Recovery Analysis for Plug-and-Play Priors using the Restricted Eigenvalue Condition

Jiaming Liu
Washington University in St. Louis
jiaming.liu@wustl.edu

M. Salman Asif
University of California, Riverside
sasif@ece.ucr.edu

Brendt Wohlberg
Los Alamos National Laboratory
brendt@ieee.org

Ulugbek S. Kamilov
Washington University in St. Louis
kamilov@wustl.edu

Abstract

The *plug-and-play priors (PnP)* and *regularization by denoising (RED)* methods have become widely used for solving inverse problems by leveraging pre-trained deep denoisers as image priors. While the empirical imaging performance and the theoretical convergence properties of these algorithms have been widely investigated, their recovery properties have not previously been theoretically analyzed. We address this gap by showing how to establish theoretical recovery guarantees for PnP/RED by assuming that the solution of these methods lies near the fixed-points of a deep neural network. We also present numerical results comparing the recovery performance of PnP/RED in compressive sensing against that of recent compressive sensing algorithms based on generative models. Our numerical results suggest that PnP with a pre-trained artifact removal network provides significantly better results compared to the existing state-of-the-art methods.

1 Introduction

Many imaging problems—such as denoising, inpainting, and super-resolution—can be formulated as an *inverse problem* involving the recovery of an image $\boldsymbol{x}^* \in \mathbb{R}^n$ from noisy measurements

$$\boldsymbol{y} = \boldsymbol{A}\boldsymbol{x}^* + \boldsymbol{e} \,, \tag{1}$$

where $\boldsymbol{A} \in \mathbb{R}^{m \times n}$ is the measurement operator and $\boldsymbol{e} \in \mathbb{R}^m$ is the noise. *Compressed sensing (CS)* [1, 2] is a related class of inverse problems that seek to recover a sparse vector $\boldsymbol{x}^*$ from $m < n$ measurements. The sparse recovery is possible under certain assumptions on the measurement matrix, such as the *restricted isometry property (RIP)* [1] or the *restricted eigenvalue condition (REC)* [3, 4]. While traditional CS recovery relies on sparsity-promoting priors, recent work on *compressed sensing using generative models (CSGM)* [5] has broadened this perspective to priors specified through pre-trained generative models. CSGM has prompted a large amount of follow-up work on the design and theoretical analysis of algorithms that can leverage generative models as priors for image recovery [6–9].

Plug-and-play priors (PnP) [10, 11] and *regularization by denoising (RED)* [12] are two methods related to CSGM that can also leverage pre-trained deep models as priors for inverse problems. However, unlike CSGM, the regularization in PnP/RED is not based on restricting the solution to the range of a generative model, but rather on denoising the iterates with an existing *additive white Gaussian noise (AWGN)* removal method. The effectiveness of PnP/RED has been shown in a number of inverse problems [13–18], which has prompted researchers to investigate the theoretical properties and interpretations of PnP/RED algorithms [19–28].

Preprint. Under review.

Despite the rich literature on both PnP/RED and CSGM, the conceptual relationship between these two classes of methods has never been formally investigated. In particular, while PnP/RED algorithms enjoy computational advantages over CSGM by not requiring nonconvex projections onto the range of a generative model, they lack theoretical recovery guarantees available for CSGM. In this paper, we address this gap by presenting the first recovery analysis of PnP/RED under the assumptions of CSGM. We show that if a measurement matrix satisfies a variant of REC from [5] over the range of a denoiser, then the distance of the PnP solutions to the true $\boldsymbol{x}^*$ can be explicitly characterized. We also present conditions under which the solutions of both PnP and RED coincide, providing sufficient conditions for the exact recovery of $\boldsymbol{x}^*$ using both methodologies. Our results highlight that the regularization in PnP/RED is achieved by giving preference to images near the *fixed points* of pre-trained deep neural networks. Besides new theory, this paper also presents numerical results directly comparing the recovery performance of PnP/RED against the recent algorithms in compressed sensing from random projections and subsampled Fourier measurements. These numerical results lead to new insights highlighting the excellent recovery performance of both PnP and RED, as well as the benefit of using priors specified as pre-trained *artifact removal (AR)* operators rather than AWGN denoisers.

All proofs and some technical details that have been omitted for space appear in the appendix, which also provides more background and simulations.

2 Background

Inverse problems. A common approach to estimate $\boldsymbol{x}^*$ in (1) is to solve an optimization problem:

$$\min_{\boldsymbol{x}\in\mathbb{R}^n} g(\boldsymbol{x}) + h(\boldsymbol{x}) \quad \text{with} \quad g(\boldsymbol{x}) = \frac{1}{2}\|\boldsymbol{y} - \boldsymbol{A}\boldsymbol{x}\|_2^2 \,, \tag{2}$$

where g is a data-fidelity term that quantifies consistency with the observed data $\boldsymbol{y}$ and h is a regularizer that encodes prior knowledge on $\boldsymbol{x}$. For example, a widely-used regularizer in inverse problems is the nonsmooth *total variation (TV)* function $h(\boldsymbol{x}) = \tau\|\boldsymbol{D}\boldsymbol{x}\|_1$, where $\boldsymbol{D}$ is the gradient operator and $\tau > 0$ is the regularization parameter [29–31].

Compressed sensing using generative models. Generative priors have recently become popular for solving inverse problems [5], which typically require solving the following optimization problem:

$$\min_{\boldsymbol{z}\in\mathbb{R}^k} \frac{1}{2}\|\boldsymbol{y} - \boldsymbol{A}\mathsf{W}(\boldsymbol{z})\|_2^2 \,, \tag{3}$$

where $\mathsf{W} : \mathbb{R}^k \to \mathsf{Im}(\mathsf{W})$ is a pre-trained generative model, such as StyleGAN-2 [32, 33]. The set $\mathsf{Im}(\mathsf{W}) \subseteq \mathbb{R}^n$ is the image set (or the range set) of the generator W. In the past few years, several algorithms have been proposed for solving this optimizatio problem [6–9], including the recent algorithms *PULSE* [34] and *intermediate layer optimization (ILO)* [35] that can recover highly-realistic images. The recovery analysis of CSGM was performed under the assumption that $\boldsymbol{A}$ satisfies the *set-restricted eigenvalue condition (S-REC)* [5] over the range of the generative model:

$$\|\boldsymbol{A}\boldsymbol{x} - \boldsymbol{A}\boldsymbol{z}\|_2^2 \geq \mu\|\boldsymbol{x} - \boldsymbol{z}\|_2^2 - \eta \quad \forall \boldsymbol{x}, \boldsymbol{z} \in \mathsf{Im}(\mathsf{W}) \,, \tag{4}$$

where $\mu > 0$ and $\eta \geq 0$. S-REC implies that the pairwise distances between vectors in the range of the generative model must be well preserved in the measurement space. It thus broadens the traditional notions of REC and the *restricted isometry property (RIP)* in CS beyond sparse vectors [36].

PnP and RED. PnP [10, 11] refers to a family of iterative algorithms that are based on replacing the proximal operator $\mathsf{prox}_{\gamma h}$ of the regularizer h within a proximal algorithm [37] by a more general denoiser $\mathsf{D} : \mathbb{R}^n \to \mathsf{Im}(\mathsf{D})$, such as BM3D [38] or DnCNN [39]. For example, the widely used *proximal gradient method (PGM)* [40–43] can be implemented as a PnP algorithm as

$$\boldsymbol{x}^k = \mathsf{T}(\boldsymbol{x}^{k-1}) \quad \text{with} \quad \mathsf{T} \coloneqq \mathsf{D}(\mathsf{I} - \gamma\nabla g) \,, \tag{5}$$

where g is the data-fidelity term in (2), I denotes the identity mapping, and $\gamma > 0$ is the step size. Remarkably, this heuristic of using denoisers not associated with any h within a proximal algorithm exhibited great empirical success [13–18] and spurred a great deal of theoretical work on PnP [19–27]. In particular, it has been recently shown in [23] that, when the residual of D is Lipschitz continuous, PnP-PGM converges to a point in the fixed-point set of the operator T that we denote $\mathsf{Fix}(\mathsf{T})$.

RED [12] is a related method, inspired by PnP, for integrating denoisers as priors for inverse problems. For example, the *steepest descent* variant of RED (SD-RED) [12] can be summarized as

$$\boldsymbol{x}^k = \boldsymbol{x}^{k-1} - \gamma \mathsf{G}(\boldsymbol{x}^{k-1}) \quad \text{with} \quad \mathsf{G} \coloneqq \nabla g + \tau(\mathsf{I} - \mathsf{D}) \,, \tag{6}$$

where $\gamma > 0$ is the step size and $\tau > 0$ is the regularization parameter. For a locally homogeneous D that has a strongly passive and symmetric Jacobian, the solution of RED solves (2) with $h(\boldsymbol{x}) = (\tau/2)\boldsymbol{x}^{\mathsf{T}}(\boldsymbol{x} - \mathsf{D}(\boldsymbol{x}))$ [12, 22]. Subsequent work has resulted in a number of extensions of RED [24, 28, 44, 45]. For example, it has been shown in [24] that, when D is a nonexpansive operator, SD-RED converges to a point in the zero set of operator G that we denote as $\mathsf{Zer}(\mathsf{G})$.

Other related work. While not directly related to our main theoretical contributions, it is worth briefly mentioning other important related families of algorithms that also use deep neural nets for regularizing ill-posed imaging inverse problems (see recent reviews of the area [46–49]). This work is most related to methods that rely on pre-trained priors that are integrated within iterative algorithms, such as a class of algorithms in compressive sensing known as *approximate message passing (AMP)* [50–53]. Another related family of algorithms are those based on the idea of *deep unrolling* (for an overview see Section IV-A in [49]). Inspired by LISTA [54], the unrolling algorithms interpret iterations of a regularized inversion as layers of a CNN and train it end-to-end in a supervised fashion [55–59]. Deep image prior [60] and deep decoder [61] also use neural networks as prior for images; instead of using a pre-trained generative network, they learn the parameters of the network while solving the inverse problem using the available measurements.

3 Recovery Analysis for PnP and RED

We present two sets of theoretical results for PnP-PGM (5) using the measurement model (1) and the least-squares data-fidelity term (2). We first establish recovery bounds for PnP under a set of sufficient conditions, and then address the relationship between the solutions of PnP and RED. The proofs of all the theorems will be provided in the supplement. We start by discussing two assumptions that serve as sufficient conditions for our analysis of PnP.

Assumption 1. *The residual* $\mathsf{R} \coloneqq \mathsf{I} - \mathsf{D}$ *of the operator* D *is bounded by* δ *and Lipschitz continuous with constant* $\alpha > 0$*, which can be written as*

$$\|\mathsf{R}(\boldsymbol{x})\|_2 \leq \delta \quad \textit{and} \quad \|\mathsf{R}(\boldsymbol{x}) - \mathsf{R}(\boldsymbol{z})\|_2 \leq \alpha \|\boldsymbol{x} - \boldsymbol{z}\|_2, \quad \forall \boldsymbol{x}, \boldsymbol{z} \in \mathbb{R}^n \,. \tag{7}$$

The rationale for stating Assumption 1 in terms of the residual R is based on our interest in *residual* deep neural nets that take a noisy or an artifact-corrupted image at the input and produce the corresponding noise or artifacts at the output. The success of residual learning in the context of image restoration is well known [39]. Prior work has also shown that Lipschitz constrained residual networks yield excellent performance without sacrificing stable convergence [23, 24].

Related assumptions have been used in earlier convergence results for PnP [19, 23]. For example, one of the most-widely known PnP convergence results relies on the boundedness of D [19]. The Lipschitz continuity of the residual R has been used in the recent analysis of several PnP algorithms in [23]. Both of these assumptions are relatively easy to implement for deep priors. For example, the boundedness of R can be enforced by simply bounding each output pixel to be within $[0, \nu]$ for images in $[0, \nu]^n \subset \mathbb{R}^n$ for some $\nu > 0$. The α-Lipschitz continuity of R can be enforced by using any of the recent techniques for training Lipschitz constrained deep neural nets [23, 62–64]. Fig. 1 presents an empirical evaluation of the Lipschitz continuity of R used in our simulations.

Assumption 2. *The measurement operator* $\boldsymbol{A} \in \mathbb{R}^{m \times n}$ *satisfies the set-restricted eigenvalue condition (S-REC) over* $\mathsf{Im}(\mathsf{D}) \subseteq \mathbb{R}^n$ *with* $\mu > 0$*, which can be written as*

$$\|\boldsymbol{A}\boldsymbol{x} - \boldsymbol{A}\boldsymbol{z}\|_2^2 \geq \mu \|\boldsymbol{x} - \boldsymbol{z}\|_2^2, \quad \forall \boldsymbol{x}, \boldsymbol{z} \in \mathsf{Im}(\mathsf{D}) \,. \tag{8}$$

S-REC in Assumption 2 was adopted from the corresponding assumption for CSGM stated in (4), which establishes a natural conceptual link between those two classes of methods. The main limitation of Assumption 2, which is also present in the traditional RIP/REC assumptions for compressive sensing, lies in the difficulty of verifying it for a given measurement operator $\boldsymbol{A}$. There has been significant activity in investigating the validity of related conditions for randomized matrices for different classes of signals [4, 65–67], including for those synthesized by generative models [5, 6, 9, 35].

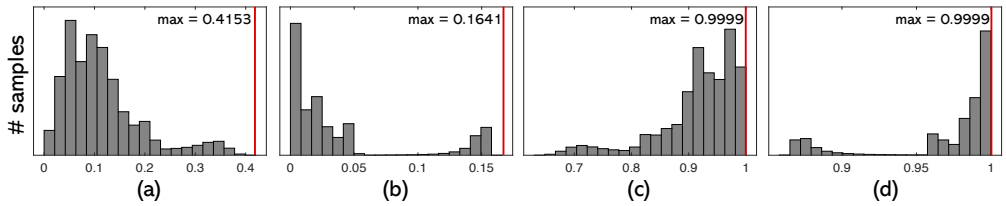


Figure 1: *Empirical evaluation of the Lipschitz continuity of* R *and* D *used in our simulations and stated in Assumptions 1 and 3. As described in the main text, we trained two types of Lipschitz constrained networks, where the first simply denoises AWGN and the second removes artifacts specific to the PnP iterations. (a) and (b) show the histograms of* $\|\mathsf{R}(\boldsymbol{x}) - \mathsf{R}(\boldsymbol{z})\|_2 / \|\boldsymbol{x} - \boldsymbol{z}\|_2$ *for the denoiser and the artifact-removal operator, respectively. (c) and (d) show the histograms of* $\|\mathsf{D}(\boldsymbol{x}) - \mathsf{D}(\boldsymbol{z})\|_2 / \|\boldsymbol{x} - \boldsymbol{z}\|_2$ *for the same two operators. Note the empirical nonexpansiveness of* D *despite the fact that Lipschitz continuity was only imposed on the residual* R *during training.*

Despite this limitation, Assumption 2 is still conceptually useful as it allows us to relax the strong convexity assumption used in the convergence analysis in [23] by stating that it is sufficient for the strong convexity to hold *only* over the image set $\mathsf{Im}(\mathsf{D})$ rather than over the whole $\mathbb{R}^n$. This suggests a new research direction for PnP on designing deep priors with range spaces restricted to satisfy S-REC for some $\boldsymbol{A}$. In the supplement, we present an empirical evaluation of μ for the measurement operators used in our experiments by sampling from $\mathsf{Im}(\mathsf{D})$.

Consider the set $\mathsf{Fix}(\mathsf{D}) \coloneqq \{\boldsymbol{x} \in \mathbb{R}^n : \boldsymbol{x} = \mathsf{D}(\boldsymbol{x})\}$ of the *fixed points* of D. Note that $\mathsf{Fix}(\mathsf{D})$ is equivalent to the set $\mathsf{Zer}(\mathsf{R}) \coloneqq \{\boldsymbol{x} \in \mathbb{R}^n : \mathsf{R}(\boldsymbol{x}) = \boldsymbol{0}\}$ of the *zeros* of the residual $\mathsf{R} = \mathsf{I} - \mathsf{D}$. Intuitively, $\mathsf{Zer}(\mathsf{R})$ consists of all images that produce no residuals, and therefore can be interpreted as the set of all *noise-free* images according to the network. Similarly, when R is trained to predict artifacts in an image, $\mathsf{Zer}(\mathsf{R})$ is the set of images that are *artifact-free* according to R. In the subsequent analysis, we use the notation $\mathsf{Zer}(\mathsf{R})$, but these results can be equivalently stated using $\mathsf{Fix}(\mathsf{D})$.

We first state the PnP recovery in the setting where there is no noise and $\boldsymbol{x}^* \in \mathsf{Zer}(\mathsf{R})$.

Theorem 1. *Run PnP-PGM for* $t \geq 1$ *iterations under Assumptions 1-2 for the problem* (1) *with no noise and* $\boldsymbol{x}^* \in \mathsf{Zer}(\mathsf{R})$. *Then, the sequence* $\boldsymbol{x}^t$ *generated by PnP-PGM satisfies*

$$\|\boldsymbol{x}^t - \boldsymbol{x}^*\|_2 \leq c\|\boldsymbol{x}^{t-1} - \boldsymbol{x}^*\|_2 \leq c^t\|\boldsymbol{x}^0 - \boldsymbol{x}^*\|_2 \,, \tag{9}$$

where $\boldsymbol{x}^0 \in \mathsf{Im}(\mathsf{D})$ *and* $c \coloneqq (1+\alpha)\max\{|1-\gamma\mu|, |1-\gamma\lambda|\}$ *with* $\lambda \coloneqq \lambda_{\mathsf{max}}(\boldsymbol{A}^{\mathsf{T}}\boldsymbol{A})$.

The proof of the theorem is available in the supplement. Theorem 1 extends the theoretical analysis of PnP in [23] by showing convergence to the true solution $\boldsymbol{x}^*$ of (1) instead of the fixed points $\mathsf{Fix}(\mathsf{T})$ of T in (5). Note that the condition $\boldsymbol{x}^0 \in \mathsf{Im}(\mathsf{D})$ can be easily enforced by simply passing any initial image through the operator D. As shown in [23], the coefficient c in Theorem 1 is less that one if

$$\frac{1}{\mu(1+1/\alpha)} < \gamma < \frac{2}{\lambda} - \frac{1}{\lambda(1+1/\alpha)} \,, \tag{10}$$

which implies that PnP-PGM recovers $\boldsymbol{x}^*$ that is the true solution for the inverse problem. Additionally, since all PnP algorithms have the same fixed points [20], our result implies that PnP can exactly recover $\boldsymbol{x}^*$, which extends the existing theory in the literature that only shows convergence to $\mathsf{Fix}(\mathsf{T})$.

We now present a more general result that relaxes the assumptions in Theorem 1.

Theorem 2. *Run PnP-PGM for* $t \geq 1$ *iterations under Assumptions 1-2 for the problem* (1) *with* $\boldsymbol{x}^* \in \mathbb{R}^n$ *and* $\boldsymbol{e} \in \mathbb{R}^m$. *Then, the sequence* $\boldsymbol{x}^t$ *generated by PnP-PGM satisfies*

$$\|\boldsymbol{x}^t - \boldsymbol{x}^*\|_2 \leq c\|\boldsymbol{x}^{t-1} - \boldsymbol{x}^*\|_2 + \varepsilon \leq c^t\|\boldsymbol{x}^0 - \boldsymbol{x}^*\|_2 + \frac{\varepsilon(1-c^t)}{(1-c)} \,, \tag{11}$$

where $\boldsymbol{x}^0 \in \mathsf{Im}(\mathsf{D})$ *and*

$$\varepsilon \coloneqq (1+c)\left[\left(1+2\sqrt{\lambda/\mu}\right)\|\boldsymbol{x}^* - \mathsf{proj}_{\mathsf{Zer}(\mathsf{R})}(\boldsymbol{x}^*)\|_2 + 2/\sqrt{\mu}\|\boldsymbol{e}\|_2 + \delta(1+1/\alpha)\right] \tag{12}$$

and $c \coloneqq (1+\alpha)\max\{|1-\gamma\mu|, |1-\gamma\lambda|\}$ *with* $\lambda \coloneqq \lambda_{\mathsf{max}}(\boldsymbol{A}^{\mathsf{T}}\boldsymbol{A})$.

Theorem 2 extends Theorem 1 by allowing $\boldsymbol{x}^*$ to be outside of $\mathsf{Zer}(\mathsf{R})$ and extends the analysis in [5] by considering operators D that do not necessarily project onto the range of a generative model. In the error bound ε, the first two terms are the distance of $\boldsymbol{x}^*$ to $\mathsf{Zer}(\mathsf{R})$ and the magnitude of the error $\boldsymbol{e}$, and have direct analogs in standard compressed sensing. The third term is the consequence of the possibility for the solution of PnP not being in the zero-set of R and one can show that when $\mathsf{Zer}(\mathsf{R}) \cap \mathsf{Zer}(\nabla g) \neq \varnothing$, then the third term disappears. As reported in the supplement, we empirically verified that the distance of the PnP solution to $\mathsf{Zer}(\mathsf{R})$ is small for both the denoiser and the artifact-removal operators used in our experiments.

Our final result explicitly relates the solutions of PnP and RED. In order to obtain the result, we need an additional assumption that the denoiser $\mathsf{D} = \mathsf{I} - \mathsf{R}$ is nonexpansive.

Assumption 3. *The denoiser* D *is nonexpansive*

$$\|\mathsf{D}(\boldsymbol{x}) - \mathsf{D}(\boldsymbol{z})\|_2 \leq \|\boldsymbol{x} - \boldsymbol{z}\|_2 \quad \forall \boldsymbol{x}, \boldsymbol{z} \in \mathbb{R}^n \ .$$

This is related but different from Assumption 1 that assumes the residual R is α-Lipschitz continuous.

The convergence of SD-RED in (6) to $\mathsf{Zer}(\mathsf{G})$ can be established for a nonexpansive operator D [24]. In principle, the nonexpansiveness of D can be enforced during the training of the prior in the same manner as that of the more general Lipschitz continuity. However, the prior in our numerical evaluations is trained to have a contractive residual R without any explicit constraints on D. As a reminder, the nonexpansiveness of R is only a necessary (but not sufficient) condition for the nonexpansiveness of D [68]. Despite this fact, our empirical evaluation of the Lipschitz constant of D in Fig. 1 indicates that D used in our experiments is nonexpansive.

Theorem 3. *Suppose that Assumptions 1-3 are satisfied and that* $\mathsf{Zer}(\nabla g) \cap \mathsf{Zer}(\mathsf{R}) \neq \varnothing$*, then PnP and RED have the same set of solutions:* $\mathsf{Fix}(\mathsf{T}) = \mathsf{Zer}(\mathsf{G})$.

As a reminder, the solutions of PnP correspond to the fixed-points of the operator T defined in (5), while those of RED to the zeroes of the operator G defined in (6). The assumption that $\mathsf{Zer}(\nabla g) \cap \mathsf{Zer}(\mathsf{R}) \neq \varnothing$ implies that there exist vectors that are noise/artifact free according to R and consistent with the measurements $\boldsymbol{y}$. While this assumption is not universally applicable to all the inverse problems and priors, it still provides a highly-intuitive sufficient condition for the PnP/RED equivalence. Although the relationship between PnP and RED has been explored in the prior work [22, 28], to the best of our knowledge, Theorem 3 is the first to prove explicit equivalence. If one additionally considers PnP-PGM with a step size that satisfies the condition in (10), then T is a contraction over $\mathsf{Im}(\mathsf{D})$, which implies that PnP-PGM converges linearly to its unique fixed point in $\mathsf{Im}(\mathsf{D})$. The direct corollary of our analysis is that, in the noiseless scenario $\boldsymbol{y} = \boldsymbol{A}\boldsymbol{x}^*$ with $\boldsymbol{x}^* \in \mathsf{Zer}(\mathsf{R})$, the image $\boldsymbol{x}^*$ is the unique fixed point of both PnP and RED over $\mathsf{Im}(\mathsf{D})$.

In summary, our theoretical analysis reveals that the fixed-point convergence of PnP/RED algorithms can be strengthened to provide recovery guarantees when S-REC from CSGM is satisfied. Since PnP/RED algorithms do not require nonconvex projections onto the range of a generative model, they enjoy computational benefits over methods that use generative models as priors. However, the literature on generative models is rich with theoretical bounds and recovery guarantees compared to that of PnP/RED. By showing that a similar analysis can be carried out for PnP/RED, we believe that our work suggests an exciting new direction of research for PnP/RED.

4 Numerical Evaluation

Before presenting our numerical results, it is important to note that PnP and RED are well-known methods and it is *not* our aim to claim any algorithmic novelty on them. However, comparing PnP/RED to state-of-the-art *compressed sensing (CS)* algorithms is of interest in the context of our theory. Our goal in this section is thus to both (a) empirically evaluate the recovery performance of PnP/RED and (b) compare their performances relative to widely-used CS algorithms.

We consider two scenarios: *(a) CS using random projections* and *(b) CS for magnetic resonance imaging (CS-MRI)*. In order to gain a deeper insights into performance under subsampling, we use an idealized noise-free setting; however, we expect similar relative performances under noise. For each scenario, we include comparisons with several well-established methods based on deep learning.

We consider two priors for PnP/RED: (i) an AWGN denoiser and (ii) an *artifact-removal (AR)* operator trained to remove artifacts specific to the PnP iterations. We implement both priors using the

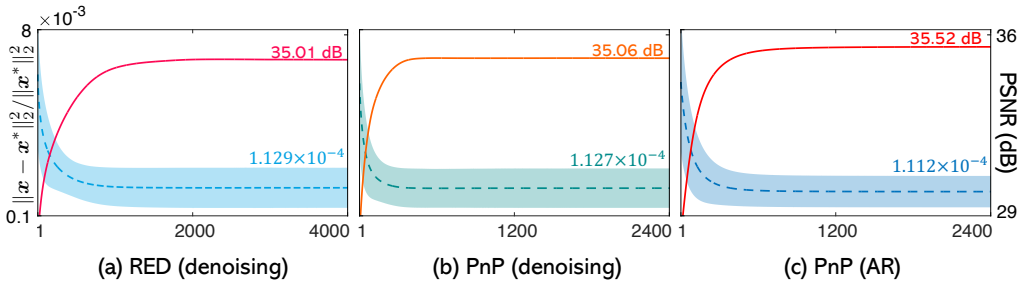


Figure 2: *Empirical evaluation of the convergence of PnP/RED to the true solution $\boldsymbol{x}^*$ using a denoiser and an artifact removal (AR) operator. Average normalized distance and PSNR relative to the true solution $\boldsymbol{x}^*$ are plotted with the shaded areas representing the range of values attained over all test images. Note the similar recovery performance of PnP and RED, as well as the improvement in performance due to a prior trained to remove artifacts specific to PnP iterations (rather than an AWGN denoiser).*

Table 1: Numerical evaluation of the CS recovery in terms of PSNR (dB) on BSD68 and Set11.

Method \ CS Ratio	BSD68				Set11			
	10%	**30%**	**40%**	**50%**	**10%**	**30%**	**40%**	**50%**
TV	24.56	28.61	30.27	31.98	24.47	30.21	32.29	34.27
SDA [71]	23.12	26.38	27.41	28.35	22.65	26.63	27.79	28.95
ReconNet [72]	24.15	27.53	29.08	29.86	24.28	28.74	30.58	31.50
ISTA-Net [58]	25.02	29.93	31.85	33.61	25.80	32.91	35.36	37.43
ISTA-Net$^+$ [58]	25.33	30.34	32.21	34.01	26.64	33.82	36.06	38.07
RED (denoising)	24.97	30.20	32.25	34.39	27.70	35.01	37.28	39.26
PnP (denoising)	25.06	30.31	32.29	34.35	27.76	35.06	37.30	39.21
PnP (AR)	**26.46**	**31.33**	**33.18**	**34.92**	**28.98**	**35.53**	**37.34**	**39.29**

DnCNN architecture [39], with its batch normalization layers removed for controlling the Lipschitz constant of the network via spectral normalization [63]. We train the denoiser as a nonexpansive residual network R that predicts the noise residual from a noisy input image. Thus, R satisfies the necessary condition for the nonexpansiveness of D. Similar to [69], we train the AR prior by including it into a *deep unfolding* architecture that performs PnP iterations. When equipped with spectral normalization [63], the residual R of the AR operator still satisfies Lipschitz continuity assumptions and achieves superior performance compared to the denoiser (as corroborated by our results). Our implementation also relies on the scaling strategy from [70] for controlling the influence of D relative to g. The reconstruction quality is quantified using the peak signal-to-noise ratio (PSNR) in dB.

4.1 Reconstruction of Natural Images from Random Projections

We adopt a simulation setup widely-used in the CS literature, in which non-overlapping 33×33 patches of an image are measured using the same $m \times n$ random Gaussian matrix $\boldsymbol{A}$, whose rows have been orthogonalized [58, 72]. The patches are vectorized to $n = 1089$-length vectors $\boldsymbol{x}^*$. The training data for the denoiser is generated by adding AWGN to the images from the BSD500 [73] and DIV2K datasets [74]. We pre-train several deep models as denoisers for $\sigma \in [1, 15]$, using σ intervals of 0.5, and use the denoiser achieving the best PSNR value in each experiment. We use the same set of 91 images as in [72] to train the AR operators that are implemented on individual image patches at a time for the CS ratios (m/n) of $\{10\%, 30\%, 40\%, 50\%\}$. In order to overcome the block-artifacts in the recovered images, we implement PnP and RED regularizers over the entire image while still using the per-patch measurement model for ∇g.

Our first numerical study in Fig. 1 evaluates the Lipschitz continuity of our pre-trained denoisers and the AR operators by following the procedure in [23]. We use the residual R and its corresponding operator $\mathsf{D} = \mathsf{I} - \mathsf{R}$ and plot the histograms of $\alpha_1 = \|\mathsf{R}(\boldsymbol{x}) - \mathsf{R}(\boldsymbol{z})\|_2/\|\boldsymbol{x} - \boldsymbol{z}\|_2$ and $\alpha_2 = \|\mathsf{D}(\boldsymbol{x}) - \mathsf{D}(\boldsymbol{z})\|_2/\|\boldsymbol{x} - \boldsymbol{z}\|_2$ over 1160 AWGN corrupted image pairs extracted from BSD68. The

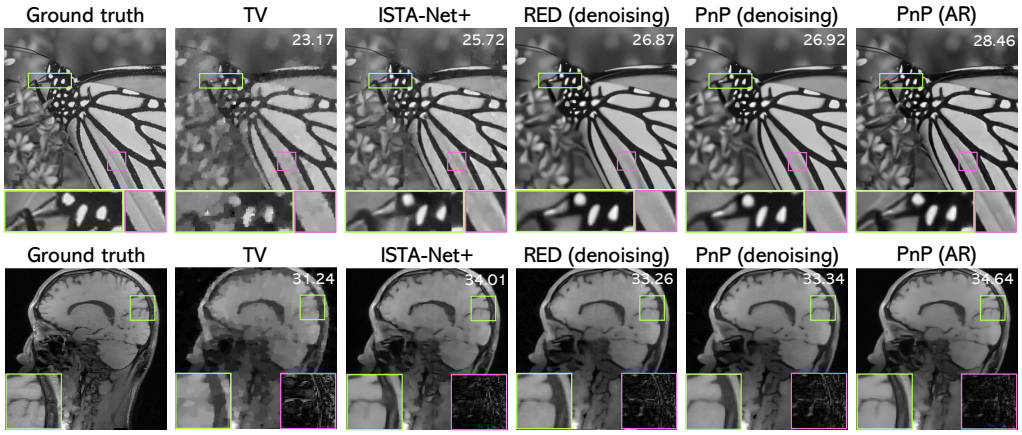


Figure 3: *Visual evaluation of various compressive sensing algorithms at* 10% *sampling on two imaging problems: (top) reconstruction of* Butterfly *from Set11; (bottom) reconstruction of a brain MR image from its radial Fourier measurements. The pink box in the bottom image provides the error residual that was amplified by* $10\times$ *for better visualization. Note the similar performance of PnP and RED, as well as the competitiveness of both relative to other methods. Additionally, note the improvement due to the usage of an AR prior instead of an AWGN denoiser within PnP.*

Table 2: Average PSNR values for various CS-MRI methods on test images from [58].

Method \ CS Ratio	10%	20%	30%	40%	50%
TV	31.36	35.62	38.41	40.43	42.20
ADMM-Net [57]	34.19	37.17	39.84	41.56	43.00
ISTA-Net$^+$ [58]	34.65	38.70	40.97	42.65	44.12
RED (denoising)	34.37	38.63	40.94	42.62	44.21
PnP (denoising)	34.56	38.74	41.06	42.73	44.24
PnP (AR)	**35.21**	**39.05**	**41.28**	**42.96**	**44.47**

maximum value of each histogram is indicated by a vertical bar, providing an empirical bound on the Lipschitz constants. Fig. 1 confirms empirically that both R and D are contractive operators.

Theorem 2 establishes that the sequence of iterates $\boldsymbol{x}^t$ generated by PnP-PGM converges to the true solutions $\boldsymbol{x}^*$ up to an error term. Fig. 2 illustrates the convergence behavior of PnP/RED in terms of $\|\boldsymbol{x}^t - \boldsymbol{x}^*\|_2^2 / \|\boldsymbol{x}^*\|_2^2$ and peak signal-to-noise ratio (PSNR) for CS with subsampling ratio of 30% on Set11. The shaded areas represent the range of values attained across all test images. The results in Fig. 2 are consistent with our general observation that the PnP/RED algorithms converge in all our experiments for both types of priors and achieve excellent recovery performance.

We also report the average PSNR values obtained by five baseline CS algorithms, namely TV [31], SDA [71], ReconNet [72], ISTA-Net [58] and ISTA-Net+ [58]. TV is an iterative methods that does not require training, while the other four are all deep learning-based methods that have publicly available implementations. The numerical results on Set11 and BSD68 with respect to four measurement rates are summarized in Table 1. We observe that the performances of PnP and RED are nearly identical to one another. The result also highlights that PnP using the AR prior provides the best performance[1] compared to all the other methods, outperforming PnP using the AWGN denoiser by at least 0.57 dB on BSD68. Fig. 3 (top) shows visual examples for an image from Set11. Note that both PnP and RED yield similar visual recovery performance. The enlarged regions in the image suggest that PnP (AR) better recovers the fine details and sharper edges compared to other methods.

[1]We did not use RED with the AR prior in our experiments since it is expected to closely match PnP.

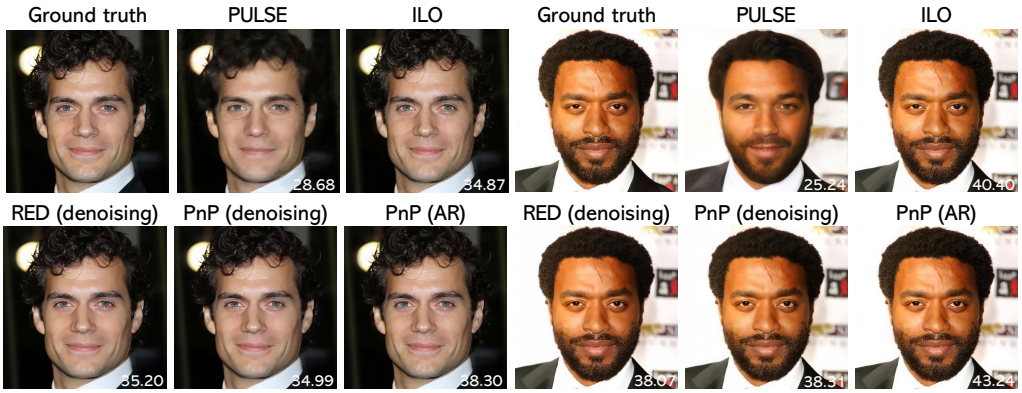


Figure 4: *Visual evaluation of PnP/RED and two methods using generative models as priors on the CelebA HQ [77] dataset at* 10% *CS sampling. Note the visual and quantitative similarity of PnP and RED when both are using AWGN denoisers. PnP using an artifact-removal (AR) prior visually matches the performance of ILO based on StyleGAN2, which highlights the benefit of using pre-trained AR operators within PnP. Best viewed by zooming in the display.*

Table 3: Average PSNR (dB) values for several algorithms on test images from CelebA HQ.

Method \ CS Ratio	10%	20%	30%	40%	50%
TV	32.13	35.24	37.41	39.35	41.29
PULSE [34]	27.45	29.98	33.06	34.25	34.77
ILO [35]	36.15	40.98	43.46	47.89	48.21
RED (denoising)	35.46	41.59	45.65	48.13	52.17
PnP (denoising)	35.61	41.51	45.71	48.05	52.24
PnP (AR)	**39.19**	**44.20**	**48.66**	**51.32**	**53.89**

4.2 Image reconstruction in Compressed Sensing MRI

MRI is a widely-used medical imaging technology that has known limitations due to the slow speed of data acquisition. CS-MRI [75, 76] seeks to recover an image $\boldsymbol{x}^*$ from its sparsely-sampled Fourier measurements. We simulate a single-coil CS-MRI using radial Fourier sampling. The measurement operator $\boldsymbol{A}$ is thus $\boldsymbol{A} = \boldsymbol{P}\boldsymbol{F}$, where $\boldsymbol{P}$ is the diagonal sampling matrix and $\boldsymbol{F}$ is the Fourier transform.

The priors for PnP/RED were trained using the brain dataset from [58], where the test set contains 50 slices of 256×256 images (i.e., $n = 65536$). We train seven variants of DnCNN, each using a separate noise level from $\sigma \in \{1, 1.5, 2, 2.5, 3, 4, 5\}$. Similarly, we separately train the AR operators for different CS ratios (m/n), initializing the weights of the models from the pre-trained denoiser with $\sigma = 2$. For these sets of experiments, we also equipped PnP/RED with *Nesterov acceleration* [79] for faster convergence. We compare PnP/RED against publicly available implementations of several well-known methods, including TV [31], ADMM-Net [57], and ISTA-Net$^+$ [58]. The last two are deep unrolling methods that train both image transforms and shrinkage functions within the algorithm.

Table 2 reports the results for five CS ratios. The visual comparison can be found in Fig. 3 (bottom). It can been seen that PnP/RED with an AWGN denoiser matches the performance of ISTA-Net$^+$ and outperforms ADMM-Net at higher sampling ratios, while PnP with an AR prior improves over PnP/RED with an AWGN denoiser [78]. Note also the similarity of PnP and RED performances.

4.3 Comparison with generative models on human faces

We numerically evaluated the recovery performance of PnP/RED in CS against two recent algorithms using generative models: PULSE [34] and ILO [35]. Similar to the measurement matrix used for grayscale images, we use orthogonalized random Gaussian matrices for sampling image blocks of size $33 \times 33 \times 3$. The test images correspond to 15 images randomly selected from CelebA HQ [77]

dataset, each of size 1024×1024 pixels. We use the DIV2K [74] and 200 high quality face images from FFHQ dataset [32] to train the PnP/RED denoisers for color image denoising at six noise levels corresponding to $\sigma \in \{1, 2, 3, 4, 7, 10\}$. We use the same training set to train the AR operators for CS ratios of $[10\%, 50\%]$, using the ratio intervals of 10%. Similar to CS-MRI, we equipped PnP/RED with Nesterov acceleration. The PSNR comparison between different methods is presented in Table 3. It can be seen that ILO outperforms PULSE in terms of PSNR, which is consistent with the results in [35]. Note also how PnP/RED match or sometimes quantitively outperform ILO at high CS ratios, with PnP (AR) leading to significantly better results compared to PnP (denoising). Fig. 4 provides visual reconstruction examples. Note the ILO images are sharper compared to PnP/RED with denoisers because ILO uses a state-of-the-art generative model specifically trained on face images. However, PnP (AR) achieves better PSNR and a similar visual quality as ILO.

5 Conclusion and Future Work

The main goal of this work is to address the theoretical gap between two-widely used classes of methods for solving inverse problems, namely PnP/RED and CSGM [5]. Motivated by the theoretical analysis of CSGM, we used S-REC to establish recovery guarantees for PnP/RED. Our theoretical results provide a new type of convergence for PnP-PGM that goes beyond a simple fixed-point convergence by showing convergence relative to the true solution. Additionally, we show the full equivalence of PnP and RED under some explicit conditions on the inverse problem. While the focus of this work is mainly theoretical, we presented several numerical evaluations that can provide additional insights into PnP/RED and their performance relative to standard methods used in compressed sensing. Empirically, we observed the similarity of PnP/RED in image reconstruction from subsampled random projections and Fourier transform. We also provided additional evidence on the suboptimality of AWGN denoisers compared to artifact-removal operators that take into account the actual artifacts within PnP iterates.

The work presented in this paper has a certain number of limitations and possible directions for improvement. The main limitation of our theoretical analysis is in the difficulty of theoretically verifying S-REC for a given measurement operator. This limitation goes beyond our work and is at the core of compressive sensing research. One can also consider the Lipschitz assumptions on R/D as a limitation, since those can have a negative impact on the recovery. However, our results suggest that even with Lipschitz constrained priors, PnP/RED are competitive with widely-known CS algorithms. While PnP/RED can be implemented using non-Lipschitz-constrained priors, we expect that this will reduce their stability and ultimately hurt their recovery performances. A relatively minor limitation of our simulations is that they were performed without AWGN. One can easily re-run the algorithms by including AWGN and we expect that the relative performances will be preserved for a reasonable amount of noise. We hope that this work will inspire further theoretical and algorithmic research on PnP/RED that will lead to extensions and improvements to our results.

6 Broader impact

This work is expected to impact the area of imaging inverse problems with potential applications to computational microscopy, computerized tomography, medical imaging, and image restoration. There is a growing need in imaging to deal with noisy and incomplete measurements by integrating multiple information sources, including physical information describing the imaging instrument and learned information characterizing the statistical distribution of the desired image. The ability to accurately solve inverse problems has the potential to enable new technological advances for imaging. These advances might lead to new imaging tools for diagnosing health conditions, understanding biological processes, or inferring properties of complex materials. Traditionally, imaging relied on linear models and fixed transforms (filtered back projection, wavelet transform) that are relatively straightforward to understand. Learning based methods, including PnP and RED, have the potential to enable new technological capabilities; yet, they also come with a downside of being much more complex. Their usage might thus lead to unexpected outcomes and surprising results when used by non-experts. While we aim to use our method to enable positive contributions to humanity, one can also imagine nonethical usage of imaging technology. This work focuses on understanding theoretical properties of imaging algorithms using learned priors, but it might be adopted within broader data science, which might lead to broader impacts that we have not anticipated.

Acknowledgments and Disclosure of Funding

Research presented in this article was supported by NSF awards CCF-1813910, CCF-2043134, and CCF-2046293 and by the Laboratory Directed Research and Development program of Los Alamos National Laboratory under project number 20200061DR.

References

[1] 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.

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

[3] P. T. Bickel, R. Ya'acov, and T. B. Alexandre, "Simultaneous analysis of lasso and Dantzig selector," *Ann. Statist.*, vol. 37, no. 4, pp. 1705–1732, 2009.

[4] M. J. Wainwright, *High-Dimensional Statistics: A Non-Asymptotic Viewpoint*, Cambridge Series in Statistical and Probabilistic Mathematics. Cambridge University Press, 2019.

[5] A. Bora, A. Jalal, E. Price, and A. G. Dimakis, "Compressed sensing using generative priors," in *Proc. 34th Int. Conf. Machine Learning (ICML)*, Sydney, Australia, Aug. 2017, pp. 537–546.

[6] V. Shah and C. Hegde, "Solving linear inverse problems using GAN priors: An algorithm with provable guarantees," in *Proc. IEEE Int. Conf. Acoustics, Speech and Signal Process.*, Calgary, AB, Canada, Apr. 2018, pp. 4609–4613.

[7] C. Hegde, "Algorithmic aspects of inverse problems using generative models," in *Proc. Allerton Conf. Communication, Control, and Computing*, Monticellu, IL, USA, Oct. 2018, pp. 166–172.

[8] F. Latorre, A. Eftekhari, and V. Cevher, "Fast and provable ADMM for learning with generative priors," in *Advances in Neural Information Processing Systems 33*, Vancouver, BC, USA, December 8-14, 2019, pp. 12027–12039.

[9] R. Hyder, V. Shah, C. Hegde, and M. S. Asif, "Alternating phase projected gradient descent with generative priors for solving compressive phase retrieval," in *Proc. IEEE Int. Conf. Acoustics, Speech and Signal Process.*, Brighton, UK, May 2019, pp. 7705–7709.

[10] 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.

[11] 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.

[12] 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.

[13] 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, HI, USA, July 2017, pp. 2808–2817.

[14] C. A. Metzler, P. Schniter, A. Veeraraghavan, and R. G. Baraniuk, "prDeep: Robust phase retrieval with a flexible deep network," in *Proc. 35th Int. Conf. Machine Learning (ICML)*, Stockholm, Sweden, June 2018, pp. 3501–3510.

[15] W. Dong, P. Wang, W. Yin, G. Shi, F. Wu, and X. Lu, "Denoising prior driven deep neural network for image restoration," *IEEE Trans. Patt. Anal. and Machine Intell.*, vol. 41, no. 10, pp. 2305–2318, Oct. 2019.

[16] K. Zhang, W. Zuo, and L. Zhang, "Deep plug-and-play super-resolution for arbitrary blur kernels," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition (CVPR)*, Long Beach, CA, USA, June 16-20, 2019, pp. 1671–1681.

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

[18] K. Wei, A. Aviles-Rivero, J. Liang, Y. Fu, C.-B. Schönlieb, and H. Huang, "Tuning-free plug-and-play proximal algorithm for inverse imaging problems," in *Proc. 37th Int. Conf. Machine Learning (ICML)*, 2020.

[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] 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, Oct. 2017, pp. 1799–1808.

[21] 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.

[22] E. T. Reehorst and P. Schniter, "Regularization by denoising: Clarifications and new interpretations," *IEEE Trans. Comput. Imag.*, vol. 5, no. 1, pp. 52–67, Mar. 2019.

[23] E. K. Ryu, J. Liu, S. Wnag, X. Chen, Z. Wang, and W. Yin, "Plug-and-play methods provably converge with properly trained denoisers," in *Proc. 36th Int. Conf. Machine Learning (ICML)*, Long Beach, CA, USA, June 2019, pp. 5546–5557.

[24] Y. Sun, J. Liu, and U. S. Kamilov, "Block coordinate regularization by denoising," in *Proc. Advances in Neural Information Processing Systems 33*, Vancouver, BC, Canada, Dec. 2019, pp. 382–392.

[25] T. Tirer and R. Giryes, "Image restoration by iterative denoising and backward projections," *IEEE Trans. Image Process.*, vol. 28, no. 3, pp. 1220–1234, 2019.

[26] A. M. Teodoro, J. M. Bioucas-Dias, and M. A. T. Figueiredo, "A convergent image fusion algorithm using scene-adapted Gaussian-mixture-based denoising," *IEEE Trans. Image Process.*, vol. 28, no. 1, pp. 451–463, Jan. 2019.

[27] X. Xu, Y. Sun, J. Liu, and U. S. Kamilov, "Provable convergence of plug-and-play priors with MMSE denoisers," *IEEE Signal Process. Lett.*, vol. 27, pp. 1280–1284, 2020.

[28] R. Cohen, M. Elad, and P. Milanfar, "Regularization by denoising via fixed-point projection (RED-PRO)," 2020, arXiv:2008.00226.

[29] 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.

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

[31] 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.

[32] T. Karras, S. Laine, and T. Aila, "A style-based generator architecture for generative adversarial networks," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition (CVPR)*, Long Beach, CA, USA, June 2019, pp. 4396–4405.

[33] T. Karras, S. Laine, M. Aittala, J. Hellsten, J. Lehtinen, and T. Aila, "Analyzing and improving the image quality of StyleGAN," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition (CVPR)*, Seattle, WA, USA, June 2020, pp. 8107–8116.

[34] S. Menon, A. Damian, S. Hu, N. Ravi, and C. Rudin, "PULSE: Self-supervised photo upsampling via latent space exploration of generative models," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition (CVPR)*, Seattle, WA, USA, June 2020, pp. 2434–2442.

[35] G. Daras, J. Dean, A. Jalal, and A. G. Dimakis, "Intermediate layer optimizationfor inverse problems using deep generative models," 2021, arXiv:2102.07364.

[36] E. Candès and M. B. Wakin, "An introduction to compressive sensing," *IEEE Signal Process. Mag.*, vol. 25, no. 2, pp. 21–30, March 2008.

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

[38] 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.

[39] K. Zhang, W. Zuo, Y. Chen, D. Meng, and L. Zhang, "Beyond a Gaussian denoiser: Residual learning of deep CNN for image denoising," *IEEE Trans. Image Process.*, vol. 26, no. 7, pp. 3142–3155, July 2017.

[40] 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.

[41] 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, November 2004.

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

[43] 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.

[44] G. Mataev, M. Elad, and P. Milanfar, "DeepRED: Deep image prior powered by RED," in *Proc. IEEE Int. Conf. Comp. Vis. Workshops (ICCVW)*, Seoul, South Korea, Oct 27-Nov 2, 2019, pp. 1–10.

[45] J. Liu, Y. Sun, C. Eldeniz, W. Gan, H. An, and U. S. Kamilov, "RARE: Image reconstruction using deep priors learned without ground truth," *IEEE J. Sel. Topics Signal Process.*, vol. 14, no. 6, pp. 1088–1099, 2020.

[46] M. T. McCann, K. H. Jin, and M. Unser, "Convolutional neural networks for inverse problems in imaging: A review," *IEEE Signal Process. Mag.*, vol. 34, no. 6, pp. 85–95, 2017.

[47] A. Lucas, M. Iliadis, R. Molina, and A. K. Katsaggelos, "Using deep neural networks for inverse problems in imaging: Beyond analytical methods," *IEEE Signal Process. Mag.*, vol. 35, no. 1, pp. 20–36, Jan. 2018.

[48] F. Knoll, K. Hammernik, C. Zhang, S. Moeller, T. Pock, D. K. Sodickson, and M. Akcakaya, "Deep-learning methods for parallel magnetic resonance imaging reconstruction: A survey of the current approaches, trends, and issues," *IEEE Signal Process. Mag.*, vol. 37, no. 1, pp. 128–140, Jan. 2020.

[49] G. Ongie, A. Jalal, C. A. Metzler, R. G. Baraniuk, A. G. Dimakis, and R. Willett, "Deep learning techniques for inverse problems in imaging," *IEEE J. Sel. Areas Inf. Theory*, vol. 1, no. 1, pp. 39–56, May 2020.

[50] J. Tan, Y. Ma, and D. Baron, "Compressive imaging via approximate message passing with image denoising," *IEEE Trans. Signal Process.*, vol. 63, no. 8, pp. 2085–2092, Apr. 2015.

[51] 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.

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

[53] A. Fletcher, S. Rangan, S. Sarkar, and P. Schniter, "Plug-in estimation in high-dimensional linear inverse problems: A rigorous analysis," in *Proc. Advances in Neural Information Processing Systems 32*, Montréal, Canada, Dec 3-8, 2018, pp. 7451–7460.

[54] K. Gregor and Y. LeCun, "Learning fast approximation of sparse coding," in *Proc. 27th Int. Conf. Machine Learning (ICML)*, Haifa, Israel, June 21-24, 2010, pp. 399–406.

[55] U. Schmidt and S. Roth, "Shrinkage fields for effective image restoration," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition (CVPR)*, Columbus, OH, USA, June 23-28, 2014, pp. 2774–2781.

[56] Y. Chen, W. Yu, and T. Pock, "On learning optimized reaction diffuction processes for effective image restoration," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition (CVPR)*, Boston, MA, USA, June 8-10, 2015, pp. 5261–5269.

[57] Y. Yang, J. Sun, H. Li, and Z. Xu, "Deep ADMM-Net for compressive sensing MRI," in *Proc. Advances in Neural Information Processing Systems 29*, 2016, pp. 10–18.

[58] J. Zhang and B. Ghanem, "ISTA-Net: Interpretable optimization-inspired deep network for image compressive sensing," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition (CVPR)*, 2018, pp. 1828–1837.

[59] H. K. Aggarwal, M. P. Mani, and M. Jacob, "MoDL: Model-based deep learning architecture for inverse problems," *IEEE Trans. Med. Imag.*, vol. 38, no. 2, pp. 394–405, Feb. 2019.

[60] D. Ulyanov, A. Vedaldi, and V. Lempitsky, "Deep image prior," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition (CVPR)*, Salt Lake City, UT, USA, June 18-22, 2018, pp. 9446–9454.

[61] R. Heckel and P. Hand, "Deep decoder: Concise image representations from untrained non-convolutional networks," in *Int. Conf. on Learning Representations (ICLR)*, 2018.

[62] M. Terris, A. Repetti, J.-C. Pesquet, and Y. Wiaux, "Building firmly nonexpansive convolutional neural networks," in *Proc. IEEE Int. Conf. Acoustics, Speech and Signal Process.*, Barcelona, Spain, May 2020, pp. 8658–8662.

[63] T. Miyato, T. Kataoka, M. Koyama, and Y. Yoshida, "Spectral normalization for generative adversarial networks," in *Int. Conf. on Learning Representations (ICLR)*, Vancouver, Canada, Apr. 2018.

[64] M. Fazlyab, A. Robey, Hassani. H., M. Marari, and G. Pappas, "Efficient and accurate estimation of Lipschitz constants for deep neural networks," in *Proc. Advances in Neural Information Processing Systems 33*, Vancouver, BC, Canada, Dec. 2019, pp. 11427–11438.

[65] M. Elad, *Sparse and Redundant Representations*, Springer, 2010.

[66] S. Foucart and H. Rauhut, *A Methematical Introduction to Compressive Sensing*, Birkhauser, 2013.

[67] R. Vershynin, *High-Dimensional Probability: An Introduction with Applications in Data Science*, Cambridge Series in Statistical and Probabilistic Mathematics. Cambridge University Press, 2018.

[68] H. H. Bauschke and P. L. Combettes, *Convex Analysis and Monotone Operator Theory in Hilbert Spaces*, Springer, 2 edition, 2017.

[69] D. Gilton, G. Ongie, and R. Willett, "Deep equilibrium architectures for inverse problems in imaging," 2021, arXiv:2102.07944.

[70] X. Xu, J. Liu, Y. Sun, B. Wohlberg, and U.S. Kamilov, "Boosting the performance of plug-and-play priors via denoiser scaling," 2020, arXiv:2002.11546.

[71] A. Mousavi, A. B. Patel, and R. G. Baraniuk, "A deep learning approach to structured signal recovery," in *Proc. Allerton Conf. Communication, Control, and Computing*, Allerton Park, IL, USA, September 30-October 2, 2015, pp. 1336–1343.

[72] K. Kulkarni, S. Lohit, P. Turaga, R. Kerviche, and A. Ashok, "ReconNet: Non-iterative reconstruction of images from compressively sensed measurements," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition (CVPR)*, Las Vegas, NV, USA, June 2016, pp. 449–458.

[73] D. Martin, C. Fowlkes, D. Tal, and J. Malik, "A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics," in *Proc. IEEE Int. Conf. Comp. Vis. (ICCV)*, Vancouver, Canada, July 7-14, 2001, pp. 416–423.

[74] E. Agustsson and R. Timofte, "Ntire 2017 challenge on single image super-resolution: Dataset and study," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition (CVPR) Workshops*, July 2017, vol. 3, pp. 126–135.

[75] M. Lustig, D. L. Donoho, and J. M. Pauly, "Sparse MRI: The application of compressed sensing for rapid MR imaging," *Magn. Reson. Med.*, vol. 58, no. 6, pp. 1182–1195, December 2007.

[76] M. Lustig, D. L. Donoho, J. M. Santos, and J. M. Pauly, "Compressed sensing MRI," *IEEE Signal Process. Mag.*, vol. 25, no. 2, pp. 72–82, 2008.

[77] T. Karras, T. Aila, S. Laine, and J. Lehtinen, "Progressive growing of gans for improved quality, stability, and variation," in *Int. Conf. on Learning Representations (ICLR)*, 2018.

[78] N. Eslahi and A. Foi, "Anisotropic spatiotemporal regularization in compressive video recovery by adaptively modeling the residual errors as correlated noise," in *IEEE Image, Video, and Multidimensional Signal Processing Workshop*, Zagorochoria, Greece, June 2018, pp. 1–5.

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

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

[81] R. T. Rockafellar, *Convex Analysis*, Princeton Univ. Press, Princeton, NJ, 1970.

[82] S. Boyd and L. Vandenberghe, *Convex Optimization*, Cambridge Univ. Press, 2004.

[83] P. Jain and P. Kar, "Non-convex optimization for machine learning," *Foundations and Trends in Machine Learning*, vol. 10, no. 3-4, pp. 142–363, 2017.

[84] 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, Sept. 2019.

Supplementary Material

The mathematical analysis presented in this supplementary document builds on two distinct lines of work: (a) monotone operator theory [68, 80] and (b) compressive sensing using generative models (CSGM) [5]. In Section A, we build on past work to prove the convergence of PnP-PGM to the true solution of the inverse problem in the absence of noise. In Section B, we extend the result in Section A to $\boldsymbol{x}^* \in \mathbb{R}^n$ and $\boldsymbol{e} \in \mathbb{R}^m$ (i.e., when the signal can be arbitrary and measurements can have noise). In Section C, we show that PnP/RED can have the same set of solutions under some specific conditions. In Section D, we provide background material useful for our theoretical analysis. Finally, in Section E, we provide additional technical details on our implementations and simulations omitted from the main paper due to space.

The algorithmic details of PnP-PGM and SD-RED are summarized in Fig. 5. It is important to note that it is not our intent to claim any algorithmic novelty in PnP/RED, which are well-known methods. However, there is a strong interest in understanding the theoretical properties of PnP/RED in terms of both recovery and convergence. The main contribution of this work is the development of new theoretical insights into the recovery and convergence of PnP/RED. Finally, our code, including pre-trained denoisers and AR operators, is also included in the supplementary material.

We follow the same notation in the supplement as in the main manuscript. The measurement model corresponds to $\boldsymbol{y} = \boldsymbol{A}\boldsymbol{x}^* + \boldsymbol{e}$, where $\boldsymbol{x}^*$ is the true solution and $\boldsymbol{e}$ is the noise. The function $g(\boldsymbol{x}) = \frac{1}{2}\|\boldsymbol{y} - \boldsymbol{A}\boldsymbol{x}\|_2^2$ denotes the data-fidelity term. The operator D denotes the PnP/RED prior, which is implemented via its residual $\mathsf{R} \coloneqq \mathsf{I} - \mathsf{D}$. The operator $\mathsf{T} \coloneqq \mathsf{D}(\mathsf{I} - \gamma\nabla g)$ denotes the PnP update and $\mathsf{G} \coloneqq \nabla g + \tau\mathsf{R}$ denotes the term used to compute RED updates.

A Proof of Theorem 1

In this section, we prove the first of the main theoretical result in this work, namely the convergence of PnP-PGM to the true solution of the problem $\boldsymbol{y} = \boldsymbol{A}\boldsymbol{x}^*$ when $\boldsymbol{x}^* \in \mathsf{Zer}(\mathsf{R})$. Our analysis extends the existing convergence analysis of PnP-PGM from [23], which proved a linear convergence of the algorithm to $\mathsf{Fix}(\mathsf{T})$. Here we extend [23] by using the fact that $\boldsymbol{x}^* \in \mathsf{Zer}(\mathsf{R}) \cap \mathsf{Zer}(\nabla g)$ and relaxing the assumption of strong convexity in [23] to S-REC over $\mathsf{Im}(\mathsf{D})$.

Suppose all the assumptions for Theorem 1 are true and the step size $\gamma > 0$ is selected in a way that satisfies eq. (10) in the main paper. First note that we have assumed that $\boldsymbol{x}^0 \in \mathsf{Im}(\mathsf{D})$ and we have

$$\boldsymbol{x}^t = \mathsf{T}(\boldsymbol{x}^{t-1}) = \mathsf{D}(\boldsymbol{x}^{t-1} - \gamma\nabla g(\boldsymbol{x}^{t-1})) \in \mathsf{Im}(\mathsf{D}) \ ,$$

which implies that all the PnP-PGM iterates $\{\boldsymbol{x}^t\}$ are in $\mathsf{Im}(\mathsf{D})$.

Note also the following equivalences

$$\mathsf{Zer}(\nabla g) = \mathsf{Fix}(\mathsf{I} - \gamma\nabla g) = \{\boldsymbol{x} \in \mathbb{R}^n : \nabla g(\boldsymbol{x}) = \boldsymbol{0}\} = \{\boldsymbol{x} \in \mathbb{R}^n : \boldsymbol{A}\boldsymbol{x} = \boldsymbol{y}\} \tag{13a}$$

$$\mathsf{Zer}(\mathsf{R}) = \mathsf{Fix}(\mathsf{D}) = \{\boldsymbol{x} \in \mathbb{R}^n : \mathsf{R}(\boldsymbol{x}) = \boldsymbol{0}\} \ , \tag{13b}$$

where the first equality in (13a) is due to the following equivalence true for any $\boldsymbol{x} \in \mathbb{R}^n$ and $\gamma > 0$

$$\nabla g(\boldsymbol{x}) = \boldsymbol{0} \quad \Leftrightarrow \quad \boldsymbol{x} - \gamma\nabla g(\boldsymbol{x}) = \boldsymbol{x} \ .$$

From the assumption $\boldsymbol{y} = \boldsymbol{A}\boldsymbol{x}^*$ with $\boldsymbol{x}^* \in \mathsf{Zer}(\mathsf{R})$ and from (13), we have the following inclusions:

$$\boldsymbol{x}^* \in \mathsf{Zer}(\nabla g) \cap \mathsf{Zer}(\mathsf{R}) \subseteq \mathsf{Fix}(\mathsf{T}) \subseteq \mathsf{Im}(\mathsf{D}) \subseteq \mathbb{R}^n \ .$$

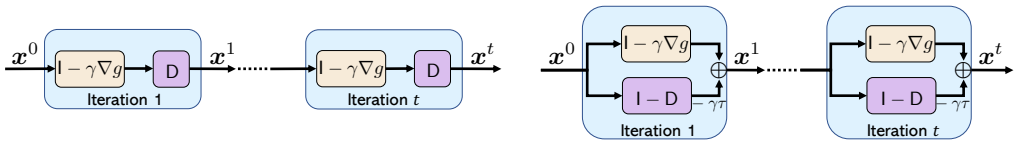
From Lemma 3 and Lemma 6, we conclude that for any $\boldsymbol{x}, \boldsymbol{z} \in \mathsf{Im}(\mathsf{D})$, we have

$$\|\mathsf{T}(\boldsymbol{x}) - \mathsf{T}(\boldsymbol{z})\|_2 \leq c\|\boldsymbol{x} - \boldsymbol{z}\|_2 \quad \text{with} \quad c = (1+\alpha)\max\{|1-\gamma\mu|, |1-\gamma\lambda|\} \ .$$

From T being a contraction over $\mathsf{Im}(\mathsf{D})$ and with Lemma 4, we can conclude that $\boldsymbol{x}^* \in \mathsf{Zer}(\nabla g) \cap \mathsf{Zer}(\mathsf{R})$ is the unique fixed point of PnP-PGM for any $\boldsymbol{x}^0 \in \mathsf{Im}(\mathsf{D})$. Thus, we have that

$$\|\boldsymbol{x}^t - \boldsymbol{x}^*\|_2 = \|\mathsf{T}(\boldsymbol{x}^{t-1}) - \mathsf{T}(\boldsymbol{x}^*)\|_2 \leq c\|\boldsymbol{x}^{t-1} - \boldsymbol{x}^*\|_2 \leq \cdots \leq c^t\|\boldsymbol{x}^0 - \boldsymbol{x}^*\|_2 \ ,$$

which establishes the desired result.



(a) PnP-PGM (b) SD-RED

Figure 5: *Algorithmic details of two optimization methods used in this work: (a) PnP-PGM and (b) SD-RED. Both algorithms are initialized with $\boldsymbol{x}^0$ and perform $t \geq 1$ iterations.*

B Proof of Theorem 2

In this section, we extend the analysis in Section A to the noisy measurement model $\boldsymbol{y} = \boldsymbol{A}\boldsymbol{x}^* + \boldsymbol{e}$ where $\boldsymbol{x}^* \in \mathbb{R}^n$ and $\boldsymbol{e} \in \mathbb{R}^m$. The following analysis builds on that of CSGM in [5] by showing that the proof techniques used for CSGM can be also used for analyzing PnP. Note also that one can improve the error term in the recovery under an additional assumption discussed in Section B.1.

Suppose all the assumptions for Theorem 2 are true and the step size $\gamma > 0$ is selected in a way that satisfies eq. (10) of the main manuscript. First note that Lemma 3 and Lemma 6 imply that for $\overline{\boldsymbol{x}} \in \mathsf{Fix}(\mathsf{T})$, we have that

$$\|\boldsymbol{x}^t - \overline{\boldsymbol{x}}\|_2 \leq c\|\boldsymbol{x}^{t-1} - \overline{\boldsymbol{x}}\|_2 \quad \text{with} \quad c = (1+\alpha)\max\{|1-\gamma\mu|, |1-\gamma\lambda|\} \in (0,1)\ . \tag{14}$$

Let $\widehat{\boldsymbol{x}} = \mathsf{proj}_{\mathsf{Zer}(\mathsf{R})}(\boldsymbol{x}^*)$, then we have that

$$\begin{aligned}
\|\overline{\boldsymbol{x}} - \widehat{\boldsymbol{x}}\| &\leq \frac{1}{\sqrt{\mu}}\left[\|\boldsymbol{y} - \boldsymbol{A}\overline{\boldsymbol{x}}\|_2 + \|\boldsymbol{y} - \boldsymbol{A}\widehat{\boldsymbol{x}}\|_2\right] \\
&\leq \frac{1}{\sqrt{\mu}}\left[\min_{\boldsymbol{x}\in\mathsf{Zer}(\mathsf{R})} \|\boldsymbol{y} - \boldsymbol{A}\boldsymbol{x}\|_2 + \sqrt{\mu}\delta(1+1/\alpha) + \|\boldsymbol{y} - \boldsymbol{A}\widehat{\boldsymbol{x}}\|_2\right] \\
&\leq \frac{2}{\sqrt{\mu}}\|\boldsymbol{y} - \boldsymbol{A}\widehat{\boldsymbol{x}}\|_2 + \delta(1+1/\alpha) \\
&\leq 2\sqrt{\frac{\lambda}{\mu}}\|\boldsymbol{x}^* - \widehat{\boldsymbol{x}}\|_2 + \frac{2}{\sqrt{\mu}}\|\boldsymbol{e}\|_2 + \delta(1+1/\alpha)\ ,
\end{aligned}$$

where the first inequality uses S-REC, the second one uses Lemma 1 in Section B.1, the third one combines two terms by picking the larger one, and the final one uses the measurement model and the triangular inequality. By using the inequality above, we can obtain the bound

$$\|\overline{\boldsymbol{x}} - \boldsymbol{x}^*\|_2 \leq \left[1 + 2\sqrt{\lambda/\mu}\right]\|\boldsymbol{x}^* - \mathsf{prox}_{\mathsf{Zer}(\mathsf{R})}(\boldsymbol{x}^*)\|_2 + [2/\sqrt{\mu}]\|\boldsymbol{e}\|_2 + \delta(1+1/\alpha) \coloneqq \varepsilon/(1+c)\ .$$

Note that the first two terms of $\varepsilon/(1+c)$ above are the distance of $\boldsymbol{x}^*$ to $\mathsf{Zer}(\mathsf{R})$ and the magnitude of the error $\boldsymbol{e}$, and have direct analogs in standard compressed sensing. The third term is the consequence of the possibility for the solution of PnP not being in $\mathsf{Zer}(\mathsf{R})$ and as discussed in Section B.1 when $\mathsf{Zer}(\mathsf{R}) \cap \mathsf{Zer}(\nabla g) \neq \varnothing$, then the third term disappears.

Then, from (14), we obtain

$$\begin{aligned}
\|\boldsymbol{x}^t - \boldsymbol{x}^*\|_2 &\leq \|\boldsymbol{x}^t - \overline{\boldsymbol{x}}\|_2 + \|\overline{\boldsymbol{x}} - \boldsymbol{x}^*\|_2 = \|\boldsymbol{x}^t - \overline{\boldsymbol{x}}\|_2 + \varepsilon/(c+1) \\
&\leq c\|\boldsymbol{x}^{t-1} - \overline{\boldsymbol{x}}\|_2 + \varepsilon/(c+1) = c\|\boldsymbol{x}^{t-1} - \boldsymbol{x}^*\|_2 + c\varepsilon/(c+1) + \varepsilon/(c+1) \\
&= c\|\boldsymbol{x}^{t-1} - \boldsymbol{x}^*\|_2 + \varepsilon \leq c^t\|\boldsymbol{x}^0 - \boldsymbol{x}^*\|_2 + \epsilon\sum_{k=0}^{t-1} c^k \\
&\leq c\|\boldsymbol{x}^0 - \boldsymbol{x}^*\|_2 + \varepsilon(1-c^t)/(1-c)\ ,
\end{aligned}$$

which establishes the desired result.

B.1 A Technical Lemma for the Proof of Theorem 2

The following lemma provides a bound used for Theorem 2. As discussed within the proof, if $\mathsf{Zer}(\mathsf{R}) \cap \mathsf{Zer}(\nabla g) \neq \varnothing$, the error term on the right of Lemma 1 can be removed by using Lemma 4. While this would lead to a tighter overall bound for Theorem 2, it would also reduce its generality. Fig. 7 empirically shows that the sequence $\|\mathsf{R}(\boldsymbol{x}^t)\|_2$ obtained by PnP-PGM in our simulations converges to a small value, suggesting that the solution obtained by the algorithm is near $\mathsf{Zer}(\mathsf{R})$.

Lemma 1. *Under the assumptions of Theorem 2 in the main manuscript, we have*

$$\|\boldsymbol{y} - \boldsymbol{A}\overline{\boldsymbol{x}}\|_2 \leq \min_{\boldsymbol{x} \in \mathsf{Zer}(\mathsf{R})} \|\boldsymbol{y} - \boldsymbol{A}\boldsymbol{x}\|_2 + \sqrt{\mu}\delta(1 + 1/\alpha) \; .$$

If in addition, we know that $\mathsf{Zer}(\mathsf{R}) \cap \mathsf{Zer}(\nabla g) \neq \varnothing$*, then*

$$\|\boldsymbol{y} - \boldsymbol{A}\overline{\boldsymbol{x}}\|_2 \leq \min_{\boldsymbol{x} \in \mathsf{Zer}(\mathsf{R})} \|\boldsymbol{y} - \boldsymbol{A}\boldsymbol{x}\|_2 \; .$$

Proof. First note that by re-expressing the fixed point equation of PnP-PGM, we obtain

$$\begin{aligned} &\overline{\boldsymbol{x}} = \mathsf{D}(\overline{\boldsymbol{x}} - \gamma \nabla g(\overline{\boldsymbol{x}})) \\ \Leftrightarrow \quad &\begin{cases} \overline{\boldsymbol{z}} = \overline{\boldsymbol{x}} - \gamma \nabla g(\overline{\boldsymbol{x}}) \\ \overline{\boldsymbol{x}} = \overline{\boldsymbol{z}} - (\overline{\boldsymbol{z}} - \mathsf{D}(\overline{\boldsymbol{z}})) = \overline{\boldsymbol{z}} - \mathsf{R}(\overline{\boldsymbol{z}}) \end{cases} \qquad \Rightarrow \quad \nabla g(\overline{\boldsymbol{x}}) + \frac{1}{\gamma}\mathsf{R}(\overline{\boldsymbol{z}}) = \boldsymbol{0} \; , \end{aligned}$$

where the final result is obtained by adding the two equalities on the left. Since g satisfies S-REC over $\mathsf{Im}(\mathsf{D})$, Lemma 5 in Section D.2 implies that for any $\boldsymbol{x} \in \mathsf{Im}(\mathsf{D})$ and $\overline{\boldsymbol{x}} \in \mathsf{Fix}(\mathsf{T})$

$$\begin{aligned} g(\boldsymbol{x}) &\geq g(\overline{\boldsymbol{x}}) + \nabla g(\overline{\boldsymbol{x}})^{\mathsf{T}}(\boldsymbol{x} - \overline{\boldsymbol{x}}) + \frac{\mu}{2}\|\boldsymbol{x} - \overline{\boldsymbol{x}}\|_2^2 \\ &= g(\overline{\boldsymbol{x}}) - (1/\gamma)\mathsf{R}(\overline{\boldsymbol{z}})^{\mathsf{T}}(\boldsymbol{x} - \overline{\boldsymbol{x}}) + \frac{\mu}{2}\|\boldsymbol{x} - \overline{\boldsymbol{x}}\|_2^2 \\ &\geq \min_{\boldsymbol{x} \in \mathsf{Im}(\mathsf{D})} \left\{ g(\overline{\boldsymbol{x}}) - (1/\gamma)\mathsf{R}(\overline{\boldsymbol{z}})^{\mathsf{T}}(\boldsymbol{x} - \overline{\boldsymbol{x}}) + \frac{\mu}{2}\|\boldsymbol{x} - \overline{\boldsymbol{x}}\|_2^2 \right\} \\ &\geq \min_{\boldsymbol{x} \in \mathbb{R}^n} \left\{ g(\overline{\boldsymbol{x}}) - (1/\gamma)\mathsf{R}(\overline{\boldsymbol{z}})^{\mathsf{T}}(\boldsymbol{x} - \overline{\boldsymbol{x}}) + \frac{\mu}{2}\|\boldsymbol{x} - \overline{\boldsymbol{x}}\|_2^2 \right\} \\ &\geq g(\overline{\boldsymbol{x}}) - \frac{1}{2\mu\gamma^2}\|\mathsf{R}(\overline{\boldsymbol{z}})\|_2^2 \; , \end{aligned}$$

where $\overline{\boldsymbol{z}} = \overline{\boldsymbol{x}} - \gamma \nabla g(\overline{\boldsymbol{x}})$. By rearranging the terms and minimizing over $\boldsymbol{x} \in \mathsf{Im}(\mathsf{D})$, we obtain

$$g(\overline{\boldsymbol{x}}) \leq \min_{\boldsymbol{x} \in \mathsf{Zer}(\mathsf{R})} g(\boldsymbol{x}) + \frac{1}{2\mu\gamma^2}\|\mathsf{R}(\overline{\boldsymbol{z}})\|_2^2 \leq \min_{\boldsymbol{x} \in \mathsf{Zer}(\mathsf{R})} g(\boldsymbol{x}) + \frac{\delta^2}{2\mu\gamma^2} \; , \tag{15}$$

where in the last inequality we used the boundedness of R.

By using the actual expression of g and the lower-bound on γ in eq. (10) in the main paper, we obtain

$$\begin{aligned} &1/\gamma < \mu\,(1 + 1/\alpha) \\ \Rightarrow \quad &\|\boldsymbol{y} - \boldsymbol{A}\overline{\boldsymbol{x}}\|_2 \leq \min_{\boldsymbol{x} \in \mathsf{Zer}(\mathsf{R})} \|\boldsymbol{y} - \boldsymbol{A}\boldsymbol{x}\|_2 + \delta/(\sqrt{\mu}\gamma) \leq \min_{\boldsymbol{x} \in \mathsf{Zer}(\mathsf{R})} \|\boldsymbol{y} - \boldsymbol{A}\boldsymbol{x}\|_2 + \delta\sqrt{\mu}(1 + 1/\alpha) \; . \end{aligned}$$

If we assume that $\mathsf{Zer}(\mathsf{R}) \cap \mathsf{Zer}(\nabla g) \neq \varnothing$, then from Lemma 4, we have $\overline{\boldsymbol{x}} \in \mathsf{Zer}(\mathsf{R}) \cap \mathsf{Zer}(\nabla g)$, which implies that $\overline{\boldsymbol{x}} = \overline{\boldsymbol{z}}$ and $\mathsf{R}(\overline{\boldsymbol{z}}) = \mathsf{R}(\overline{\boldsymbol{x}}) = \boldsymbol{0}$. In this case, we can eliminate the error term in (15)

$$g(\overline{\boldsymbol{x}}) \leq \min_{\boldsymbol{x} \in \mathsf{Zer}(\mathsf{R})} g(\boldsymbol{x}) \; .$$

□

C Proof of Theorem 3

The SD-RED algorithm in eq. (6) of the main manuscript seeks zeroes of the operator

$$\mathsf{G} = \nabla g + \tau \mathsf{R} \; .$$

It is clear that

$$\nabla g(\boldsymbol{z}) = \mathbf{0} \quad \text{and} \quad \mathsf{R}(\boldsymbol{z}) = \mathbf{0} \quad \Rightarrow \quad \mathsf{G}(\boldsymbol{z}) = \mathbf{0} \; ,$$

which corresponds to the inclusion $\mathsf{Zer}(\nabla g) \cap \mathsf{Zer}(\mathsf{R}) \subseteq \mathsf{Zer}(\mathsf{G})$.

We now prove the reverse inclusion under the assumptions of Theorem 3. Let $\boldsymbol{x} \in \mathsf{Zer}(\mathsf{G})$ and $\boldsymbol{z} \in \mathsf{Zer}(\nabla g) \cap \mathsf{Zer}(\mathsf{R})$. Since ∇g is λ-Lipschitz continuous with $\lambda = \lambda_{\mathsf{max}}(\boldsymbol{A}^{\mathsf{T}}\boldsymbol{A})$, Lemma 7 in Section D.2 implies that ∇g is also $(1/\lambda)$-cocoercive. Therefore, we have that

$$\nabla g(\boldsymbol{x})^{\mathsf{T}}(\boldsymbol{x}-\boldsymbol{z}) = (\nabla g(\boldsymbol{x}) - \nabla g(\boldsymbol{z}))^{\mathsf{T}}(\boldsymbol{x}-\boldsymbol{z}) \geq (1/\lambda)\|\nabla g(\boldsymbol{x}) - \nabla g(\boldsymbol{z})\|_2^2 = (1/\lambda)\|\nabla g(\boldsymbol{x})\|_2^2 \; .$$

On the other hand, since D is nonexpansive, $\mathsf{R} = \mathsf{I} - \mathsf{D}$ is $(1/2)$-cocoercive, which implies that

$$\mathsf{R}(\boldsymbol{x})^{\mathsf{T}}(\boldsymbol{x}-\boldsymbol{z}) = (\mathsf{R}(\boldsymbol{x}) - \mathsf{R}(\boldsymbol{z}))^{\mathsf{T}}(\boldsymbol{x}-\boldsymbol{z}) \geq (1/2)\|\mathsf{R}(\boldsymbol{x}) - \mathsf{R}(\boldsymbol{z})\|_2^2 = (1/2)\|\mathsf{R}(\boldsymbol{x})\|_2^2 \; .$$

By using the fact that $\mathsf{G}(\boldsymbol{x}) = \mathbf{0}$ and the two inequalities above, we obtain

$$0 = \mathsf{G}(\boldsymbol{x})^{\mathsf{T}}(\boldsymbol{x}-\boldsymbol{z}) = \nabla g(\boldsymbol{x})^{\mathsf{T}}(\boldsymbol{x}-\boldsymbol{z}) + \tau \mathsf{R}(\boldsymbol{x})^{\mathsf{T}}(\boldsymbol{x}-\boldsymbol{z}) \geq (1/\lambda)\|\nabla g(\boldsymbol{x})\|_2^2 + (1/2)\|\mathsf{R}(\boldsymbol{x})\|_2^2 \; , \quad (16)$$

which directly leads to the conclusion

$$\mathsf{G}(\boldsymbol{x}) = \mathbf{0} \quad \Rightarrow \quad \nabla g(\boldsymbol{x}) = \mathbf{0} \quad \text{and} \quad \mathsf{R}(\boldsymbol{x}) = \mathbf{0} \; .$$

Therefore, we have that $\mathsf{Zer}(\mathsf{G}) = \mathsf{Zer}(\nabla g) \cap \mathsf{Zer}(\mathsf{R})$.

Note also that from Lemma 3, we know that when $\mathsf{Zer}(\nabla g) \cap \mathsf{Zer}(\mathsf{R}) \neq \varnothing$, we have $\mathsf{Fix}(\mathsf{T}) = \mathsf{Zer}(\nabla g) \cap \mathsf{Zer}(\mathsf{R})$, which directly leads to our result

$$\mathsf{Zer}(\mathsf{G}) = \mathsf{Zer}(\nabla g) \cap \mathsf{Zer}(\mathsf{R}) = \mathsf{Fix}(\mathsf{T}) \; .$$

D Background Material

The results in this sections are well-known and can be found in different forms in standard textbooks [68, 79, 81, 82]. For completeness, we summarize the key results used in our analysis.

D.1 Properties of Monotone Operators

Definition 1. *An operator T is Lipschitz continuous with constant $\lambda > 0$ if*

$$\|\mathsf{T}(\boldsymbol{x}) - \mathsf{T}(\boldsymbol{z})\|_2 \leq \lambda \|\boldsymbol{x} - \boldsymbol{z}\|_2 \quad \forall \boldsymbol{x}, \boldsymbol{y} \in \mathbb{R}^n \; .$$

When $\lambda = 1$, we say that T is nonexpansive. When $\lambda < 1$, we say that T is a contraction.

Definition 2. *T is monotone if*

$$(\mathsf{T}(\boldsymbol{x}) - \mathsf{T}(\boldsymbol{z}))^{\mathsf{T}}(\boldsymbol{x} - \boldsymbol{z}) \geq 0 \quad \forall \boldsymbol{x}, \boldsymbol{y} \in \mathbb{R}^n \; .$$

We say that T is strongly monotone with parameter $\theta > 0$ if

$$(\mathsf{T}(\boldsymbol{x}) - \mathsf{T}(\boldsymbol{z}))^{\mathsf{T}}(\boldsymbol{x} - \boldsymbol{z}) \geq \theta \|\boldsymbol{x} - \boldsymbol{z}\|_2^2 \quad \forall \boldsymbol{x}, \boldsymbol{y} \in \mathbb{R}^n \; .$$

Definition 3. *T is cocoercive with constant $\beta > 0$ if*

$$(\mathsf{T}(\boldsymbol{x}) - \mathsf{T}(\boldsymbol{z}))^{\mathsf{T}}(\boldsymbol{x} - \boldsymbol{z}) \geq \beta \|\mathsf{T}(\boldsymbol{x}) - \mathsf{T}(\boldsymbol{z})\|_2^2 \quad \forall \boldsymbol{x}, \boldsymbol{z} \in \mathbb{R}^n \; .$$

When $\beta = 1$, we say that T is firmly nonexpansive.

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

The following lemma is derived from the definitions above.

Lemma 2. *Consider $\mathsf{R} = \mathsf{I} - \mathsf{D}$ where $\mathsf{D} : \mathbb{R}^n \to \mathbb{R}^n$. Then,*

$$\mathsf{D} \text{ is nonexpanisve} \quad \Leftrightarrow \quad \mathsf{R} \text{ is } (1/2)\text{-cocoercive} \; .$$

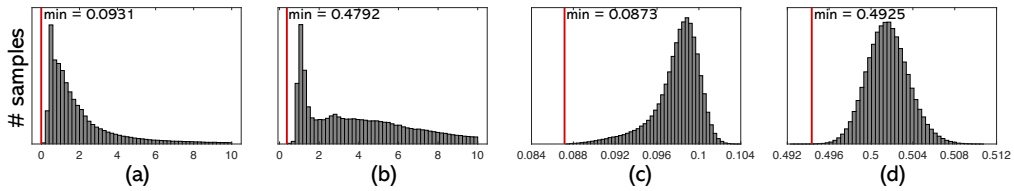


Figure 6: *Empirical evaluation of the S-REC constant $\mu > 0$ for the measurement operators $\boldsymbol{A}$ used in our simulations. We tested both the AWGN denoisers and the AR operators by randomly sampling from their image spaces $\mathsf{Im}(\mathsf{D})$. The $x-$axis is the value of $\|\boldsymbol{A}\boldsymbol{x} - \boldsymbol{A}\boldsymbol{y}\|_2^2/\|\boldsymbol{x} - \boldsymbol{y}\|_2^2$. (a) and (b) show the histograms for the radially sub-sampled MRI matrices at 10% and 50% sampling ratios, respectively. (c) and (d) show the histograms for the random Gaussian matrices for the same two sampling ratios. As expected, one can observe the increase in μ for the higher sampling ratio of 50%.*

Proof. First suppose that R is $1/2$ cocoercive. Let $\boldsymbol{h} := \boldsymbol{x} - \boldsymbol{z}$ for any $\boldsymbol{x}, \boldsymbol{z} \in \mathbb{R}^n$. We then have

$$\frac{1}{2}\|\mathsf{R}(\boldsymbol{x}) - \mathsf{R}(\boldsymbol{z})\|_2^2 \leq (\mathsf{R}(\boldsymbol{x}) - \mathsf{R}(\boldsymbol{z}))^{\mathsf{T}}\boldsymbol{h} = \|\boldsymbol{h}\|_2^2 - (\mathsf{D}(\boldsymbol{x}) - \mathsf{D}(\boldsymbol{z}))^{\mathsf{T}}\boldsymbol{h} \; .$$

We also have that

$$\frac{1}{2}\|\mathsf{R}(\boldsymbol{x}) - \mathsf{R}(\boldsymbol{z})\|_2^2 = \frac{1}{2}\|\boldsymbol{h}\|^2 - (\mathsf{D}(\boldsymbol{x}) - \mathsf{D}(\boldsymbol{z}))^{\mathsf{T}}\boldsymbol{h} + \frac{1}{2}\|\mathsf{D}(\boldsymbol{x}) - \mathsf{D}(\boldsymbol{z})\|_2^2 \; .$$

By combining these two and simplifying the expression

$$\|\mathsf{D}(\boldsymbol{x}) - \mathsf{D}(\boldsymbol{z})\|_2 \leq \|\boldsymbol{h}\|_2 \; .$$

The converse can be proved by following this logic in reverse. □

The following lemma relates the Lipschitz continuity of the residual $\mathsf{S} = \mathsf{I} - \mathsf{T}$ to that of T.

Lemma 3. *The operator $\mathsf{R} = \mathsf{I} - \mathsf{D}$ is α-Lipschitz continuous if and only if the operator $(1/(1+\alpha))\mathsf{D}$ is nonexpansive and $\alpha/(1+\alpha)$-averaged.*

Proof. See Lemma 9 in [23]. □

The following lemma considers fixed points of a composite operator.

Lemma 4. *Let $\mathsf{T} = \mathsf{D} \cdot \mathsf{S}$ with $\mathsf{Fix}(\mathsf{D}) \cap \mathsf{Fix}(\mathsf{S}) \neq \varnothing$ be a contraction with constant $\lambda \in (0, 1)$ over the set $\mathsf{Im}(\mathsf{D}) \subseteq \mathbb{R}^n$. Then, we have that $\mathsf{Fix}(\mathsf{T}) = \mathsf{Fix}(\mathsf{D}) \cap \mathsf{Fix}(\mathsf{S})$.*

Proof. We modify the proof of Proposition 4.49 from [68] to be consistent with our assumptions.

It is clear that $\mathsf{Fix}(\mathsf{D}) \cap \mathsf{Fix}(\mathsf{S}) \subseteq \mathsf{Fix}(\mathsf{T})$ and our goal is to show the reverse inclusion. Let $\boldsymbol{x} \in \mathsf{Fix}(\mathsf{T})$ and consider three cases.

- *Case* $\mathsf{S}(\boldsymbol{x}) \in \mathsf{Fix}(\mathsf{D})$: We have that
$$\mathsf{S}(\boldsymbol{x}) = \mathsf{D}(\mathsf{S}(\boldsymbol{x})) = \mathsf{T}(\boldsymbol{x}) = \boldsymbol{x} \in \mathsf{Fix}(\mathsf{D}) \cap \mathsf{Fix}(\mathsf{S}) \; .$$
- *Case* $\boldsymbol{x} \in \mathsf{Fix}(\mathsf{S})$: We have that
$$\mathsf{D}(\boldsymbol{x}) = \mathsf{D}(\mathsf{S}(\boldsymbol{x})) = \mathsf{T}(\boldsymbol{x}) = \boldsymbol{x} \in \mathsf{Fix}(\mathsf{D}) \cap \mathsf{Fix}(\mathsf{S}) \; .$$
- *Case* $\mathsf{S}(\boldsymbol{x}) \notin \mathsf{Fix}(\mathsf{D})$ *and* $\boldsymbol{x} \notin \mathsf{Fix}(\mathsf{S})$: Since $\mathsf{T} = \mathsf{D} \cdot \mathsf{S}$ is a contraction over $\mathsf{Im}(\mathsf{D})$
$$\|\boldsymbol{x} - \boldsymbol{z}\|_2 = \|\mathsf{T}(\boldsymbol{x}) - \mathsf{T}(\boldsymbol{z})\|_2 \leq \lambda\|\boldsymbol{x} - \boldsymbol{z}\|_2 \quad \forall \boldsymbol{z} \in \mathsf{Fix}(\mathsf{D}) \cap \mathsf{Fix}(\mathsf{S}) \; ,$$
which is impossible.

□

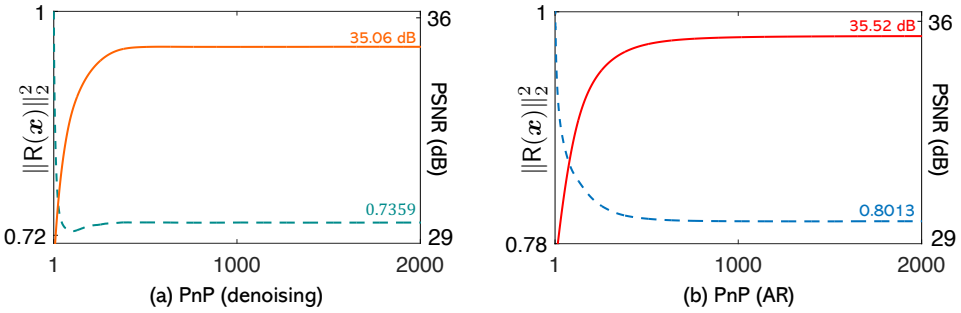


Figure 7: *Illustration of the convergence of PnP under nonexpensive denoisers and AR operators. Average normalized distance to* $\|\mathsf{R}(\boldsymbol{x})\|_2^2 = \|\boldsymbol{x} - \mathsf{D}(\boldsymbol{x})\|_2^2$ *and PSNR (dB) are plotted as dashed and solid lines, respectively, against the iteration number. This plot illustrates that PnP in our experiments converges to vectors close to* $\mathsf{Zer}(\mathsf{R})$*, which is consistent with the view that it regularizes inverse problems by obtaining solutions near the fixed-points of a denoiser/AR operator.*

D.2 Convexity, restricted strong convexity, and set-restricted eigenvalue condition

S-REC in the main manuscript can be generalized to the *restricted strong convexity (RSC)* assumption, which is widely-used in the nonconvex analysis of the gradient methods (see Section 3.2 in [83]).

Definition 5. *A continuously differentiable function* g *is said to satisfy restricted strong convexity (RSC) over* $\mathcal{X} \subseteq \mathbb{R}^n$ *with* $\mu > 0$ *if*

$$g(\boldsymbol{z}) \geq g(\boldsymbol{x}) + \nabla g(\boldsymbol{x})^{\mathsf{T}}(\boldsymbol{z} - \boldsymbol{x}) + \frac{\mu}{2}\|\boldsymbol{z} - \boldsymbol{x}\|_2^2 \quad \forall \boldsymbol{x}, \boldsymbol{z} \in \mathcal{X} \ .$$

In fact, for $g(\boldsymbol{x}) = \frac{1}{2}\|\boldsymbol{y} - \boldsymbol{A}\boldsymbol{x}\|_2^2$, S-REC is equivalent to RSC in Definition 5.

Lemma 5. *Let* $g(\boldsymbol{x}) = \frac{1}{2}\|\boldsymbol{y} - \boldsymbol{A}\boldsymbol{x}\|_2^2$ *and consider* $\mathcal{X} \subseteq \mathbb{R}^n$*. Then,*

$$g \text{ satisfies S-REC with } \mu \text{ over } \mathcal{X} \quad \Leftrightarrow \quad g \text{ satisfies } \mu\text{-RSC over } \mathcal{X} \ .$$

Proof. Suppose g is the least-squares function that satisfies S-REC with μ, then for any $\boldsymbol{x}, \boldsymbol{z} \in \mathcal{X}$

$$\begin{aligned} g(\boldsymbol{z}) &= g(\boldsymbol{x}) + \nabla g(\boldsymbol{x})^{\mathsf{T}}(\boldsymbol{z} - \boldsymbol{x}) + \frac{1}{2}(\boldsymbol{z} - \boldsymbol{x})\boldsymbol{A}^{\mathsf{T}}\boldsymbol{A}(\boldsymbol{z} - \boldsymbol{x}) \\ &= g(\boldsymbol{x}) + \nabla g(\boldsymbol{x})^{\mathsf{T}}(\boldsymbol{z} - \boldsymbol{x}) + \frac{1}{2}\|\boldsymbol{A}(\boldsymbol{z} - \boldsymbol{x})\|_2^2 \\ &\geq g(\boldsymbol{x}) + \nabla g(\boldsymbol{x})^{\mathsf{T}}(\boldsymbol{z} - \boldsymbol{x}) + \frac{\mu}{2}\|\boldsymbol{z} - \boldsymbol{x}\|_2^2 \ , \end{aligned}$$

which implies that g satisfies μ-RSC. To show S-REC using μ-RSC, follow the logic in reverse. □

One can use the previous and the following lemma to show that the gradient step of PnP-PGM can be a contraction for any vector in $\mathsf{Im}(\mathsf{D})$ for a properly chosen step size.

Lemma 6. *Assume* g *satisfies* μ*-RSC over* $\mathcal{X} \subseteq \mathbb{R}^n$ *and* ∇g *is* λ*-Lipschitz continuous. Then,*

$$\|(\mathsf{I} - \gamma\nabla g)(\boldsymbol{x}) - (\mathsf{I} - \gamma\nabla g)(\boldsymbol{z})\|_2 \leq \max\{|1 - \gamma\mu|, |1 - \gamma\lambda|\}\|\boldsymbol{x} - \boldsymbol{z}\|_2 \quad \forall \boldsymbol{x}, \boldsymbol{z} \in \mathcal{X} \ .$$

Proof. Since for any $\boldsymbol{x}, \boldsymbol{z} \in \mathcal{X}$, the function g is strongly convex with constant μ, this lemma is a simple modification of Lemma 7 in [23]. □

Lemma 7. *For a convex and continuously differentiable function* g*, we have*

$$\nabla g \text{ is } \lambda\text{-Lipschitz continuous} \quad \Leftrightarrow \quad \nabla g \text{ is } (1/\lambda)\text{-cocoercive} \ .$$

Proof. See Theorem 2.1.5 in Section 2.1 of [79]. □

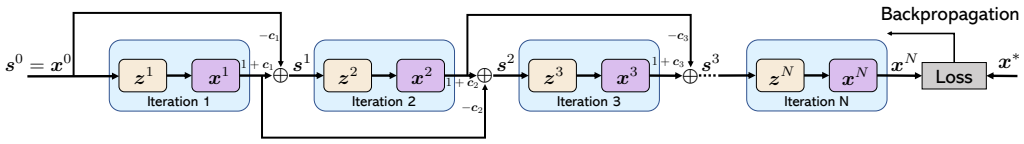


Figure 8: *Detailed architecture used for training the AR operator by unrolling the iterations of PnP-FISTA [84] with the DnCNN prior. Each layer contains one iteration consisting of a data-consistency update and an image prior update. The input of the unrolling network is the initialization $\boldsymbol{x}^0$ and the output is the reconstructed image from the Nth iteration, which is subsequently used within the training loss. In order to make the AR operator satisfy the Assumption A, we impose the spectral normalization and weight sharing on DnCNN across different iterations. Note that once DnCNN is pre-trained following this scheme, it is used as an AR operator within PnP.*

E Additional Technical Details and Numerical Results

We designed two types of deep priors for PnP/RED: (i) an AWGN denoiser and (ii) an artifact-removal (AR) operator trained to remove artifacts specific to the PnP iterations[2]. Both of these deep priors share the same neural network architecture, based on DnCNN [39]. The networks contain three components. The first part is a composite convolutional layer, consisting of a normal convolutional layer and a rectified linear units (ReLU) layer. It convolves the $n_1 \times n_2$ input to $n_1 \times n_2 \times 64$ features maps by using 64 filters of size 3×3. The second part is a sequence of 10 composite convolutional layers, each having 64 filters of size $3 \times 3 \times 64$. Those composite layers further process the feature maps generated by the first part. The third part of the network, a single convolutional layer, generates the final output image by convolving the feature maps with a $3 \times 3 \times 64$ filter. Every convolution is performed with a stride $= 1$, so that the intermediate feature maps share the same spatial size of the input image. We train several denoisers to optimize the *mean squared error (MSE)* by using the Adam optimizer. All the experiments in this work were performed on a machine equipped with an Intel Xeon Gold 6130 Processor and eight NVIDIA GeForce RTX 2080 Ti GPUs.

We now present the implementation details of training the AR operators used in this work. Inspired by ISTA-Net$^+$ [3], we implement our own deep unfolding neural network for training the AR operator. Given an initial solution $\boldsymbol{x}^0$, *i.e.* $\boldsymbol{x}^0 = \boldsymbol{A}^\mathsf{T}\boldsymbol{y}$, we iteratively refine it by infusing information from both the gradient of the data-fidelity term ∇g and the learned operator D defined as

$$\mathsf{R}(\boldsymbol{x}) = (\mathsf{I} - \mathsf{D})(\boldsymbol{x}) = \boldsymbol{x} - \mathsf{D}(\boldsymbol{x}) \;, \tag{17}$$

where R is the residual of the deep neural network. We use Nesterov acceleration in the unrolled architecture, fixing the total number of unrolling iterations to $N \geq 1$

$$\boldsymbol{z}^k = \boldsymbol{s}^{k-1} - \gamma \nabla g(\boldsymbol{s}^{k-1}) \tag{18}$$

$$\boldsymbol{x}^k = \mathsf{D}(\boldsymbol{z}^k) \tag{19}$$

$$c_k = (q_{k-1} - 1)/q_k \tag{20}$$

$$\boldsymbol{s}^k = \boldsymbol{x}^k + c_k(\boldsymbol{x}^k - \boldsymbol{x}^{k-1}) \;, \tag{21}$$

where $\gamma > 0$ is a step-size parameter and the value of $q_k = 1/2(1 + \sqrt{1 + 4q_{k-1}^2})$ is adapted during the training for better PSNR performance. Fig. 8 illustrates the algorithmic details for training the AR operator. In our implementation, we opted to share the weights of the AR operator across different iterations to satisfy our theoretical assumptions. We trained several AR operators for N unfolded iterations using the MSE loss

$$\mathcal{L}_{\text{MSE}} = \frac{1}{M} \sum_{j=1}^{M} \|\boldsymbol{x}_j^N - \boldsymbol{x}_j^*\|_2^2 \;, \tag{22}$$

[2]The implementation of our pre-trained denoisers and AR operators are also available in the supplement.

[3]ISTA-Net$^+$ is publicly available at `https://github.com/jianzhangcs/ISTA-Net-PyTorch`.

where $\boldsymbol{x}_j^*$ is the ground truth. We also included a *smoothness-constraint* loss across different iterations, defined as

$$\mathcal{L}_{\textbf{Smooth}} = \frac{1}{M} \sum_{j=1}^{M} \sum_{k=N-q}^{N} \|\boldsymbol{x}_j^k - \mathsf{D}(\boldsymbol{z}_j^k)\|_2^2 \,. \tag{23}$$

We observe that the AR operators trained with this smoothness-constraint outperform those trained without it, especially, when the AR operator is integrated into the PnP algorithm. The total AR training loss is thus $\mathcal{L} = \mathcal{L}_{\textbf{MSE}} + \beta \mathcal{L}_{\textbf{Smooth}}$, where $\beta > 0$ controls the amount of smoothing. For the experiments in this paper, we set $N = 90, \beta = 10$ for all gray and color AR operators' training, while we set $N = 27, \beta = 10$ for CS-MRI.

We used a pre-training strategy to accelerate the training of the weights within the AR operator. Since the weights are shared across iterations of the deep unfolding network, we can then initialize them with those obtained from pre-trained AWGN denoisers. We observe that this pre-training strategy is considerably more efficient than initializing the entire unfolding network with the random weights. Since we initialize our learned components with deep denoisers, the initial setup for our method exactly corresponds to tuning a PnP approach with a deep denoiser. Such training adapts the operator D to a particular inverse problem and data distribution.

In Fig. 6, we report the empirical evaluation of μ for the measurement operators used in our experiments by sampling images from $\mathsf{Im}(\mathsf{D})$. Specifically, we test two types of measurement matrixes for CS, namely random matrix and radially subsampled Fourier matrix, both at subsampling rates of 10% and 50%. For each type of matrix, we first use the operator D to generate several denoised image pairs on BSD68 and medical brain images, respectively. This ensures the tested image pairs are all in the range of D. We plot the histograms of $\mu = \|\boldsymbol{A}\boldsymbol{x} - \boldsymbol{A}\boldsymbol{z}\|_2^2 / \|\boldsymbol{x} - \boldsymbol{z}\|_2^2$, and the minimum value of each histogram is indicated by a vertical bar, providing an empirical lower bound on the values of μ. Fig. 6 illustrates that empirically the measurement operators $\boldsymbol{A}$ used in this work satisfies S-REC over $\mathsf{Im}(\mathsf{D})$ with $\mu > 0$.

In Fig. 7, we report the convergence of $\|\mathsf{R}(\boldsymbol{x}^t)\|_2^2 = \|\boldsymbol{x}^t - \mathsf{D}(\boldsymbol{x}^t)\|_2^2$ for both the AWGN denoisers and the AR operators use in our experiments. As can be observed from the plots, in both cases, PnP converges to vectors close to $\mathsf{Zer}(\mathsf{R})$, which is consistent with the view that it regularizes inverse problems by obtaining solutions near the fixed-points of a denoiser/AR operator. Note that this view is completely backward compatible with the traditional sparsity-promoting priors and ISTA-algorithms, where one achieves regularization by promoting sparse solutions in some transform domain.

We provide additional visualizations of the solutions produced by PnP/RED and various baseline methods considered in our work. Fig. 9 (top) reports the visual comparison of multiple methods on Set11 with CS ratios of 10%, while Fig. 9 (bottom) reports the comparison on medical brain images for CS-MRI with under-sampling ratios of 20%. Fig. 10 illustrates the numerical comparison on BSD68 for CS ratios of 30% (top) and 10% (bottom), respectively. Fig. 11 reports the visual comparison between PnP/RED and two CS methods based on StyleGAN2. Note that in all figures, PnP/RED achieves competitive results, with PnP (AR) achieving superior reconstruction results compared to PnP (denoising).

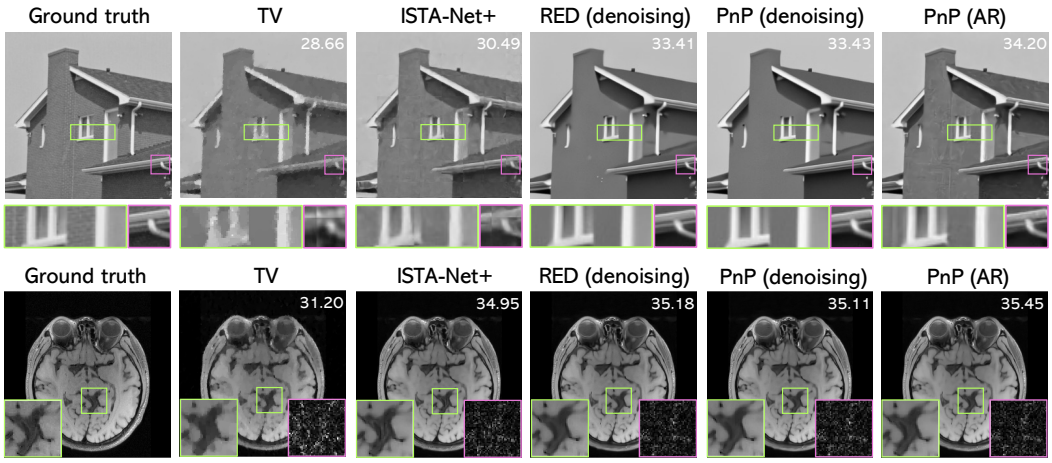


Figure 9: *Additional visual comparisons between various methods for CS and CS-MRI. Top: reconstruction results on the "House" image in Set 11 at CS ratios of* 10%*. Bottom: results on MRI images with radially under-samplying at CS ratios of* 20% *(The pink box provides the error residual that was amplified by* $10\times$ *for better visualization.). Best viewed by zooming in the display.*

Figure 10: *Supplementary visual comparisons between various methods on BSD68. Top: Results at* 30% *sampling ratio. Bottom: Results at* 10% *sampling ratio.*

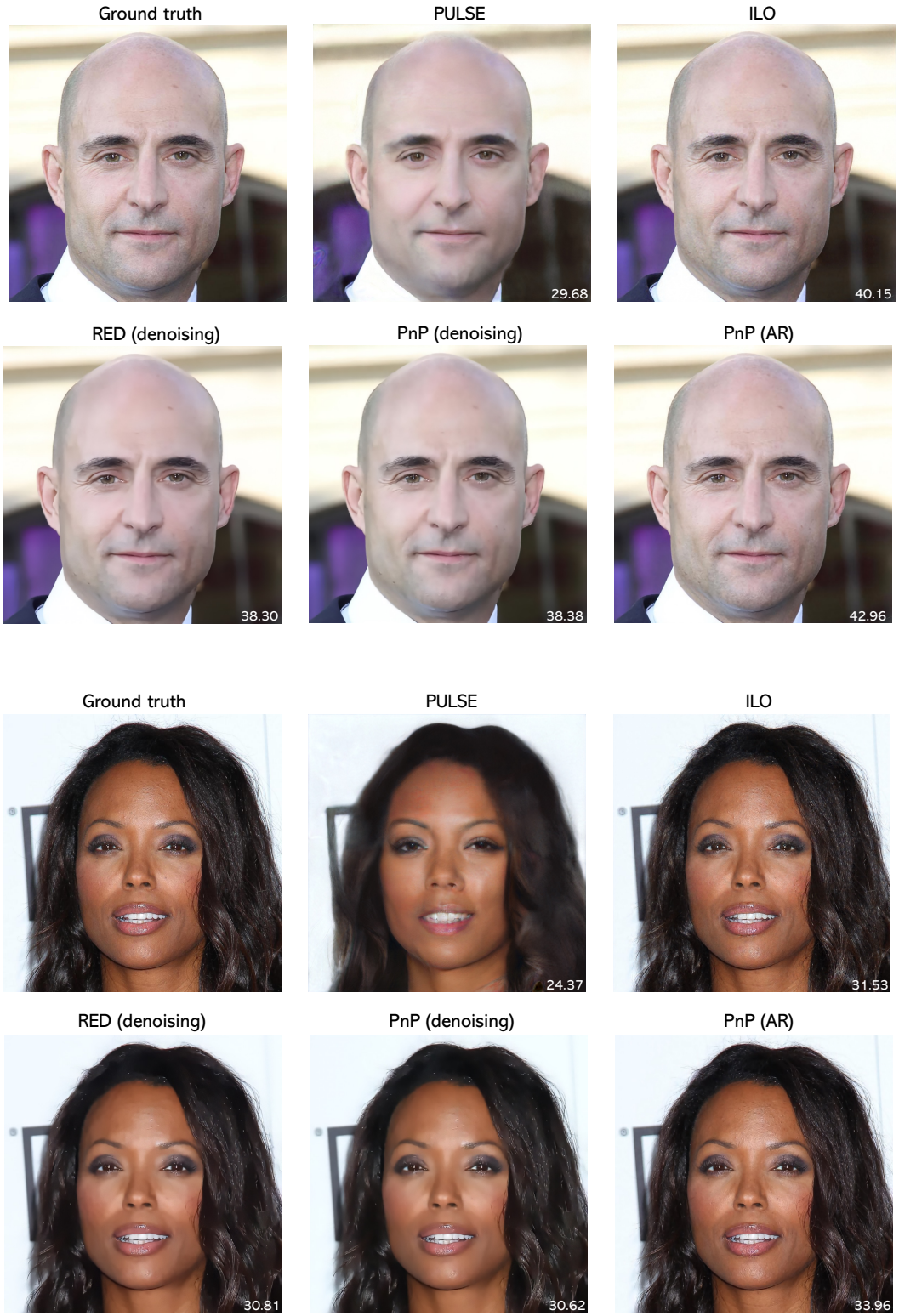


Figure 11: *Supplementary visual comparison between PnP/RED and two methods using generative models, when applied to the CelebA HQ dataset at* 10% *sampling ratio. Note the similarity between the RED and PnP solutions. PnP (AR) leads to sharp images, comparable to those obtained by ILO with StyleGAN2. This highlights the benefit of using pre-trained AR operators.*