 17 in [49]). As stated next, the composition of two averaged operators is also averaged.

Proposition 6. *Let F_1 be α_1-averaged and F_2 be α_2-averaged. Then, the composite operator $\mathsf{F} \triangleq \mathsf{F}_2 \circ \mathsf{F}_1 = \mathsf{F}_2\mathsf{F}_1$ is*

$$\alpha \triangleq \frac{\alpha_1 + \alpha_2 - 2\alpha_1\alpha_2}{1 - \alpha_1\alpha_2} \tag{19}$$

averaged operator.

Proof. See Proposition 4.44 in [48]. □

The direct consequence of this theorem, is that the composition of the proximal operator and the gradient-step is also an averaged operator. The following classical result was used in Definition 1 and is central for our subsequent analysis.

Proposition 7. *For a nonexpansive operator D and a constant $\alpha \in (0, 1)$, the following are equivalent:*

(a) D is α-averaged.
(b) $(1 - 1/\alpha)\mathsf{I} + (1/\alpha)\mathsf{D}$ is nonexpansive.
(c) For all $\boldsymbol{x}, \boldsymbol{y} \in \mathbb{R}^n$, we have that

$$\|\mathsf{D}(\boldsymbol{x}) - \mathsf{D}(\boldsymbol{y})\|_2^2 \leq \|\boldsymbol{x} - \boldsymbol{y}\|_2^2 - \left(\frac{1-\alpha}{\alpha}\right)\|\boldsymbol{x} - \mathsf{D}(\boldsymbol{x}) - \boldsymbol{y} + \mathsf{D}(\boldsymbol{y})\|_2^2$$

Proof. See Proposition 4.35 in [48]. □

B. Proximal Optimization Algorithms

PGM, its accelerated variant APGM, and ADMM are some of the most widely used algorithms in image reconstruction. They have been extensively discussed and analyzed in literature [5], [13], [67]. In this section, we briefly review their formulation leading directly to Algorithms 1 and 2.

To understand PGM and APGM, consider the optimization problem (1), where both d and r are convex, but where r is possibly non-differentiable. The iterates of PGM can then be expressed as

$$\boldsymbol{x}^k \leftarrow \mathsf{prox}_{\gamma r}(\boldsymbol{x}^{k-1} - \gamma \nabla d(\boldsymbol{x}^{k-1})) \tag{20}$$

where $\gamma > 0$ is the step-size. Hence, PGM first computes a gradient-descent step with respect to d and then evaluates the proximal operator of r defined in (4). When ∇d is Lipschitz continuous with constant $L > 0$, PGM can be shown to converge for any $\gamma \in (0, 1/L]$ to a minimizer of the objective function with rate $O(1/t)$, where $t \geq 1$ is the number of PGM iterations [68]. APGM is an extension of PGM that includes an additional extrapolation step in each iteration

$$\boldsymbol{x}^k \leftarrow \mathsf{prox}_{\gamma r}(\boldsymbol{s}^{k-1} - \gamma \nabla d(\boldsymbol{s}^{k-1})) \tag{21a}$$
$$\boldsymbol{s}^k \leftarrow \boldsymbol{x}^k + \beta_k(\boldsymbol{x}^k - \boldsymbol{x}^{k-1}), \tag{21b}$$

where $\beta_k \in [0, 1)$ is an extrapolation parameter. It is clear that when $\beta_k = 0$ for all $k \geq 1$, PGM and APGM are perfectly equivalent. On the other hand, when $\{\beta_k\}$ are selected in specific ways, one can accelerate the convergence of the algorithm [68]. In such accelerated settings, it is possible to show that APGM converges to the minimizer of the objective function f with rate $O(1/t^2)$ for any step-size $\gamma \in (0, 1/L]$.

To develop ADMM, we consider the following optimization problem over $(\boldsymbol{x}, \boldsymbol{z})$ equivalent to (1)

$$\text{minimize} \quad d(\boldsymbol{z}) + r(\boldsymbol{x}) \text{ subject to } \boldsymbol{z} = \boldsymbol{x}. \tag{22}$$

This process of introducing an additional variable $\boldsymbol{z}$ is known as *variable splitting*. To solve this constrained optimization problem, we form the *augmented Lagrangian* [69]

$$\begin{aligned} &L_\gamma(\boldsymbol{z}, \boldsymbol{x}, \boldsymbol{\mu}) \\ &= d(\boldsymbol{z}) + r(\boldsymbol{x}) + \boldsymbol{\mu}^\mathsf{T}(\boldsymbol{z} - \boldsymbol{x}) + \frac{1}{2\gamma}\|\boldsymbol{z} - \boldsymbol{x}\|_2^2 \qquad (23a) \\ &= d(\boldsymbol{z}) + r(\boldsymbol{x}) + \frac{1}{2\gamma}\|\boldsymbol{z} - \boldsymbol{x} + \gamma\boldsymbol{\mu}\|_2^2 - \frac{\gamma}{2}\|\boldsymbol{\mu}\|_2^2, \qquad (23b) \end{aligned}$$

where $\gamma > 0$ is a parameter and $\boldsymbol{\mu} \in \mathbb{R}^n$ is the dual variable. We can re-write the augmented Lagrangian by introducing the *scaled dual variable* $\boldsymbol{s} \triangleq \gamma\boldsymbol{\mu}$, which leads to

$$L_\gamma(\boldsymbol{z}, \boldsymbol{x}, \boldsymbol{s}) = d(\boldsymbol{z}) + r(\boldsymbol{x}) + \frac{1}{2\gamma}\|\boldsymbol{z} - \boldsymbol{x} + \boldsymbol{s}\|_2^2 - \frac{1}{2\gamma}\|\boldsymbol{s}\|_2^2. \tag{24}$$

This optimization problem can be solved via the *method of multipliers* [69] that has the following form for $k \geq 1$

$$(\boldsymbol{z}^k, \boldsymbol{x}^k) \leftarrow \underset{\boldsymbol{x}, \boldsymbol{z}}{\mathsf{arg\,min}} \left\{ L_\gamma(\boldsymbol{z}, \boldsymbol{x}, \boldsymbol{s}^{k-1}) \right\} \tag{25a}$$
$$\boldsymbol{s}^k \leftarrow \boldsymbol{s}^{k-1} + (\boldsymbol{z}^k - \boldsymbol{x}^k), \tag{25b}$$

starting from $\boldsymbol{s}^0 = \boldsymbol{0}$. Note, however, the difficulty of running this algorithm due to the need to jointly minimize over both $\boldsymbol{z}$ and $\boldsymbol{x}$. ADMM precisely circumvents this issue by splitting this step into two as follows

$$\boldsymbol{z}^k \leftarrow \underset{\boldsymbol{z} \in \mathbb{R}^n}{\mathsf{arg\,min}} \left\{ L_\gamma(\boldsymbol{z}, \boldsymbol{x}^{k-1}, \boldsymbol{s}^{k-1}) \right\} \tag{26a}$$
$$\boldsymbol{x}^k \leftarrow \underset{\boldsymbol{x} \in \mathbb{R}^n}{\mathsf{arg\,min}} \left\{ L_\gamma(\boldsymbol{z}^k, \boldsymbol{x}, \boldsymbol{s}^{k-1}) \right\} \tag{26b}$$
$$\boldsymbol{s}^k \leftarrow \boldsymbol{s}^{k-1} + (\boldsymbol{z}^k - \boldsymbol{x}^k), \tag{26c}$$

which directly leads to Algorithm 2.

C. Proof of Proposition 1

Proposition 1 is a direct consequence of the well-known fixed-point interpretation of PGM (see p. 150 in [5]). We provide the proof here for completeness by using the following characterization of the proximal operator

$$\boldsymbol{x} = \mathsf{prox}_{\gamma r}(\boldsymbol{z}) \quad \Leftrightarrow \quad \frac{\boldsymbol{z} - \boldsymbol{x}}{\gamma} \in \partial r(\boldsymbol{x}), \tag{27}$$

valid for all $\boldsymbol{z} \in \mathbb{R}^n$, where $\partial r(\boldsymbol{x})$ is the subdifferential of r at $\boldsymbol{x}$ [70]. Let $\mathsf{denoise}_\sigma(\cdot) = \mathsf{prox}_{\gamma r}(\cdot)$ and $\boldsymbol{x}^* \in \mathsf{fix}(\mathsf{P})$. Then, from (27), we have that

$$\begin{aligned} \boldsymbol{x}^* = \mathsf{P}(\boldsymbol{x}^*) &= \mathsf{prox}_{\gamma r}(\boldsymbol{x}^* - \gamma \nabla d(\boldsymbol{x}^*)) \\ &\Leftrightarrow \quad -\nabla d(\boldsymbol{x}^*) \in \partial r(\boldsymbol{x}^*) \\ &\Leftrightarrow \quad \boldsymbol{0} \in \nabla d(\boldsymbol{x}^*) + \partial r(\boldsymbol{x}^*), \end{aligned}$$

which establishes the desired result.

D. Proof of Proposition 2

As mentioned in Appendix A, the iterative application of an averaged operator is well known as Krasnosel'skii-Mann iteration [71], [72] and its convergence has been extensively discussed in literature [48], [49], [60]. Below, we use this theory to establish a novel convergence result for PnP-PGM.

From our assumptions, the denoiser D_σ is θ-averaged and the gradient-step operator G_γ is $(\gamma L/2)$-averaged for any $\gamma \in (0, 2/L)$. From Proposition 6, we have that their composition $\mathsf{P} = \mathsf{D}_\sigma \circ \mathsf{G}_\gamma$ is

$$\alpha = \frac{\theta + \frac{\gamma L}{2} - \theta\gamma L}{1 - \frac{\theta\gamma L}{2}}$$

averaged. Consider a single iteration $\boldsymbol{x}^+ = \mathsf{P}(\boldsymbol{x})$, then we have for any $\boldsymbol{x}^* \in \mathsf{fix}(\mathsf{P})$ that

$$\begin{aligned} \|\boldsymbol{x}^+ - \boldsymbol{x}^*\|_2^2 &= \|\mathsf{P}(\boldsymbol{x}) - \mathsf{P}(\boldsymbol{x}^*)\|_2^2 \\ &\leq \|\boldsymbol{x} - \boldsymbol{x}^*\|_2^2 - \left(\frac{1-\alpha}{\alpha}\right)\|\boldsymbol{x} - \mathsf{P}(\boldsymbol{x}) - \boldsymbol{x}^* + \mathsf{P}(\boldsymbol{x}^*)\|_2^2 \\ &= \|\boldsymbol{x} - \boldsymbol{x}^*\|_2^2 - \left(\frac{1-\alpha}{\alpha}\right)\|\boldsymbol{x} - \mathsf{P}(\boldsymbol{x})\|_2^2, \end{aligned}$$

where we used Proposition 7(c) and the fact that $\boldsymbol{x}^* = \mathsf{P}(\boldsymbol{x}^*)$. By considering the iteration $k \geq 1$ and rearranging the terms, we obtain

$$\begin{aligned} &\|\boldsymbol{x}^{k-1} - \mathsf{P}(\boldsymbol{x}^{k-1})\|_2^2 \\ &\qquad \leq \left(\frac{\alpha}{1-\alpha}\right)\left[\|\boldsymbol{x}^{k-1} - \boldsymbol{x}^*\|_2^2 - \|\boldsymbol{x}^k - \boldsymbol{x}^*\|_2^2\right]. \end{aligned}$$

By averaging this inequality over $t \geq 1$ iterations and dropping the last term $\|\boldsymbol{x}^t - \boldsymbol{x}^*\|_2^2$, we obtain

$$\frac{1}{t}\sum_{k=1}^{t} \|\boldsymbol{x}^{k-1} - \mathsf{P}(\boldsymbol{x}^{k-1})\|_2^2 \leq \frac{1}{t}\left(\frac{\alpha}{1-\alpha}\right)\|\boldsymbol{x}^0 - \boldsymbol{x}^*\|_2^2.$$

To obtain the result that depends on $\theta \in (0,1)$, we note that for any $\gamma \in (0, 1/L]$, we can write

$$\frac{\alpha}{1-\alpha} = \frac{\theta + \frac{\gamma L}{2} - \theta\gamma L}{(1-\theta)(1-\frac{\gamma L}{2})} \leq \frac{\theta + \frac{1}{2}}{\frac{1-\theta}{2}} \leq 2\left(\frac{1+\theta}{1-\theta}\right). \quad (28)$$

To express the result as in (12), simply take the minimum of $\|\boldsymbol{x}^{k-1} - \mathsf{P}(\boldsymbol{x}^{k-1})\|_2^2$ over a window of past $t \geq 1$ iterations out of the sum to form a lower bound. The desired result is obtained by rearranging the terms.

E. Proof of Proposition 3

Proposition 3 is a variant of the result in [25]. For completeness, we provide a proof based on the fixed-point interpretation of ADMM (see p. 157 in [5]).

First note that both D_σ and $\mathsf{prox}_{\gamma d}$ are continuous (since they are nonexpansive). Fixed points $\boldsymbol{x}^*, \boldsymbol{z}^*, \boldsymbol{s}^*$ of PnP-ADMM satisfy

$$\boldsymbol{z}^* = \mathsf{prox}_{\gamma d}(\boldsymbol{x}^* - \boldsymbol{s}^*) \quad (29a)$$
$$\boldsymbol{x}^* = \mathsf{D}_\sigma(\boldsymbol{z}^* + \boldsymbol{s}^*) \quad (29b)$$
$$\boldsymbol{s}^* = \boldsymbol{s}^* + \boldsymbol{z}^* - \boldsymbol{x}^*. \quad (29c)$$

From (29c), we conclude that $\boldsymbol{z}^* = \boldsymbol{x}^*$. By using the smoothness of d and the characterization (27) in (29a), we obtain

$$\boldsymbol{x}^* - \boldsymbol{s}^* - \boldsymbol{z}^* = \gamma\nabla d(\boldsymbol{z}^*) \quad \Rightarrow \quad \boldsymbol{s}^* = -\gamma\nabla d(\boldsymbol{x}^*).$$

Finally, by using this in (29b), we obtain

$$\boldsymbol{x}^* = \mathsf{D}_\sigma(\boldsymbol{x}^* - \gamma\nabla d(\boldsymbol{x}^*)) = \mathsf{P}(\boldsymbol{x}^*),$$

which means that $\boldsymbol{x}^* = \boldsymbol{z}^* \in \mathsf{fix}(\mathsf{P})$ and completes the proof.

F. Proof of Proposition 4

We prove by providing a specific counter example. For simplicity, we assume $n = 1$, but the same example can be generalized for any $n \in \mathbb{N}$. Consider the data fidelity given by the Huber function

$$d(x) \triangleq \begin{cases} \frac{1}{2}x^2 & \text{if } |x| \leq 1 \\ |x| - \frac{1}{2} & \text{if } |x| > 1 \end{cases}. \quad (30)$$

This function is convex and has a Lipschitz continuous gradient with constant $L = 1$

$$d'(x) = \begin{cases} x & \text{if } |x| \leq 1 \\ \mathsf{sgn}(x) & \text{if } |x| > 1 \end{cases}, \quad (31)$$

where $\mathsf{sgn}(\cdot)$ denotes the sign function. We also consider the denoiser defined as

$$\mathsf{D}_\sigma(z) \triangleq z + \sigma\sqrt{c}\,\mathsf{sgn}(z), \quad (32)$$

where $c > 0$ is some constant independent of $\sigma > 0$. Since

$$|\mathsf{D}_\sigma(x) - x|^2 = \sigma^2\, c, \quad (33)$$

this denoiser satisfies the definition of boundedness in (8). Then, for $q_k = 1$, a single iteration of PnP-PGM can be rewritten as

$$x = \mathsf{D}_\sigma(z) = z + \sigma\sqrt{c}\,\mathsf{sgn}(z)$$
$$z^+ = x - \gamma d'(x) = \begin{cases} (1-\gamma)x & \text{if } |x| \leq 1 \\ x - \gamma\mathsf{sgn}(x) & \text{if } |x| > 1 \end{cases},$$

where we assume any $\gamma \in (0,1)$. By combining these equations, we obtain

$$z^+ = \begin{cases} (1-\gamma)(|z| + \sigma\sqrt{c})\mathsf{sgn}(z) & \text{if } |z| \leq 1 - \sigma\sqrt{c} \\ (|z| + \sigma\sqrt{c} - \gamma)\mathsf{sgn}(z) & \text{if } |z| > 1 - \sigma\sqrt{c}, \end{cases}$$

where we used the fact that $\mathsf{sgn}(x) = \mathsf{sgn}(z)$ and expressed $z = |z|\,\mathsf{sgn}(z)$. For $|z| \leq 1 - \sigma\sqrt{c}$, we have that

$$\begin{aligned} |z^+| &= (1-\gamma)(|z| + \sigma\sqrt{c}) \\ &= |z| + \sigma\sqrt{c} - \gamma|z| - \gamma\sigma\sqrt{c} \\ &\geq |z| + \sigma\sqrt{c} - \gamma(1 - \sigma\sqrt{c}) - \gamma\sigma\sqrt{c} \\ &= |z| + \sigma\sqrt{c} - \gamma. \end{aligned}$$

On the other hand, for $|z| > 1 - \sigma\sqrt{c}$, we have that

$$|z^+| = |z| + \sigma\sqrt{c} - \gamma.$$

This means that the iterates of PnP-PGM satisfy

$$|z^t| \geq |z^0| + t(\sigma\sqrt{c} - \gamma), \quad \forall t \in \mathbb{N}.$$

Therefore, for any $\sigma > \gamma/\sqrt{c}$ and any $z^0 \in \mathbb{R}$, the sequence $\{z^t\}_{t\in\mathbb{N}}$ generated by PnP-PGM diverges. Since the denoiser is bounded, this implies that the sequence $\{x^t\}_{t\in\mathbb{N}}$ also diverges. This completes the proof.

G. Proof of Proposition 5

We define the full proximal-gradient operator

$$\mathsf{P}(\boldsymbol{x}) \triangleq \mathsf{D}_\sigma(\boldsymbol{x} - \gamma\nabla d(\boldsymbol{x})) \quad (34)$$

and its online variant over a minibatch of size $B \geq 1$

$$\widehat{\mathsf{P}}(\boldsymbol{x}) \triangleq \mathsf{D}_\sigma(\boldsymbol{x} - \gamma\hat{\nabla} d(\boldsymbol{x})), \quad (35)$$

where $\hat{\nabla} d$ denotes the minibatch gradient. The variance bound in Assumption 2(d) implies that for all $\boldsymbol{x} \in \mathbb{R}^n$, we have that

$$\begin{aligned} &\mathbb{E}\left[\|\mathsf{P}(\boldsymbol{x}) - \widehat{\mathsf{P}}(\boldsymbol{x})\|_2^2\right] \\ &= \mathbb{E}\left[\|\mathsf{D}_\sigma(\boldsymbol{x} - \gamma\nabla d(\boldsymbol{x})) - \mathsf{D}_\sigma(\boldsymbol{x} - \gamma\hat{\nabla} d(\boldsymbol{x}))\|_2^2\right] \\ &\leq \mathbb{E}\left[\|\boldsymbol{x} - \gamma\nabla d(\boldsymbol{x}) - \boldsymbol{x} + \gamma\hat{\nabla} d(\boldsymbol{x})\|_2^2\right] \\ &\leq \gamma^2\mathbb{E}\left[\|\nabla d(\boldsymbol{x}) - \hat{\nabla} d(\boldsymbol{x})\|_2^2\right] \leq \frac{\gamma^2\nu^2}{B}, \end{aligned} \quad (36)$$

TABLE III
LIST OF OPTIMAL σ VALUES FOR EACH TEST IMAGE.

Images	PnP-ADMM (10)	PnP-ADMM (20)	PnP-ADMM (30)	PnP-APGM (10)	PnP-APGM (20)	PnP-APGM (30)	PnP-SPGM (10)	PnP-SPGM (20)	PnP-SPGM (30)
Barbara	8.13e-4	7.26e-4	9.62e-4	2.78e-2	8.83e-4	1.66e-3	5.74e-4	8.71e-4	1.11e-3
Boat	1.02e-3	1.17e-3	1.19e-3	5.74e-2	2.00e-3	1.62e-3	7.03e-4	9.74e-4	1.28e-3
Foreman	1.82e-3	1.32e-3	1.07e-3	1.72e-2	1.74e-3	1.83e-3	7.40e-4	1.05e-3	1.32e-3
House	2.68e-3	2.58e-3	1.71e-3	4.35e-2	1.72e-3	2.03e-3	6.97e-4	1.12e-3	1.32e-3
Lenna	1.27e-3	1.18e-3	9.60e-4	1.66e-3	2.14e-3	1.65e-3	6.30e-4	9.45e-4	1.17e-3
Monarch	1.32e-3	2.20e-3	1.77e-3	6.20e-2	2.13e-3	1.85e-3	6.61e-4	9.45e-4	1.23e-3
Parrot	1.69e-3	1.38e-3	1.64e-3	6.94e-2	2.16e-3	1.76e-3	6.19e-4	8.63e-4	1.19e-3
Pepper	8.98e-3	1.35e-3	1.11e-3	3.84e-2	1.60e-3	1.79e-3	6.61e-4	9.74e-4	1.19e-3

where in the third row we used the nonexpansiveness of D_σ. Consider a single iteration $\boldsymbol{x}^k = \widehat{\mathsf{P}}(\boldsymbol{x}^{k-1})$, then we have for any $\boldsymbol{x}^* \in \mathsf{fix}(\mathsf{P})$ that

$$\begin{aligned}
&\|\boldsymbol{x}^k - \boldsymbol{x}^*\|_2^2 = \|\widehat{\mathsf{P}}(\boldsymbol{x}^{k-1}) - \mathsf{P}(\boldsymbol{x}^{k-1}) + \mathsf{P}(\boldsymbol{x}^{k-1}) - \mathsf{P}(\boldsymbol{x}^*)\|_2^2 \\
&= \|\mathsf{P}(\boldsymbol{x}^{k-1}) - \mathsf{P}(\boldsymbol{x}^*)\|_2^2 + \|\widehat{\mathsf{P}}(\boldsymbol{x}^{k-1}) - \mathsf{P}(\boldsymbol{x}^{k-1})\|_2^2 \qquad (37)\\
&\quad + 2(\widehat{\mathsf{P}}(\boldsymbol{x}^{k-1}) - \mathsf{P}(\boldsymbol{x}^{k-1}))^{\mathsf{T}}(\mathsf{P}(\boldsymbol{x}^{k-1}) - \mathsf{P}(\boldsymbol{x}^*)) \\
&\leq \|\boldsymbol{x}^{k-1} - \boldsymbol{x}^*\|_2^2 - \left(\frac{1-\alpha}{\alpha}\right)\|\boldsymbol{x}^{k-1} - \mathsf{P}(\boldsymbol{x}^{k-1})\|_2^2 \\
&\quad + \|\widehat{\mathsf{P}}(\boldsymbol{x}^{k-1}) - \mathsf{P}(\boldsymbol{x}^{k-1})\|_2^2 \\
&\quad + 2\|\widehat{\mathsf{P}}(\boldsymbol{x}^{k-1}) - \mathsf{P}(\boldsymbol{x}^{k-1})\|_2 \cdot \|\mathsf{P}(\boldsymbol{x}^{k-1}) - \mathsf{P}(\boldsymbol{x}^*)\|_2,
\end{aligned}$$

where we used Proposition 7(c) and the Cauchy-Schwarz inequality. Note that due to nonexpansiveness of the operator P, we have that

$$\|\mathsf{P}(\boldsymbol{x}^{k-1}) - \mathsf{P}(\boldsymbol{x}^*)\|_2 \leq \|\boldsymbol{x}^{k-1} - \boldsymbol{x}^*\|_2 \leq \|\boldsymbol{x}^0 - \boldsymbol{x}^*\|_2. \quad (38)$$

Additionally, by applying Jensen's inequality to (36), we conclude that for all $\boldsymbol{x} \in \mathbb{R}^n$

$$\begin{aligned}
&\mathbb{E}\left[\|\mathsf{P}(\boldsymbol{x}) - \widehat{\mathsf{P}}(\boldsymbol{x})\|_2\right] = \mathbb{E}\left[\sqrt{\|\mathsf{P}(\boldsymbol{x}) - \widehat{\mathsf{P}}(\boldsymbol{x})\|_2^2}\right] \qquad (39)\\
&\leq \sqrt{\mathbb{E}\left[\|\mathsf{P}(\boldsymbol{x}) - \widehat{\mathsf{P}}(\boldsymbol{x})\|_2^2\right]} \leq \frac{\gamma\nu}{\sqrt{B}}. \qquad (40)
\end{aligned}$$

By taking a conditional expectation of (37) and using these bounds, we obtain

$$\begin{aligned}
&\mathbb{E}\left[\|\boldsymbol{x}^k - \boldsymbol{x}^*\|_2^2 - \|\boldsymbol{x}^{k-1} - \boldsymbol{x}^*\|_2^2 \mid \boldsymbol{x}^{k-1}\right] \\
&\leq \left(\frac{\alpha-1}{\alpha}\right)\|\boldsymbol{x}^{k-1} - \mathsf{P}(\boldsymbol{x}^{k-1})\|_2^2 \\
&\quad + \frac{2\gamma\nu}{\sqrt{B}}\|\boldsymbol{x}^0 - \boldsymbol{x}^*\|_2 + \frac{\gamma^2\nu^2}{B},
\end{aligned}$$

which can be rearanged into

$$\begin{aligned}
&\|\boldsymbol{x}^{k-1} - \mathsf{P}(\boldsymbol{x}^{t-1})\|_2^2 \\
&\leq \left(\frac{\alpha}{1-\alpha}\right)\Big[\frac{\gamma^2\nu^2}{B} + \frac{2\gamma\nu}{\sqrt{B}}\|\boldsymbol{x}^0 - \boldsymbol{x}^*\|_2 \\
&\quad + \mathbb{E}\left[\|\boldsymbol{x}^{k-1} - \boldsymbol{x}^*\|_2^2 - \|\boldsymbol{x}^k - \boldsymbol{x}^*\|_2^2 \mid \boldsymbol{x}^{k-1}\right]\Big].
\end{aligned}$$

By averaging the inequality over $t \geq 1$ iterations, taking the total expectation, and dropping the last term, we obtain

$$\begin{aligned}
&\mathbb{E}\left[\frac{1}{t}\sum_{k=1}^{t}\|\boldsymbol{x}^{k-1} - \mathsf{P}(\boldsymbol{x}^{k-1})\|_2^2\right] \\
&\quad \leq \frac{\alpha}{1-\alpha}\left[\frac{\gamma^2\nu^2}{B} + \frac{2\gamma\nu}{\sqrt{B}}\|\boldsymbol{x}^0 - \boldsymbol{x}^*\|_2 + \frac{\|\boldsymbol{x}^0 - \boldsymbol{x}^*\|_2^2}{t}\right],
\end{aligned}$$

where we used the law of total expectation. By using the inequality (28), we can rewrite this expression as

$$\begin{aligned}
&\mathbb{E}\left[\frac{1}{t}\sum_{k=1}^{t}\|\boldsymbol{x}^{k-1} - \mathsf{P}(\boldsymbol{x}^{k-1})\|_2^2\right] \\
&\quad \leq 2\left(\frac{1+\theta}{1-\theta}\right)\left[\frac{\gamma^2\nu^2}{B} + \frac{2\gamma\nu}{\sqrt{B}}\|\boldsymbol{x}^0 - \boldsymbol{x}^*\|_2 + \frac{\|\boldsymbol{x}^0 - \boldsymbol{x}^*\|_2^2}{t}\right]
\end{aligned}$$

Note that to obtain the results in Corollary 1, simply replace given values for γ and B into the inequality, and use the following bounds that are valid for any $t \in \mathbb{N}$

$$\frac{1}{t} \leq \frac{1}{\sqrt{t}} \quad \text{and} \quad \frac{1}{t^2} \leq \frac{1}{t}.$$

This establishes the desired results.

H. List of Selected Hyperparameters

We optimized the algorithmic hyperparameters of PnP-SPGM, PnP-APGM, and PnP-ADMM for each DT reconstruction with the fixed per-iteration budget of measurements. The $\gamma > 0$ was empirically evaluated at 300 iterations by using APGM with the backtracking selection of step-size. The parameter $\rho > 0$ of PnP-ADMM is fixed to 1×10^{-3} in favor of a better searching range of $\sigma > 0$, which controls strength of denoising. Table III lists the optimal σ for each algorithm for the reconstruction of every test image.

REFERENCES

[1] L. I. Rudin, S. Osher, and E. Fatemi, "Nonlinear total variation based noise removal algorithms," *Physica D*, vol. 60, no. 1–4, pp. 259–268, November 1992.

[2] M. A. T. Figueiredo and R. D. Nowak, "Wavelet-based image estimation: An empirical Bayes approach using Jeffreys' noninformative prior," *IEEE Trans. Image Process.*, vol. 10, no. 9, pp. 1322–1331, September 2001.

[3] M. Elad and M. Aharon, "Image denoising via sparse and redundant representations over learned dictionaries," *IEEE Trans. Image Process.*, vol. 15, no. 12, pp. 3736–3745, December 2006.

[4] A. Danielyan, V. Katkovnik, and K. Egiazarian, "BM3D frames and variational image deblurring," *IEEE Trans. Image Process.*, vol. 21, no. 4, pp. 1715–1728, April 2012.

[5] N. Parikh and S. Boyd, "Proximal algorithms," *Foundations and Trends in Optimization*, vol. 1, no. 3, pp. 123–231, 2014.

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

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

[8] J. Bect, L. Blanc-Feraud, G. Aubert, and A. Chambolle, "A ℓ_1-unified variational framework for image restoration," in *Proc. ECCV*, Springer, Ed., vol. 3024, New York, 2004, pp. 1–13.

[9] A. Beck and M. Teboulle, "Fast gradient-based algorithm for constrained total variation image denoising and deblurring problems," *IEEE Trans. Image Process.*, vol. 18, no. 11, pp. 2419–2434, November 2009.

[10] J. Eckstein and D. P. Bertsekas, "On the Douglas-Rachford splitting method and the proximal point algorithm for maximal monotone operators," *Mathematical Programming*, vol. 55, pp. 293–318, 1992.

[11] M. V. Afonso, J. M.Bioucas-Dias, and M. A. T. Figueiredo, "Fast image recovery using variable splitting and constrained optimization," *IEEE Trans. Image Process.*, vol. 19, no. 9, pp. 2345–2356, September 2010.

[12] M. K. Ng, P. Weiss, and X. Yuan, "Solving constrained total-variation image restoration and reconstruction problems via alternating direction methods," *SIAM J. Sci. Comput.*, vol. 32, no. 5, pp. 2710–2736, August 2010.

[13] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, "Distributed optimization and statistical learning via the alternating direction method of multipliers," *Foundations and Trends in Machine Learning*, vol. 3, no. 1, pp. 1–122, 2011.

[14] S. V. Venkatakrishnan, C. A. Bouman, and B. Wohlberg, "Plug-and-play priors for model based reconstruction," in *Proc. IEEE Global Conf. Signal Process. and Inf. Process. (GlobalSIP)*, Austin, TX, USA, December 3-5, 2013, pp. 945–948.

[15] 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. 16, pp. 2080–2095, August 2007.

[16] S. Gu, L. Zhang, W. Zuo, and X. Feng, "Weighted nuclear norm minimization with application to image denoising," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition (CVPR)*, Columbus, OH, USA, September 30-October 3, 2014, pp. 2862–2869.

[17] Y. Chen and T. Pock, "Trainable nonlinear reaction diffusion: A flexible framework for fast and effective image restoration," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 6, pp. 1256–1272, June 2017.

[18] S. Sreehari, S. V. Venkatakrishnan, B. Wohlberg, G. T. Buzzard, L. F. Drummy, J. P. Simmons, and C. A. Bouman, "Plug-and-play priors for bright field electron tomography and sparse interpolation," *IEEE Trans. Comp. Imag.*, vol. 2, no. 4, pp. 408–423, December 2016.

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

[20] A. Brifman, Y. Romano, and M. Elad, "Turning a denoiser into a super-resolver using plug and play priors," in *Proc. IEEE Int. Conf. Image Proc. (ICIP)*, Phoenix, AZ, USA, September 25-28, 2016, pp. 1404–1408.

[21] A. M. Teodoro, J. M. Biocas-Dias, and M. A. T. Figueiredo, "Image restoration and reconstruction using variable splitting and class-adapted image priors," in *Proc. IEEE Int. Conf. Image Proc. (ICIP)*, Phoenix, AZ, USA, September 25-28, 2016, pp. 3518–3522.

[22] K. Zhang, W. Zuo, S. Gu, and L. Zhang, "Learning deep CNN denoiser prior for image restoration," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition (CVPR)*, Honolulu, USA, July 21-26, 2017, pp. 3929–3938.

[23] A. Teodoro, J. M. Bioucas-Dias, and M. Figueiredo, "Scene-adapted plug-and-play algorithm with convergence guarantees," in *Proc. IEEE Int. Workshop on Machine Learning for Signal Processing*, Tokyo, Japan, September 25-28, 2017.

[24] S. Ono, "Primal-dual plug-and-play image restoration," *IEEE Signal. Proc. Let.*, vol. 24, no. 8, pp. 1108–1112, 2017.

[25] T. Meinhardt, M. Moeller, C. Hazirbas, and D. Cremers, "Learning proximal operators: Using denoising networks for regularizing inverse imaging problems," in *Proc. IEEE Int. Conf. Comp. Vis. (ICCV)*, Venice, Italy, October 22-29, 2017, pp. 1799–1808.

[26] U. S. Kamilov, H. Mansour, and B. Wohlberg, "A plug-and-play priors approach for solving nonlinear imaging inverse problems," *IEEE Signal. Proc. Let.*, vol. 24, no. 12, pp. 1872–1876, December 2017.

[27] L. Bottou and O. Bousquet, "The tradeoffs of large scale learning," in *Proc. Advances in Neural Information Processing Systems (NIPS)*, Vancouver, BC, Canada, December 3-6, 2007, pp. 161–168.

[28] U. S. Kamilov, I. N. Papadopoulos, M. H. Shoreh, A. Goy, C. Vonesch, M. Unser, and D. Psaltis, "Optical tomographic image reconstruction based on beam propagation and sparse regularization," *IEEE Trans. Comp. Imag.*, vol. 2, no. 1, pp. 59–70,, March 2016.

[29] K. Degraux, U. S. Kamilov, P. T. Boufounos, and D. Liu, "Online convolutional dictionary learning for multimodal imaging," in *Proc. IEEE Int. Conf. Image Proc. (ICIP)*, Beijing, China, September 17-20, 2017, pp. 1617–1621.

[30] A. C. Kak and M. Slaney, *Principles of Computerized Tomographic Imaging*. IEEE, 1988.

[31] R. Tibshirani, "Regression and selection via the lasso," *J. R. Stat. Soc. Series B (Methodological)*, vol. 58, no. 1, pp. 267–288, 1996.

[32] 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, February 2006.

[33] D. L. Donoho, "Compressed sensing," *IEEE Trans. Inf. Theory*, vol. 52, no. 4, pp. 1289–1306, April 2006.

[34] J. J. Moreau, "Proximité et dualité dans un espace hilbertien," *Bull. Soc. Math. France*, vol. 93, pp. 273–299, 1965.

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

[36] A. Matakos, S. Ramani, and J. A. Fessler, "Accelerated edge-preserving image restoration without boundary artifacts," *IEEE Trans. Image Process.*, vol. 22, no. 5, pp. 2019–2029, May 2013.

[37] M. Almeida and M. Figueiredo, "Deconvolving images with unknown boundaries using the alternating direction method of multipliers," *IEEE Trans. Ima*, vol. 22, no. 8, pp. 3074–3086, August 2013.

[38] C. Garcia-Cardona and B. Wohlberg, "Convolutional dictionary learning: A comparative review and new algorithms," *IEEE Trans. Comp. Imag.*, vol. 4, no. 3, pp. 366–381, Sep. 2018.

[39] Y. Nesterov, *Introductory Lectures on Convex Optimization: A Basic Course*. Kluwer Academic Publishers, 2004.

[40] M. Zinkevich, "Online convex programming and generalized infinitesimal gradient ascent," in *Proc. 20th Int. Conf. Machine Learning (ICML)*, Washington DC, USA, August 21-24, 2003.

[41] J. Duchi and Y. Singer, "Efficient online and batch learning using forward backward splitting," *J. Mach. Learn. Res*, vol. 10, pp. 2899–2934, 2009.

[42] M. Schmidt, N. Le Roux, and F. Bach, "Convergence rates of inexact proximal-gradient methods for convex optimization," in *Proc. Advances in Neural Information Processing Systems (NIPS)*, Granada, Spain, December 12-15, 2011, pp. 1458–1466.

[43] D. P. Bertsekas, "Incremental proximal methods for large scale convex optimization," *Math. Program. Ser. B*, vol. 129, pp. 163–195, 2011.

[44] O. Devolder, F. Glineur, and Y. Nesterov, "First-order methods of smooth convex optimization with inexact oracle," *Math. Program. Ser. A*, vol. 146, no. 1-2, pp. 37–75, 2013.

[45] U. S. Kamilov, E. Bostan, and M. Unser, "Variational justification of cycle spinning for wavelet-based solutions of inverse problems," *IEEE Signal Process. Lett.*, vol. 21, no. 11, pp. 1326–1330, November 2014.

[46] S. Ghadimi and G. Lan, "Accelerated gradient methods for nonconvex nonlinear and stochastic programming," *Math. Program. Ser. A*, vol. 156, no. 1, pp. 59–99, March 2016.

[47] U. S. Kamilov, "A parallel proximal algorithm for anisotropic total variation minimization," *IEEE Trans. Image Process.*, vol. 26, no. 2, pp. 539–548, February 2017.

[48] H. H. Bauschke and P. L. Combettes, *Convex Analysis and Monotone Operator Theory in Hilbert Spaces*, 2nd ed. Springer, 2010.

[49] E. K. Ryu and S. Boyd, "A primer on monotone operator methods," *Appl. Comput. Math.*, vol. 15, no. 1, pp. 3–43, 2016.

[50] C. A. Metzler, A. Maleki, and R. G. Baraniuk, "From denoising to compressed sensing," *IEEE Trans. Inf. Theory*, vol. 62, no. 9, pp. 5117–5144, September 2016.

[51] C. A. Metzler, A. Maleki, and R. Baraniuk, "BM3D-PRGAMP: Compressive phase retrieval based on BM3D denoising," in *Proc. IEEE Int. Conf. Image Proc. (ICIP)*, Phoenix, AZ, USA, September 25-28, 2016, pp. 2504–2508.

[52] A. Fletcher, S. Rangan, S. Sarkar, and P. Schniter, "Plug-in estimation in high-dimensional linear inverse problems: A rigorous analysis," 2018, arXiv:1806.10466 [cs.IT].

[53] D. L. Donoho, A. Maleki, and A. Montanari, "Message-passing algorithms for compressed sensing," *Proc. Nat. Acad. Sci.*, vol. 106, no. 45, pp. 18 914–18 919, November 2009.

[54] M. Bayati and A. Montanari, "The dynamics of message passing on dense graphs, with applications to compressed sensing," *IEEE Trans. Inf. Theory*, vol. 57, no. 2, pp. 764–785, February 2011.
[55] F. Caltagirone, L. Zdeborová, and F. Krzakala, "On convergence of approximate message passing," in *Proc. IEEE Int. Symp. Information Theory*, Honolulu, HI, USA, June 29-July 4, 2014, pp. 1812–1816.
[56] S. Rangan, P. Schniter, and A. Fletcher, "On the convergence of approximate message passing with arbitrary matrices," in *Proc. IEEE Int. Symp. Information Theory*, Honolulu, HI, USA, June 29-July 4, 2014, pp. 236–240.
[57] S. Rangan, A. K. Fletcher, P. Schniter, and U. S. Kamilov, "Inference for generalized linear models via alternating directions and bethe free energy minimization," *IEEE Trans. Inf. Theory*, vol. 63, no. 1, pp. 676–697, January 2017.
[58] 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.
[59] E. T. Reehorst and P. Schniter, "Regularization by denoising: Clarifications and new interpretations," 2018, arXiv:1806.02296 [cs.CV].
[60] 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, September 2018.
[61] J. Bernstein, Y.-X. Wang, K. Azizzadenesheli, and A. Anandkumar, "signSGD: Compressed optimization for non-convex problems," in *Proc. 35th Int. Conf. Machine Learning (ICML)*, vol. 80, Stockholm, Sweden, Jul. 2018, pp. 560–569.
[62] U. S. Kamilov, "signProx: One-bit proximal algorithm for nonconvex stochastic optimization," 2018, arXiv:1807.08023 [math.OC].
[63] E. Wolf, "Three-dimensional structure determination of semi-transparent objects from holographic data," *Opt. Commun.*, vol. 1, no. 4, pp. 153–156, September/October 1969.
[64] M. M. Bronstein, A. M. Bronstein, M. Zibulevsky, and H. Azhari, "Reconstruction in diffraction ultrasound tomography using nonuniform FFT," *IEEE Trans. Med. Imag.*, vol. 21, no. 11, pp. 1395–1401, November 2002.
[65] Y. Sung, W. Choi, C. Fang-Yen, K. Badizadegan, R. R. Dasari, and M. S. Feld, "Optical diffraction tomography for high resolution live cell imaging," *Opt. Express*, vol. 17, no. 1, pp. 266–277, December 2009.
[66] H.-Y. Liu, D. Liu, H. Mansour, P. T. Boufounos, L. Waller, and U. S. Kamilov, "SEAGLE: Sparsity-driven image reconstruction under multiple scattering," *IEEE Trans. Comput. Imaging*, vol. 4, no. 1, pp. 73–86, March 2018.
[67] A. Beck, *First-Order Methods in Optimization*, ser. MOS-SIAM Series on Optimization. SIAM, 2017.
[68] A. Beck and M. Teboulle, *Convex Optimization in Signal Processing and Communications*. Cambridge, 2009, ch. Gradient-Based Algorithms with Applications to Signal Recovery Problems, pp. 42–88.
[69] J. Nocedal and S. J. Wright, *Numerical Optimization*, 2nd ed. Springer, 2006.
[70] S. Boyd and L. Vandenberghe, "Subgradients," April 2008, class notes for Convex Optimization II. http://see.stanford.edu/materials/lsocoee364b/01-subgradients_notes.pdf.
[71] W. R. Mann, "Mean value methods in iteration," *Proc. Amer. Math. Soc.*, vol. 4, pp. 506–510, 1953.
[72] M. A. Kasnosel'skii, "Two remarks on the method of successive approximations," *Usp. Mat. Nauk*, vol. 10, no. 1, pp. 123–127, 1955.

Yu Sun received the B.Eng. in electronics and information from Sichuan University (SCU), Chengdu, China, and M.S. in data analytics & statistics from Washington University in St. Louis (WUSTL), St. Louis, USA, in 2015 and 2017, respectively. He is currently pursuing the Ph.D. degree at the Computational Imaging Group (CIG) at WUSTL. His research interests include computational imaging, machine learning, deep learning, and optimization. He is a student member of IEEE.

Brendt Wohlberg received the BSc(Hons) degree in applied mathematics, and the MSc(Applied Science) and PhD degrees in electrical engineering from the University of Cape Town, South Africa, in 1990, 1993 and 1996 respectively. He is currently a staff scientist in Theoretical Division at Los Alamos National Laboratory, Los Alamos, NM. His primary research interest is in regularization methods for signal and image processing inverse problems. He was an associate editor for IEEE TRANSACTIONS ON IMAGE PROCESSING from 2010 to 2014, and for IEEE TRANSACTIONS ON COMPUTATIONAL IMAGING from 2015 to 2017, and was Chair of the Computational Imaging Special Interest Group (now the Computational Imaging Technical Committee) of the IEEE Signal Processing Society from 2015 to 2017.

Ulugbek S. Kamilov (S'11–M'15) is an Assistant Professor and Director of Computational Imaging Group at Washington University in St. Louis. From 2015 to 2017, he was a Research Scientist at Mitsubishi Electric Research Laboratories (MERL), Cambridge, MA, USA. He received the BSc and MSc degrees in Communication Systems, and the PhD degree in Electrical Engineering from EPFL, Switzerland, in 2008, 2011, and 2015, respectively. His main research area is computational imaging with an emphasis on the mathematical and computational aspects of image reconstruction. He is a recipient of the IEEE Signal Processing Society's 2017 Best Paper Award (with V. Goyal and S. Rangan). His PhD thesis was selected as a finalist for EPFL's Doctorate Award in 2016. He is a member of Computational Imaging Technical Committee of IEEE Signal Processing Society since 2016.