

# Convergent Primal-Dual Plug-and-Play Image Restoration: A General Algorithm and Applications

Yodai Suzuki, *Student Member, IEEE*, Ryosuke Isono, *Graduate Student Member, IEEE*, and Shunsuke Ono, *Senior Member, IEEE*

**Abstract**—We propose a general deep plug-and-play (PnP) algorithm with a theoretical convergence guarantee. PnP strategies have demonstrated outstanding performance in various image restoration tasks by exploiting the powerful priors underlying Gaussian denoisers. However, existing PnP methods often lack theoretical convergence guarantees under realistic assumptions due to their ad-hoc nature, resulting in inconsistent behavior. Moreover, even when convergence guarantees are provided, they are typically designed for specific settings or require a considerable computational cost in handling non-quadratic data-fidelity terms and additional constraints, which are key components in many image restoration scenarios. To tackle these challenges, we integrate the PnP paradigm with primal-dual splitting (PDS), an efficient proximal splitting methodology for solving a wide range of convex optimization problems, and develop a general convergent PnP framework. Specifically, we establish theoretical conditions for the convergence of the proposed PnP algorithm under a reasonable assumption. Furthermore, we show that the problem solved by the proposed PnP algorithm is not a standard convex optimization problem but a more general monotone inclusion problem, where we provide a mathematical representation of the solution set. Our approach efficiently handles a broad class of image restoration problems with guaranteed theoretical convergence. Numerical experiments on specific image restoration tasks validate the practicality and effectiveness of our theoretical results.

**Index Terms**—Image restoration, plug-and-play (PnP) algorithms, primal-dual splitting (PDS), convergence guarantee.

## I. INTRODUCTION

### A. Image Restoration by Proximal Splitting Algorithms

IMAGE restoration is a longstanding and essential problem with diverse applications, ranging from remote sensing, geoscience, and astronomy to biomedical imaging. In general, it is reduced to the inverse problem of estimating an original image from an observed image, which is degraded by some linear observation process and noise contamination (e.g., additive white Gaussian noise and Poisson noise).

Since this inverse problem is often ill-posed or ill-conditioned, a standard approach formulates the restoration

task as a convex optimization problem to characterize a desirable solution. In this scheme, we typically minimize the sum of a data-fidelity term with respect to a given observed image and a regularization term reflecting prior knowledge on natural images. Typically, the data-fidelity is chosen as the negative log-likelihood of the statistical distribution assumed for the noise contamination. Conversely, the choice of the regularization term is highly diverse. One of the most popular techniques is Total Variation (TV) [1] and its extensions [2]–[5], which promote the spatial piecewise smoothness of images. In any case, these data-fidelity and regularization terms are possibly nonsmooth, which makes it difficult to solve the convex optimization problem analytically or using gradient-based methods.

This difficulty has been overcome by proximal splitting algorithms, which assume that the proximal operators associated with these terms can be computed efficiently [6]–[8]. Notable examples include Forward-Backward Splitting (FBS) [9], Alternating Direction Method of Multipliers (ADMM) [10], and Primal-Dual Splitting (PDS) [11]–[14]. Among these algorithms, PDS offers significant flexibility and efficiency, as it can handle nonsmooth convex terms involving linear operators without requiring matrix inversions or performing inner iterations, which is generally impossible in the case of FBS or ADMM. These features offer several advantages for image restoration applications [15], [16] (refer to Section III-B for more detailed explanation with specific formulations).

### B. Plug-and-Play (PnP) Algorithms

Recently, regularization using Gaussian denoisers has become an active area of research. This line of work was pioneered by a PnP paradigm that was proposed in [20] and later extended, e.g., in [21], [28], [29]. The basic idea of PnP approaches is to replace the proximity operator of the regularization term used in proximal splitting algorithms with an off-the-shelf Gaussian denoiser, such as Block-Matching and 3D Filtering (BM3D) [30], Non-Local Means (NLM) [31], Trainable Nonlinear Reaction Diffusion (TNRD) [32], and Denoising Convolutional Neural Networks (DnCNN) [33]. Numerous studies have shown that the PnP strategy achieves superior performance across a variety of applications, including tomographic imaging [20], [22], biomedical imaging [34], magnetic resonance imaging [35], and remote sensing [36] (see [37] for a recent review).

Following these impressive results, several works explore attempts to plug image restoration sophisticated denoisers, Zhang et

Y. Suzuki is with the Department of Computer Science, Institute of Science Tokyo, Yokohama, 226-8503, Japan (e-mail: suzuki.y.bd8c@m.isct.ac.jp).

R. Isono is with the Department of Computer Science, Institute of Science Tokyo, Yokohama, 226-8503, Japan (e-mail: isono.r.1f44@m.isct.ac.jp).

S. Ono is with the Department of Computer Science, Institute of Science Tokyo, Yokohama, 226-8503, Japan (e-mail: ono@comp.isct.ac.jp).

This work was supported in part by JST FOREST under Grant JPMJFR232M, JST AdCORP under Grant JPMJKB2307, JST CREST under Grant JPMJCR25Q5, JST ACT-X under Grant JPMJAX24C1, JST BOOST, Japan Grant Number JPMJBS2430, and in part by JSPS KAKENHI under Grant 22H03610, 22H00512, 23H01415, 23K17461, 24K03119, 24K22291, 25H01296, and 25K03136.

The code is available at [https://github.com/MDI-TokyoTech/Convergent\\_Primal-Dual\\_PnP\\_Plug-Image\\_Restoration\\_Denoisers\\_Zhang](https://github.com/MDI-TokyoTech/Convergent_Primal-Dual_PnP_Plug-Image_Restoration_Denoisers_Zhang).

TABLE I: Pros and Cons of Existing PnP Methods and the Proposed Method.

Algorithms	Authors	Data-fidelity	Additional constraints	Inversion free	Convergence guarantees	Solution characterization	Assumption on denoiser $J$
PnP-FBS	Sun et.al. [17]	Quadratic	—	✓	✓	—	$J$ is averaged.
	Pesquet et.al. [18]	Quadratic	—	✓	✓	✓	$J$ is firmly nonexpansive.
	Ebner et al. [19]	Quadratic	—	✓	✓	✓	$J$ is contractive.
PnP-ADMM	Venkatakrishnan et al. [20]	Any*	—	—	—	—	—
	Rond et al. [21]	GKL div.	—	—	—	—	—
	Sreehari et al. [22]	Any*	—	—	✓	—	$\nabla J$ is a symmetric matrix, etc.
	S. H. Chan et al. [23]	Any*	—	—	✓	—	$J$ is bounded, etc.
	P. Nair et al. [24]	Any*	—	—	✓	—	$J$ is firmly nonexpansive.
	Sun et al. [25]	Any*	✓	—	✓	✓	$J$ is firmly nonexpansive.
PnP-PDS	Ono [26]	Any*	✓	✓	—	—	—
	Garcia et al. [27]	Constraint	✓	✓	✓	✓	$J$ is firmly nonexpansive.
	Ours	Any*	✓	✓	✓	✓	$J$ is firmly nonexpansive.

\* Data-fidelity functions should be proper lower-semicontinuous convex, and their proximal operators should have closed-form expressions or be efficiently computable.

al. proposed a neural network specifically designed for use in PnP methods [38], known as DRUNet. Duff et al. proposed *generative regularizers* [39], realized by Autoencoders (AE), Variational Autoencoders (VAE) [40], and Generative Adversarial Networks (GAN) [41]. Zhu et al. showed that diffusion models [42] also have promising performance as prior knowledge used in PnP methods [43]. We should mention that despite such extensive research, the theoretical convergence guarantee for PnP methods remains a challenging open question [17], [19], [22], [23], [44]. This is mainly because replacing the proximity operator of the regularization term with a denoiser does not necessarily guarantee the existence of the underlying convex optimization problem.

To explore explicit data-driven regularization, alternative frameworks including regularization by denoising (RED) [45], [46], gradient step denoisers (GSD) [47]–[50], and its generalization in Bregman geometry [51] have been proposed. Unlike PnP strategies, they explicitly define a regularization term derived from the output of a denoiser, incorporating it directly into the formulation of the optimization problem. Subsequently, they construct algorithms to solve this problem as a special case of traditional proximal splitting algorithms. Nonetheless, it should be noted that they still rely on conditions that may not be desirable in real-world settings. For example, RED imposes strict assumptions on the denoiser to guarantee its convergence, which are often impractical for deep denoisers [52] (see Section III-C4 for details). The GSD approach introduces a non-convex regularization term, which makes it difficult to guarantee convergence to globally optimal solutions and may cause the final solution to depend on the initialization.

In contrast, within the context of PnP methods, some studies have been conducted to establish their convergence under realistic assumptions, focusing on the firm nonexpansiveness of denoisers. Pesquet et al. showed theoretically that a subclass of maximally monotone operators (MMO) can be fully represented by the firmly nonexpansive DnCNN with a certain network structure. They also constructed a PnP method based on forward-backward splitting (PnP-FBS) with convergence guarantees for image restoration [18]. As highlighted by the authors, their technique can be extended to any algorithm, as long as its analysis relies on the theory of MMOs, including

PDS. Sun et al. proposed a PnP based on ADMM (PnP-ADMM) with the convergence guarantee by plugging in a firmly nonexpansive denoiser [25]. Hertrich et al. [53] developed a method to train averaged CNNs by stochastic gradient descent on the Stiefel manifold.

Regarding the flexibility of PnP algorithms, PnP based on PDS (PnP-PDS) offers the advantage of efficiently handling a wider range of image restoration problems than PnP-FBS and PnP-ADMM, as mentioned above. Ono proposed the first PnP-PDS and demonstrated its practicality, while no theoretical convergence guarantees were provided [26]. On the other hand, Garcia et al. proposed a convergent PnP-PDS with a firmly nonexpansive denoiser for a specific case and achieved successful results in the restoration of images obtained by a photonic lantern [27]. We remark that the denoiser assumption of firmly nonexpansiveness is reasonable in actual applications, as shown in [18]. Nevertheless, the convergence of PnP-PDS in its general form has not yet been fully analyzed. Once a general PnP-PDS with convergence guarantees is established, various image restoration problems can be approached with rich priors of Gaussian denoisers in an efficient and flexible fashion.

### C. Contributions and Paper Organization

Now, the following natural question arises: *Can we construct a general PnP-PDS with theoretical convergence guarantees under realistic assumptions on denoisers?* In this paper, we answer this question by proposing a general convergent PnP-PDS strategy with a theoretical guarantee that has a wide range of practical applications. The main contributions of our paper are outlined below:

- We study a general PnP-PDS algorithm that extends the framework proposed in [26] and establish its theoretical convergence conditions, building upon the foundational results presented in [18].
- We show that the problem solved by our PnP-PDS algorithm is not a standard convex optimization problem but a certain monotone inclusion problem, and provide a mathematical representation of the solution set.
- Based on the above foundations, we show that our PnP-PDS efficiently solves typical inverse problems formulated with non-quadratic data-fidelity, with convergence guarantees. In

particular, it allows the use of an  $\ell_2$ -ball data-fidelity constraint for image restoration under Gaussian noise, offering more flexibility in parameter selection and improving the versatility of the denoiser. Moreover, it effectively handles the generalized Kullback–Leibler (GKL) divergence for image restoration under Poisson noise.

In Table I, we briefly summarize how the proposed method differs from related works (see Section III-C for detailed and comprehensive discussion). The advantages of the proposed convergent PnP-PDS are illustrated by experimental results on a number of typical image restoration tasks.

The remainder of this paper is organized as follows. In Section II, we set up proximal tools, the PDS algorithm, and the firmly nonexpansive denoiser to construct our PnP-PDS. Section III presents our main results. We first establish a convergence guarantee for the general PnP-PDS algorithm and characterize its solutions. Subsequently, we introduce several specific forms of the algorithm designed for image restoration tasks, along with their corresponding solution characterizations. In Section IV, we conduct image restoration experiments, focusing on deblurring and inpainting tasks under two noise conditions: Gaussian noise and Poisson noise, and demonstrate that our PnP-PDS achieves state-of-the-art performance and high stability.

The preliminary version of this paper, without the applications to deblurring and inpainting under Poisson noise, the introduction of a box constraint, and the complete theoretical proof, can be found in the conference proceedings [54].

## II. PRELIMINARIES

### A. Notations

Let  $\|\cdot\|_1$  and  $\|\cdot\|_2$  denote the  $\ell_1$ -norm and  $\ell_2$ -norm, respectively. The transpose of a vector is denoted by  $(\cdot)^\top$ . The notation  $[\cdot]_m$  indicates the  $m$ -th element of a vector. The adjoint of a bounded linear operator is denoted by  $(\cdot)^*$ , and the operator norm is represented as  $\|\cdot\|_{\text{op}}$ .

Let  $\mathcal{H}$  be a real Hilbert space with its inner product  $\langle \cdot, \cdot \rangle$  and norm  $\|\cdot\|$ . The notation  $\Gamma_0(\mathcal{H})$  represents the set of proper lower-semicontinuous convex functions from  $\mathcal{H}$  to  $\mathbb{R} \cup \{\infty\}$ . For  $\mathcal{F} \in \Gamma_0(\mathcal{H})$ , its effective domain is  $\text{dom}(\mathcal{F}) := \{\mathbf{x} \in \mathcal{H} : \mathcal{F}(\mathbf{x}) < \infty\}$ .

For  $D \subset \mathcal{H}$ , we consider an operator  $T : D \rightarrow \mathcal{H}$ . If  $\|T\mathbf{x}_1 - T\mathbf{x}_2\| \leq \|\mathbf{x}_1 - \mathbf{x}_2\|$  holds for every  $(\mathbf{x}_1, \mathbf{x}_2) \in D^2$ ,  $T$  is said to be nonexpansive. When  $2T - \text{Id}$  is nonexpansive,  $T$  is called firmly nonexpansive [55, Proposition 4.4] ( $\text{Id}$  denotes an identity operator). If  $\kappa T$  is firmly nonexpansive for some  $\kappa \in (0, \infty)$ , then  $T$  is  $\kappa$ -cocoercive.

Let  $A : \mathcal{H} \rightarrow 2^{\mathcal{H}}$  be a set-valued operator. Its graph is  $\text{gra}A = \{(\mathbf{x}, \mathbf{y}) \in \mathcal{H}^2 : \mathbf{y} \in A\mathbf{x}\}$ . If  $\langle \mathbf{x}_1 - \mathbf{x}_2, \mathbf{y}_1 - \mathbf{y}_2 \rangle \geq 0$  holds for every  $(\mathbf{x}_1, \mathbf{y}_1) \in \text{gra}A$  and  $(\mathbf{x}_2, \mathbf{y}_2) \in \text{gra}A$ ,  $A$  is said to be monotone. The operator  $A$  is called maximally monotone if there exists no monotone operator  $B : \mathcal{H} \rightarrow 2^{\mathcal{H}}$  such that  $\text{gra}B$  properly contains  $\text{gra}A$ .

We introduce several specific convex functions and sets that we use in this paper. Let  $C$  be a nonempty closed convex set

on  $\mathcal{H}$ . The indicator function of  $C$ , denoted by  $\iota_C$ , is defined as

$$\iota_C(\mathbf{x}) := \begin{cases} 0, & \text{if } \mathbf{x} \in C, \\ \infty, & \text{otherwise.} \end{cases}$$

Let  $B_{2,\varepsilon}^{\mathbf{v}}$  be the  $\mathbf{v}$ -centered  $\ell_2$ -norm ball with radius  $\varepsilon > 0$ , which is given by

$$B_{2,\varepsilon}^{\mathbf{v}} := \{\mathbf{x} \in \mathcal{H} \mid \|\mathbf{x} - \mathbf{v}\|_2 \leq \varepsilon\}.$$

The Generalized Kullback–Leibler (GKL) divergence between  $\mathbf{x} \in \mathbb{R}^N$  and  $\mathbf{v} \in \mathbb{R}^N$  is defined as

$$\begin{aligned} \text{GKL}_{\mathbf{v}}(\mathbf{x}) \\ := \sum_{i=1}^N \begin{cases} \eta[\mathbf{x}]_i - [\mathbf{v}]_i \ln \eta[\mathbf{x}]_i, & \text{if } [\mathbf{v}]_i > 0 \text{ and } [\mathbf{x}]_i > 0, \\ \eta[\mathbf{x}]_i, & \text{if } [\mathbf{v}]_i = 0 \text{ and } [\mathbf{x}]_i \geq 0, \\ \infty, & \text{otherwise,} \end{cases} \quad (1) \end{aligned}$$

where  $\eta > 0$  is a certain scaling parameter, which will be detailed in Section III-B2.

### B. Proximal Tools

For  $\mathcal{F} \in \Gamma_0(\mathcal{H})$ , the proximity operator of an index  $\gamma > 0$  is defined as follows [56]:

$$\text{prox}_{\gamma\mathcal{F}}(\mathbf{x}) := \underset{\mathbf{y} \in \mathcal{H}}{\text{argmin}} \mathcal{F}(\mathbf{y}) + \frac{1}{2\gamma} \|\mathbf{y} - \mathbf{x}\|_2^2.$$

Let us consider the convex conjugate function of  $\mathcal{F}$ , defined as follows:

$$\mathcal{F}^*(\mathbf{x}) := \sup_{\mathbf{y} \in \mathcal{H}} \langle \mathbf{x}, \mathbf{y} \rangle - \mathcal{F}(\mathbf{y}).$$

According to the Moreau's identity [57, Theorem 3.1 (ii)],  $\text{prox}_{\gamma\mathcal{F}^*}$  can be computed via  $\text{prox}_{\mathcal{F}/\gamma}$  as

$$\text{prox}_{\gamma\mathcal{F}^*}(\mathbf{x}) = \mathbf{x} - \gamma \text{prox}_{\mathcal{F}/\gamma}(\mathbf{x}/\gamma).$$

The proximity operator of  $\iota_C$  equals to the metric projection onto  $C$ , i.e.,

$$\text{prox}_{\iota_C}(\mathbf{x}) = P_C(\mathbf{x}) := \underset{\mathbf{y} \in C}{\text{argmin}} \|\mathbf{y} - \mathbf{x}\|_2.$$

In particular, it can be calculated for  $C := B_{2,\varepsilon}^{\mathbf{v}}$  as follows:

$$\text{prox}_{\iota_{B_{2,\varepsilon}^{\mathbf{v}}}}(\mathbf{x}) = \begin{cases} \mathbf{x}, & \text{if } \mathbf{x} \in C, \\ \mathbf{v} + \frac{\varepsilon(\mathbf{x} - \mathbf{v})}{\|\mathbf{x} - \mathbf{v}\|_2}, & \text{otherwise.} \end{cases}$$

The proximity operator of the GKL divergence can be computed as follows:

$$[\text{prox}_{\gamma\text{GKL}_{\mathbf{v}}}(\mathbf{x})]_i = \frac{1}{2}([\mathbf{x}]_i - \gamma\eta + \sqrt{([\mathbf{x}]_i - \gamma\eta)^2 + 4\gamma[\mathbf{v}]_i}).$$

### C. Primal-Dual Splitting Algorithm

Let us review a PDS algorithm [13], [14], which is a basis of our PnP strategy. Let  $\mathcal{X}$  and  $\mathcal{Y}$  be two real Hilbert spaces. Suppose that  $f \in \Gamma_0(\mathcal{X})$  is differentiable with its gradient  $\nabla f$  being  $\beta$ -Lipschitz continuous with  $\beta > 0$ ,  $g \in \Gamma_0(\mathcal{X})$  and  $h \in \Gamma_0(\mathcal{Y})$  are proximable functions, i.e., the proximity operators have a closed form solution or can be computed efficiently, and  $\mathbf{L} : \mathcal{X} \rightarrow \mathcal{Y}$  is a bounded linear operator. Consider the following convex optimization problem:

$$\underset{\mathbf{x} \in \mathcal{X}}{\text{minimize}} f(\mathbf{x}) + g(\mathbf{x}) + h(\mathbf{L}\mathbf{x}), \quad (2)$$

where the set of solutions to (2) is assumed to be nonempty. Its dual formulation is given as follows [55, Chap. 15]:

$$\underset{\mathbf{y} \in \mathcal{Y}}{\text{minimize}} (f + g)^*(-\mathbf{L}^*\mathbf{y}) + h^*(\mathbf{y}). \quad (3)$$

The PDS algorithm jointly solves both (2) and (3) by the following iterative procedure:

$$\begin{cases} \mathbf{x}_{n+1} = \text{prox}_{\gamma_1 g}(\mathbf{x}_n - \gamma_1(\nabla f(\mathbf{x}_n) + \mathbf{L}^*\mathbf{y}_n)), \\ \mathbf{y}_{n+1} = \text{prox}_{\gamma_2 h^*}(\mathbf{y}_n + \gamma_2 \mathbf{L}(2\mathbf{x}_{n+1} - \mathbf{x}_n)), \end{cases} \quad (4)$$

where  $\gamma_1 > 0$  and  $\gamma_2 > 0$  are parameters that satisfy

$$\gamma_1 \left( \frac{\beta}{2} + \gamma_2 \|\mathbf{L}\|_{\text{op}}^2 \right) < 1.$$

Additionally, we consider the following monotone inclusion problem derived from the optimality conditions of (2) and (3):

$$\begin{aligned} &\text{Find } (\hat{\mathbf{x}}, \hat{\mathbf{y}}) \in \mathcal{X} \times \mathcal{Y} \\ &\text{s.t. } \begin{pmatrix} \mathbf{0} \\ \mathbf{0} \end{pmatrix} \in \begin{pmatrix} \partial g(\hat{\mathbf{x}}) + \mathbf{L}^*\hat{\mathbf{y}} + \nabla f(\hat{\mathbf{x}}) \\ -\mathbf{L}\hat{\mathbf{x}} + \partial h^*(\hat{\mathbf{y}}) \end{pmatrix}, \end{aligned} \quad (5)$$

where  $\partial(\cdot)$  denotes the subdifferential of a convex function. Under a certain mild condition, the solutions to (5) coincide with the solutions to (2) and (3) (see [13] and [14] for details).

### D. Firmly Nonexpansive Denoiser [18]

We review the method proposed in [18] for training a firmly nonexpansive denoiser represented by DnCNN, which is essential for the construction of convergent PnP algorithms. Basically, they add a specific penalty during training to ensure the firm nonexpansiveness of the denoiser.

Let  $J : \mathbb{R}^K \rightarrow \mathbb{R}^K$  be a Gaussian denoiser represented by DnCNN. To train a firmly nonexpansive denoiser, we first consider the operator  $Q := 2J - \text{Id}$ , which is assumed to be differentiable.

By the definition of firm nonexpansiveness,  $J$  is firmly nonexpansive if and only if  $Q$  is nonexpansive. Thus, our objective is to ensure the nonexpansiveness of  $Q$  rather than the firm nonexpansiveness of  $J$ . If  $Q$  is differentiable, its nonexpansiveness is equivalent to its Jacobian  $\nabla Q$  satisfying the following condition for all  $\mathbf{x} \in \mathbb{R}^K$ :

$$\|\nabla Q(\mathbf{x})\|_{\text{sp}} \leq 1, \quad (6)$$

where  $\|\cdot\|_{\text{sp}}$  represents the spectral norm.

Based on the above fact, the authors of [18] proposed a penalty to encourage  $Q$  to satisfy the condition in (6), and defined a new loss function expressed by the sum of the

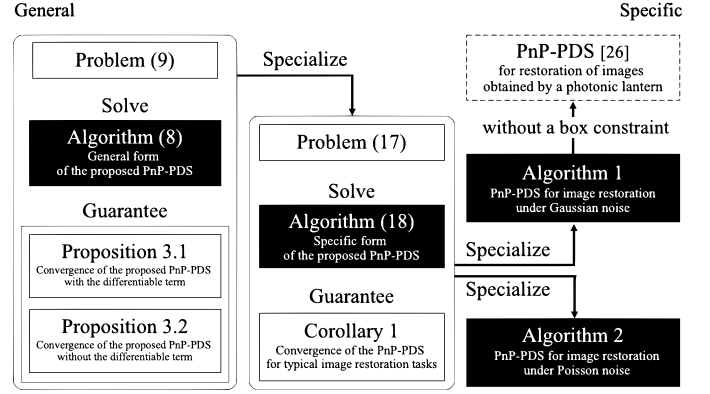


Fig. 1: A flowchart showing the basic structure of the theoretical discussions in this paper. The dotted and solid lines represent existing and new results, respectively.

squared prediction error and the penalty. Specifically, the loss function is calculated for each image as

$$\|J(\mathbf{x}_\ell) - \bar{\mathbf{x}}_\ell\|_2^2 + \tau \max(\|\nabla Q(\tilde{\mathbf{x}}_\ell)\|_{\text{sp}}^2, 1 - \xi). \quad (7)$$

Here,  $\tau > 0$  is a penalization parameter,  $\xi > 0$  is a parameter that defines a safety margin,  $\bar{\mathbf{x}}_\ell$  is the  $\ell$ -th ground truth image in the training dataset,  $\mathbf{x}_\ell$  is the image corrupted with Gaussian noise, and  $\tilde{\mathbf{x}}_\ell$  is the image calculated by the following equation:

$$\tilde{\mathbf{x}}_\ell = \rho \bar{\mathbf{x}}_\ell + (1 - \rho) J(\mathbf{x}_\ell), \quad (8)$$

where  $\rho$  is a random value chosen according to a uniform distribution in the range from 0 to 1.

Note that  $\|\nabla Q(\tilde{\mathbf{x}}_\ell)\|_{\text{sp}}$  can be computed using the power iterative method [58]. For more detailed information on the training method, see [18, Section 3.2].

## III. PROPOSED METHOD

We propose a general convergent PnP-PDS, demonstrate its application to image restoration with specific cases of our PnP-PDS, and provide detailed discussions on its differences from related methods. For clarity, we present a flowchart in Fig. 1 that illustrates the relationships between our theoretical results.

### A. General Form of Convergent PnP-PDS and Its Convergence Analysis

We propose a convergent PnP-PDS that substitutes the proximity operator of  $g$  in (4) with a firmly nonexpansive denoiser  $J$  represented by DnCNN. The iterative procedures of our PnP-PDS is given by

$$\begin{cases} \tilde{\mathbf{x}}_{n+1} = J(\mathbf{x}_n - \gamma_1(\nabla f(\mathbf{x}_n) + \mathbf{L}^*\mathbf{y}_n)), \\ \tilde{\mathbf{y}}_{n+1} = \text{prox}_{\gamma_2 h^*}(\mathbf{y}_n + \gamma_2 \mathbf{L}(2\tilde{\mathbf{x}}_{n+1} - \mathbf{x}_n)), \\ \mathbf{x}_{n+1} = \rho_n \tilde{\mathbf{x}}_{n+1} + (1 - \rho_n) \mathbf{x}_n, \\ \mathbf{y}_{n+1} = \rho_n \tilde{\mathbf{y}}_{n+1} + (1 - \rho_n) \mathbf{y}_n, \end{cases} \quad (9)$$

where  $\gamma_1$ ,  $\gamma_2$  and  $\{\rho_n\}$  are the parameters and the definitions of  $f$ ,  $h$ , and  $\mathbf{L}$  are the same as in (2). The parameters  $\gamma_1$  and  $\gamma_2$  are fixed constants across the iterations.

The following two propositions present our main theoretical results on the convergence of PnP-PDS in (9). Proposition 3.1 corresponds to the case with the differentiable term  $f$ , while Proposition 3.2 corresponds to the case without it.

**Proposition 3.1.** *Suppose that  $\beta > 0$ . Let  $J : \mathcal{X} \rightarrow \mathcal{X}$  be a firmly nonexpansive denoiser and defined over the full domain. Then, there exists a maximally monotone operator  $A_J$  such that  $J = (A_J + \text{Id})^{-1}$ . Assume that the solution set of the following monotone inclusion problem:*

$$\begin{aligned} &\text{Find } (\hat{\mathbf{x}}, \hat{\mathbf{y}}) \in \mathcal{X} \times \mathcal{Y} \\ &\text{s.t. } \begin{pmatrix} \mathbf{0} \\ \mathbf{0} \end{pmatrix} \in \begin{pmatrix} \gamma_1^{-1} A_J(\hat{\mathbf{x}}) + \mathbf{L}^* \hat{\mathbf{y}} + \nabla f(\hat{\mathbf{x}}) \\ -\mathbf{L} \hat{\mathbf{x}} + \partial h^*(\hat{\mathbf{y}}) \end{pmatrix} \end{aligned} \quad (10)$$

is nonempty, and suppose that the following conditions hold:

- (i)  $\frac{1}{\gamma_1} - \gamma_2 \|\mathbf{L}\|_{\text{op}}^2 > \frac{\beta}{2}$ ,
- (ii)  $\rho_n \in (0, \delta)$ ,
- (iii)  $\sum_{n \in \mathbb{N}} \rho_n (\delta - \rho_n) = \infty$ ,

where  $\gamma_1 > 0$ ,  $\gamma_2 > 0$ , and  $\delta$  is defined as follows:

$$\delta := 2 - \frac{\beta}{2} \left( \frac{1}{\gamma_1} - \gamma_2 \|\mathbf{L}\|_{\text{op}}^2 \right)^{-1}.$$

Then, the sequence  $\{\mathbf{x}_n, \mathbf{y}_n\}_{n \in \mathbb{N}}$  generated by (9) weakly converges to a solution to (10).

**Proposition 3.2.** *Suppose that  $f = 0$ . Let  $J : \mathcal{X} \rightarrow \mathcal{X}$  be a firmly nonexpansive denoiser and defined over the full domain. Then, there exists a maximally monotone operator  $A_J$  such that  $J = (A_J + \text{Id})^{-1}$ . Assume that the solution set of (10) is nonempty, and the following conditions hold:*

- (i)  $\frac{1}{\gamma_1} - \gamma_2 \|\mathbf{L}\|_{\text{op}}^2 > 0$ ,
- (ii)  $\rho_n \in (0, 2)$ ,
- (iii)  $\sum_{n \in \mathbb{N}} \rho_n (2 - \rho_n) = \infty$ ,

where  $\gamma_1 > 0$  and  $\gamma_2 > 0$ . Then, the sequence  $\{\mathbf{x}_n, \mathbf{y}_n\}_{n \in \mathbb{N}}$  generated by (9) weakly converges to a solution to (10).

*Proof Sketch of Propositions 3.1 and 3.2.* Recall that the set-valued operator  $A_J = J^{-1} - \text{Id}$  is maximally monotone if and only if  $J$  is firmly nonexpansive and has the full domain [55, Proposition 23.8 (iii)]. Hence,  $\gamma_1^{-1} A_J$  is also maximally monotone [55, Proposition 20.22]. Therefore, Proposition 3.1 and Proposition 3.2 are justified by [13, Theorem 3.1] and [13, Theorem 3.2], respectively. These proofs build upon and directly apply the standard convergence analysis of the PDS algorithm. For completeness and ease of reference, we provide the detailed proofs of these propositions in the Appendix.  $\square$

**Remark 1** (Connection to the convex optimization problem in (2)). The monotone inclusion problem in (10) has the same form as (5) with  $\partial g = \gamma_1^{-1} A_J$ . The classical monotone operator theory states that there exists  $g \in \Gamma_0(\mathbb{R}^N)$  such that  $\partial g = \gamma_1^{-1} A_J$  if and only if  $\gamma_1^{-1} A_J$  is a *maximal cyclically monotone operator*, a specific type of maximally monotone operator [55, Theorem 22.18]. However, the operator  $\gamma_1^{-1} A_J$ , where  $J = (A_J + \text{Id})^{-1}$ , is not necessarily maximal cyclically monotone only because  $J$  is firmly nonexpansive. Thus, the existence of  $g \in \Gamma_0(\mathbb{R}^N)$  cannot always be guaranteed. Furthermore, it should be noted that  $A_J$  is scaled by  $\gamma_1^{-1}$  in

the solution set of the monotone inclusion problem (10). This implies that the value of  $\gamma_1$  affects the solution set, unlike the standard PDS algorithm.

### B. Applications to Image Restoration

Now we move on to the applications of our general PnP-PDS to image restoration. First, we derive a special case of our PnP algorithm in (9), which is set up for typical image restoration problems. Then, we demonstrate its application to specific image restoration tasks with two types of noise contamination: Gaussian and Poisson. In the following,  $\mathbf{u} \in \mathbb{R}^K$  and  $\mathbf{v} \in \mathbb{R}^K$  denote original and observed images, respectively. Let  $\Phi : \mathbb{R}^K \rightarrow \mathbb{R}^K$  be an operator representing some observation process, which may include blurring or random sampling. We assume that the pixel values of each image are normalized to the range of 0 to 1.

First, we introduce an observation model that covers many image restoration scenarios as follows:

$$\mathbf{v} = \mathcal{N}(\Phi \mathbf{u}), \quad (11)$$

where  $\mathcal{N} : \mathbb{R}^K \rightarrow \mathbb{R}^K$  represents noise contamination (not necessarily additive). Image restoration under (11) can be formulated as the following optimization problem:

$$\underset{\hat{\mathbf{u}} \in \mathbb{R}^K}{\text{minimize}} \mathcal{D}(\Phi \hat{\mathbf{u}}) + \mathcal{R}_J(\hat{\mathbf{u}}) \quad \text{s.t. } \hat{\mathbf{u}} \in C, \quad (12)$$

or equivalently,

$$\underset{\hat{\mathbf{u}} \in \mathbb{R}^K}{\text{minimize}} \mathcal{D}(\Phi \hat{\mathbf{u}}) + \mathcal{R}_J(\hat{\mathbf{u}}) + \iota_C(\hat{\mathbf{u}}),$$

where  $\mathcal{D} \in \Gamma_0(\mathbb{R}^K)$  is a proximable data-fidelity term derived from the noise model  $\mathcal{N}$ ,  $\mathcal{R}_J$  is a proximable regularization term,  $C$  is a nonempty closed convex set on  $\mathbb{R}^K$  to which  $\hat{\mathbf{v}}$  is assumed to belong, and  $\iota_C$  is the indicator function of  $C$ . We also assume that  $\mathcal{R}_J$  models the prior distribution associated with the Gaussian denoiser  $J$  with  $\text{prox}_{\mathcal{R}_J} = J$ , since  $\text{prox}_{\mathcal{R}_J}$  corresponds to the MAP estimate under additive white Gaussian noise [20]. Note that  $\mathcal{R}_J$  is introduced for convenience as an explicit regularization term, but the existence of such a function is not guaranteed (see Remark 1).

Then, we can derive a convergent PnP-PDS algorithm for this type of image restoration as a special case of our general PnP-PDS in (9). The algorithm and its convergence property are summarized in the following corollary.

**Corollary 1.** *Let  $J : \mathbb{R}^K \rightarrow \mathbb{R}^K$  be a firmly nonexpansive denoiser and defined over the full domain. Then, there exists a maximally monotone operator  $A_J$  such that  $J = (A_J + \text{Id})^{-1}$ . Assume that the solution set of the following monotone inclusion problem:*

$$\begin{aligned} &\text{Find } (\hat{\mathbf{x}}, \hat{\mathbf{y}}^{(1)}, \hat{\mathbf{y}}^{(2)}) \in \mathbb{R}^{3K} \\ &\text{s.t. } \begin{pmatrix} \mathbf{0} \\ \mathbf{0} \\ \mathbf{0} \end{pmatrix} \in \begin{pmatrix} \gamma_1^{-1} A_J(\hat{\mathbf{x}}) + \Phi^* \hat{\mathbf{y}}^{(1)} + \hat{\mathbf{y}}^{(2)} \\ -\Phi \hat{\mathbf{x}} + \partial \mathcal{D}^*(\hat{\mathbf{y}}^{(1)}) \\ -\hat{\mathbf{x}} + \partial \iota_C^*(\hat{\mathbf{y}}^{(2)}) \end{pmatrix} \end{aligned} \quad (13)$$

is nonempty, and suppose that the following conditions hold for  $\gamma_1 > 0$  and  $\gamma_2 > 0$ :

$$\frac{1}{\gamma_1} - \gamma_2 (\|\Phi\|_{\text{op}}^2 + 1) > 0,$$

and  $\{\rho_n\}$  satisfies the condition in (ii) and (iii) of Proposition 3.2. Then, the sequence  $\{\mathbf{x}_n, \mathbf{y}_n^{(1)}, \mathbf{y}_n^{(2)}\}_{n \in \mathbb{N}}$  generated by the following algorithm:

$$\begin{cases} \tilde{\mathbf{x}}_{n+1} = J\left(\mathbf{x}_n - \gamma_1 \left(\Phi^* \mathbf{y}_n^{(1)} + \mathbf{y}_n^{(2)}\right)\right), \\ \tilde{\mathbf{y}}_{n+1}^{(1)} = \text{prox}_{\gamma_2 \mathcal{D}^*}\left(\mathbf{y}_n^{(1)} + \gamma_2 \Phi(2\tilde{\mathbf{x}}_{n+1} - \mathbf{x}_n)\right), \\ \tilde{\mathbf{y}}_{n+1}^{(2)} = \text{prox}_{\gamma_2 \iota_C^*}\left(\mathbf{y}_n^{(2)} + \gamma_2 (2\tilde{\mathbf{x}}_{n+1} - \mathbf{x}_n)\right), \\ \mathbf{x}_{n+1} = \rho_n \tilde{\mathbf{x}}_{n+1} + (1 - \rho_n) \mathbf{x}_n, \\ \mathbf{y}_{n+1}^{(1)} = \rho_n \tilde{\mathbf{y}}_{n+1}^{(1)} + (1 - \rho_n) \mathbf{y}_n^{(1)}, \\ \mathbf{y}_{n+1}^{(2)} = \rho_n \tilde{\mathbf{y}}_{n+1}^{(2)} + (1 - \rho_n) \mathbf{y}_n^{(2)}, \end{cases} \quad (14)$$

converges to a solution to (13).

This is a direct consequence of Proposition 3.2. Let us define  $f(\mathbf{x})$ ,  $h(\mathbf{y}^{(1)}, \mathbf{y}^{(2)})$  in (10) as follows:

$$\begin{aligned} f(\mathbf{x}) &:= 0, \\ h(\mathbf{y}^{(1)}, \mathbf{y}^{(2)}) &:= \mathcal{D}(\mathbf{y}^{(1)}) + \iota_C(\mathbf{y}^{(2)}), \\ \mathbf{L} &:= (\Phi, \mathbf{I}), \end{aligned}$$

where  $\mathbf{I}$  represents an identity matrix. Then, the monotone inclusion problem in (10) is reduced to (13) [55, Proposition 13.30, Proposition 16.9, Corollary 16.48], and we obtain Algorithm (14) from (9). The condition regarding  $\gamma_1$  and  $\gamma_2$  is obtained from the assumption in (i) of Proposition 3.2 and the following fact [59, Theorem 1]:

$$\|\mathbf{L}\|_{\text{op}}^2 \leq \|\Phi\|_{\text{op}}^2 + \|\mathbf{I}\|_{\text{op}}^2 = \|\Phi\|_{\text{op}}^2 + 1.$$

1) *Image Restoration under Gaussian Noise*: Consider the observation model in (11) with  $\mathcal{N}(\mathbf{x}) := \mathbf{x} + \mathbf{n}$ , where  $\mathbf{n} \in \mathbb{R}^K$  is additive white Gaussian noise. Image restoration under this observation model can be formulated as the following optimization problem:

$$\underset{\hat{\mathbf{u}} \in \mathbb{R}^K}{\text{minimize}} \mathcal{R}_J(\hat{\mathbf{u}}) \quad \text{s.t.} \quad \begin{cases} \|\Phi \hat{\mathbf{u}} - \mathbf{v}\|_2 \leq \varepsilon, \\ \hat{\mathbf{u}} \in [0, 1]^K, \end{cases} \quad (15)$$

where  $\varepsilon > 0$  is the radius of the  $\ell_2$  data-fidelity constraint, which can be determined by the standard deviation of  $\mathbf{n}$ .

To apply the algorithm in (14), we define  $\mathcal{D}(\cdot)$  and  $C$  in (12) as follows:

$$\begin{aligned} \mathcal{D}(\cdot) &:= \iota_{B_{2, \varepsilon}^{\mathbf{v}}}(\cdot), \\ C &:= [0, 1]^K, \end{aligned}$$

where  $B_{2, \varepsilon}^{\mathbf{v}}$  is a  $\mathbf{v}$ -centered  $\ell_2$ -norm ball with the radius  $\varepsilon$ . Then, the problem in (15) can be expressed as the form of (12), allowing the application of (14). If we set  $\rho_n = 1$  for all  $n \in \mathbb{N}$ , we obtain Algorithm 1.

*Remark 2* (Constrained formulation of data-fidelity). In (15), data-fidelity is expressed as a hard constraint using the  $\ell_2$ -norm ball. The benefits of expressing data-fidelity or regularization terms as hard constraints have been illustrated in the literature [60]–[64]. We will discuss such advantages in the specific case of (15) for PnP methods in Section III-C1.

---

**Algorithm 1** Image restoration under Gaussian noise by PnP-PDS.

---

**Input**  $\mathbf{u}_0, \mathbf{w}_0^{(1)}, \mathbf{w}_0^{(2)}, \gamma_1 > 0, \gamma_2 > 0, \varepsilon > 0$   
1: **while** a stopping criterion is not satisfied **do**  
2:    $\mathbf{u}_{n+1} \leftarrow J\left(\mathbf{u}_n - \gamma_1 \left(\Phi^* \mathbf{w}_n^{(1)} + \mathbf{w}_n^{(2)}\right)\right);$   
3:    $\bar{\mathbf{w}}_n^{(1)} \leftarrow \mathbf{w}_n^{(1)} + \gamma_2 \Phi(2\mathbf{u}_{n+1} - \mathbf{u}_n);$   
4:    $\bar{\mathbf{w}}_n^{(2)} \leftarrow \mathbf{w}_n^{(2)} + \gamma_2 (2\mathbf{u}_{n+1} - \mathbf{u}_n);$   
5:    $\mathbf{w}_{n+1}^{(1)} \leftarrow \bar{\mathbf{w}}_n^{(1)} - \gamma_2 P_{B_{2, \varepsilon}^{\mathbf{v}}}(\frac{1}{\gamma_2} \bar{\mathbf{w}}_n^{(1)});$   
6:    $\mathbf{w}_{n+1}^{(2)} \leftarrow \bar{\mathbf{w}}_n^{(2)} - \gamma_2 P_{[0, 1]^K}(\frac{1}{\gamma_2} \bar{\mathbf{w}}_n^{(2)});$   
7: **end while**  
**Output**  $\mathbf{u}_n$

---



---

**Algorithm 2** Image restoration under Poisson noise by PnP-PDS.

---

**Input**  $\mathbf{u}_0, \mathbf{w}_0^{(1)}, \mathbf{w}_0^{(2)}, \gamma_1 > 0, \gamma_2 > 0, \lambda > 0$   
1: **while** a stopping criterion is not satisfied **do**  
2:    $\mathbf{u}_{n+1} \leftarrow J\left(\mathbf{u}_n - \gamma_1 \left(\Phi^* \mathbf{w}_n^{(1)} + \mathbf{w}_n^{(2)}\right)\right);$   
3:    $\bar{\mathbf{w}}_n^{(1)} \leftarrow \mathbf{w}_n^{(1)} + \gamma_2 \Phi(2\mathbf{u}_{n+1} - \mathbf{u}_n);$   
4:    $\bar{\mathbf{w}}_n^{(2)} \leftarrow \mathbf{w}_n^{(2)} + \gamma_2 (2\mathbf{u}_{n+1} - \mathbf{u}_n);$   
5:    $\mathbf{w}_{n+1}^{(1)} \leftarrow \bar{\mathbf{w}}_n^{(1)} - \gamma_2 \text{prox}_{\frac{\lambda}{\gamma_2} \text{GKL}}(\frac{1}{\gamma_2} \bar{\mathbf{w}}_n^{(1)});$   
6:    $\mathbf{w}_{n+1}^{(2)} \leftarrow \bar{\mathbf{w}}_n^{(2)} - \gamma_2 P_{[0, 1]^K}(\frac{1}{\gamma_2} \bar{\mathbf{w}}_n^{(2)});$   
7: **end while**  
**Output**  $\mathbf{u}_n$

---

2) *Image Restoration under Poisson Noise*: Image restoration under Poisson noise has been studied across various domains, including medical imaging, astronomical imaging, and remote sensing. The observation process considering Poisson noise contamination can be modeled by (11) when  $\mathcal{N}(\mathbf{x}) := \mathcal{P}_\eta(\mathbf{x})$ , where  $\mathcal{P}_\eta$  represents the corruption by Poisson noise with the scaling coefficient  $\eta$ .

In this observation model, the data-fidelity is expressed as the GKL divergence (refer to [65]), and image restoration can be reduced to the following optimization problem:

$$\underset{\hat{\mathbf{u}} \in \mathbb{R}^K}{\text{minimize}} \lambda \text{GKL}_{\mathbf{v}}(\Phi \hat{\mathbf{u}}) + \mathcal{R}_J(\hat{\mathbf{u}}) \quad \text{s.t.} \quad \hat{\mathbf{u}} \in [0, 1]^K \quad (16)$$

where  $\lambda > 0$  is a parameter that determines the balance between data-fidelity and regularization (see (1) for the definition of GKL).

Let us define  $\mathcal{D}(\cdot)$  and  $C$  in (12) as follows:

$$\begin{aligned} \mathcal{D}(\cdot) &:= \lambda \text{GKL}_{\mathbf{v}}(\cdot), \\ C &:= [0, 1]^K. \end{aligned}$$

Then, the problem in (16) is reformulated as (12), and the iterative procedure in (14) becomes applicable. Setting  $\rho_n = 1$  for all  $n \in \mathbb{N}$ , the resulting algorithm is given as Algorithm 2.

### C. Comparison with Prior Works

1) *Convergent PnP-FBS [18]*: Pesquet et al. proposed a PnP-FBS algorithm using DnCNN as a convergent PnP approach [18]. This method can be applied to image restoration

under Gaussian noise, addressing the following optimization problem:

$$\underset{\hat{\mathbf{u}} \in \mathbb{R}^K}{\text{minimize}} \quad \frac{\lambda}{2} \|\Phi \hat{\mathbf{u}} - \mathbf{v}\|_2^2 + \mathcal{R}_J(\hat{\mathbf{u}}), \quad (17)$$

where the first term is an  $\ell_2$  data-fidelity term, and  $\lambda > 0$  is a balancing parameter.

This convergent PnP-FBS is a landmark method that first ensures the convergence of PnP-FBS with the characterization of the solution set under realistic assumptions. At the same time, it should be noted that the problem in (15), where the data-fidelity term is represented as a hard constraint, cannot be solved by PnP-FBS.

Consequently, the proposed PnP-PDS (Algorithm 1) inherits the versatility of a once-trained denoiser in this setting, compared to PnP-FBS. This advantage arises from its high flexibility in parameter selection. In the case of the constrained formulation in (15), a natural choice for  $\varepsilon$  is given as follows [61]:

$$\varepsilon = \sigma\sqrt{K}, \quad (18)$$

where  $\sigma$  denotes the standard deviation of the Gaussian noise in the observed images. We remark that the convergence of the proposed PnP-PDS is guaranteed regardless of the choice of  $\varepsilon$ .

In contrast, PnP-FBS limits the flexibility of the denoiser in certain cases, due to constraints on the parameter to ensure convergence. Assume that the denoiser is trained with a noise level of  $\sigma_J$ , and that  $\Phi$  represents a circular convolution with kernel  $h$ . In the additive formulation in (17), the authors suggest the following value for  $\lambda$  [18]:

$$\lambda_{\text{opt}} := \frac{\sigma_J}{2\sigma\|h\|}. \quad (19)$$

However,  $\lambda$  must also satisfy the condition to ensure the convergence of PnP-FBS:

$$\lambda < \frac{2}{\|\Phi\|_{\text{op}}^2}. \quad (20)$$

Hence, if  $\lambda_{\text{opt}}$  does not satisfy the constraint in (20), the convergence of PnP-FBS cannot be ensured with its optimal parameter. This limitation was observed in our experiments (see Section IV-B1).

Another limitation of PnP-FBS is its relatively narrow applicability to problem formulations. Specifically, it does not handle additional constraints explicitly, such as the box constraint in (15), and cannot be applied to the Poisson noise setting in (16), which is formulated with the generalized Kullback–Leibler (GKL) divergence.

2) *Convergent PnP-ADMM* [24], [25]: Several works have proposed PnP algorithms based on ADMM (PnP-ADMM) and established their convergence guarantees. Nair et al. have proposed a convergent PnP-ADMM and provided extensive analysis on its fixed-point [24]. Sun et al. have proposed a convergent PnP-ADMM algorithm that handles multiple data-fidelity terms, making it practical for large scale datasets [25]. In both studies, the convergence properties are ensured under the firm nonexpansiveness of the denoiser. PnP-ADMM can

address a wider range of tasks than PnP-FBS, including image restoration with Poisson noise.

The main advantage of the proposed PnP-PDS over PnP-ADMM is the elimination of inner iterations. To solve a subproblem in PnP-ADMM algorithms, it is often necessary to perform inner iterations, which results in increasing computational cost and unstable numerical convergence. On the other hand, the proposed PnP-PDS does not need to perform them thanks to the fact that the computations associated with the linear operator  $L$  are fully decoupled from the proximity operator of  $h^*$  in the algorithm.

3) *Convergent PnP-PDS* [27]: Garcia et al. proposed a convergent PDS-based PnP algorithm for restoration of images obtained by a photonic lantern [27], published contemporaneously with our conference paper [54]. They consider the observation process  $\mathbf{v} = \Phi \mathbf{u} + \mathbf{n}$ , where  $\mathbf{v}$  is an observed image,  $\mathbf{u}$  is an original image,  $\Phi$  is a linear degradation process, and  $\mathbf{n} \in \mathbb{R}^K$  is an additive noise with an unknown distribution, which is assumed to satisfy  $\|\mathbf{n}\|_2 < \varepsilon$  for  $\varepsilon > 0$ . Based on this observation process, they address the following optimization problem by a convergent PnP-PDS:

$$\underset{\hat{\mathbf{u}} \in \mathbb{R}^K}{\text{minimize}} \quad \mathcal{R}_J(\hat{\mathbf{u}}) \quad \text{s.t.} \quad \|\Phi \hat{\mathbf{u}} - \mathbf{v}\|_2 \leq \varepsilon.$$

We show the relationship between this algorithm and our PnP-PDS algorithm in Fig. 1. In essence, the algorithm in [27] is a special case of our Algorithm 1 without the box constraint.

We note that introducing a box constraint ensures that the solution to the optimization (or monotone inclusion) problem remains within a certain convex set, which in turn promotes more stable behavior of PnP iterations for some experimental settings. The experimental results in Section IV illustrate this effect.

4) *Regularization by Denoising (RED)* [45], [52]: Romano et al. proposed a framework called RED, which defines an explicit objective function by incorporating the output of a denoiser. For image restoration under Gaussian noise and image restoration under Poisson noise, the optimization problems with RED are given respectively as

$$\underset{\hat{\mathbf{u}} \in \mathbb{R}^K}{\text{minimize}} \quad \lambda \|\Phi \hat{\mathbf{u}} - \mathbf{v}\|_2^2 + \hat{\mathbf{u}}^\top (\hat{\mathbf{u}} - \mathfrak{G}(\hat{\mathbf{u}})), \quad (21)$$

$$\underset{\hat{\mathbf{u}} \in \mathbb{R}^K}{\text{minimize}} \quad \lambda \text{GKL}_{\mathbf{v}}(\hat{\mathbf{u}}) + \hat{\mathbf{u}}^\top (\hat{\mathbf{u}} - \mathfrak{G}(\hat{\mathbf{u}})), \quad (22)$$

where  $\mathfrak{G}$  represents a Gaussian denoiser. In RED, the optimization algorithms are constructed by computing the gradient of the second terms in (21) and (22) using the following expression:

$$\nabla(\hat{\mathbf{u}}^\top (\hat{\mathbf{u}} - \mathfrak{G}(\hat{\mathbf{u}}))) = 2(\hat{\mathbf{u}} - \mathfrak{G}(\hat{\mathbf{u}})). \quad (23)$$

For differentiable  $\mathfrak{G}$ , this equation holds if and only if  $\mathfrak{G}$  is locally homogeneous and has a symmetric Jacobian [45], [52]. For non-differentiable  $\mathfrak{G}$ , a more recent study [46] demonstrates that (23) still holds under the assumption that  $\text{Id} - \mathfrak{G}$  is *maximally cyclically monotone*, or that  $\mathfrak{G}$  is *cyclically firmly nonexpansive*. Furthermore, if  $\mathfrak{G}$  is also nonexpansive in addition to either of these conditions, we can establish a convergent algorithm based on gradient computations, such as gradient descent or FBS.

TABLE II: Summary of Compared Methods, Their Optimization Algorithms, and Regularization.

Method	Algorithm	Regularization
PnP-FBS	FBS	FNE* DnCNN
PnP-ADMM	ADMM	FNE* DnCNN
PnP-PDS (Unstable)	PDS	DnCNN (not FNE*)
PnP-PDS (NoBox)	PDS	FNE* DnCNN
TV	PDS	TV
RED (Gaussian)	SD <sup>†</sup>	FNE* DnCNN
RED (Poisson)	PDS	FNE* DnCNN
DRUNet	HQS <sup>‡</sup>	DRUNet
BPG	BPG	Bregman Score Denoiser
Proposed	PDS	FNE* DnCNN

\*FNE: firmly nonexpansive

†SD: steepest descent

‡HQS: half-quadratic splitting

The main advantage of using RED is the simplicity resulting from the existence of a clear convex objective function. In contrast, PnP algorithms including our PnP-PDS do not necessarily have a convex objective function. Instead, they solve a monotone inclusion problem expressed as (10).

However, RED also has a considerable difficulty: the assumptions for the denoiser. First, local homogeneity is an unrealistic property, especially for deep denoisers. In fact, the gradients of the regularization terms computed in the RED algorithms often contain substantial computational errors due to the lack of local homogeneity in the denoisers [52]. Second, *cyclically firm nonexpansiveness* [46] is even stronger assumption than the firm nonexpansiveness assumed in our work. These challenges make it difficult to ensure the convergence of RED algorithms when we employ state-of-the-art denoisers, including DnCNN.

#### IV. EXPERIMENTS

We performed two types of experiments to demonstrate the stability, performance, and versatility of the proposed method. The first experiment involves deblurring/inpainting under Gaussian noise, and the second experiment involves deblurring/inpainting under Poisson noise. The purpose of these experiments is to confirm the following two facts:

- The proposed method operates stably and converges in all experimental settings.
- In comparison to other state-of-the-art methods, the proposed method demonstrates higher restoration performance thanks to its stability.

##### A. Experimental Setup

In the two experiments, we considered two types of observation operators: blur and random sampling. For the blur case, we used circular convolution with the ten kernels (a)–(j) from [66], [67], visualized in Fig. 2, and we scaled them so that  $\|\Phi\|_{\text{op}} = 1$ . In Sec. IV-B-1 we evaluated all ten kernels (a)–(j), whereas in the other sections we used kernel (a) as a representative case. For the random sampling case, 20% of the image pixels were randomly selected and masked. The operator norm also satisfies  $\|\Phi\|_{\text{op}} = 1$  in this case.

We evaluated the restoration performance by Peak Signal to Noise Ratio (PSNR) and Structural Similarity Index Measure (SSIM). To investigate stability, the update rate  $c_n$  for  $n \in \mathbb{N} \setminus \{0\}$  was defined by  $c_n := \|\mathbf{u}_n - \mathbf{u}_{n-1}\|_2 / \|\mathbf{u}_{n-1}\|_2$ , where  $\mathbf{u}_n$  is the intermediate solution at the  $n$ -th iteration by each algorithm.

We used color images for Gaussian noise case and grayscale images for Poisson noise case. The color images were randomly sampled from ImageNet [68], consisting of 7 images cropped to  $128 \times 128$  pixels. The grayscale images were selected from 3 widely used image sets [18], all with a size of  $256 \times 256$  pixels.

In each experiment, we selected the appropriate methods from PnP-FBS [18], PnP-ADMM [25], PnP-PDS (Unstable) [26], PnP-PDS (NoBox) [27], TV [62], RED [45], DRUNet [38], and Bregman Proximal Gradient (BPG) [51], and compared them with the proposed method. The optimization algorithms and regularization for each method are summarized in Table II. Details of PnP-FBS, PnP-ADMM, and RED are given in Section III-C. PnP-PDS (NoBox) corresponds to the proposed method without the box constraint, with all other settings aligned. We adopted the stopping criterion proposed by the authors for BPG. For the other methods, a fixed iteration count was used. Specifically, in the Poisson noise case with  $\eta \in \{1, 2, 10\}$ , the number of iterations was fixed to 4800 for deblurring and 12000 for inpainting. In the other cases, the number of iterations was fixed at 1200 for the blur case and 3000 for the random sampling case. For DRUNet, we employed the half-quadratic splitting algorithm, and used the same fixed iteration counts as the other methods. In addition, we also evaluate DRUNet (8 iter), where DRUNet is executed with 8 iterations of the HQS algorithm, to examine its performance in the low-iteration regime commonly used in practice.

For all settings, we used the DnCNN [18] trained with the noise level  $\sigma_J$ , for the regularization in PnP-FBS, PnP-ADMM, PnP-PDS (NoBox), RED, and the proposed method. This DnCNN is learned as described in Section II-D to promote its firm nonexpansiveness. For a fair comparison with PnP-FBS, whose the optimal parameter depends on  $\sigma_J$ , we adopted the model distributed by Pesquet et al., trained with  $\sigma_J = 0.01$  in Section IV-B, and used the same setting in Section IV-C. In Section IV-D, we investigate the influence of  $\sigma_J$  by examining two cases: (i)  $\sigma_J$  fixed to one of  $\{0.0075, 0.01, 0.05, 0.1\}$  and (ii)  $\sigma_J$  randomly sampled from a uniform distribution over  $[0, 0.1]$  for each training sample.

Throughout our experiments, the parameters  $\gamma_1$  and  $\gamma_2$  in Algorithm 1 and Algorithm 2 were set to 0.5 and 0.99, respectively, satisfying the condition in Proposition 3.2. All of our experiments were conducted on a Windows 11 PC, equipped with an Intel Core i9 3.70GHz processor and 32.0GB of RAM.

##### B. Deblurring/Inpainting under Gaussian Noise

We performed experiments on deblurring/inpainting under Gaussian noise. Here, we investigate the versatility of a once-trained denoiser in the proposed method, and compare its performance with existing methods.

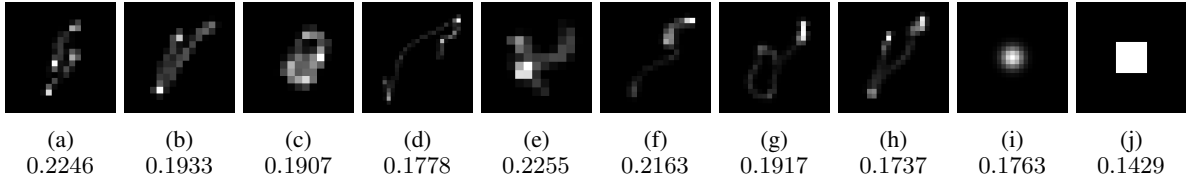


Fig. 2: Blur kernels used in our experiments. (a)–(h) correspond to kernels (1)–(8) in [66], while (i) and (j) correspond to the ‘GaussianA’ and ‘Square’ setup in [67], respectively. The Frobenius norm of each kernel is shown in the second row.

TABLE III: PSNR [dB] Values for 10 Blur Kernels Across Different Noise Levels. The Symbol ‘—’ Denotes That the Algorithm Diverged and PSNR Values Could not be Obtained. The Better Results are Highlighted in Bold.

Noise level	Method	(a)	(b)	(c)	(d)	(e)	(f)	(g)	(h)	(i)	(j)	Average
$\sigma = 0.0025$	PnP-FBS	31.28*	30.42*	30.28*	30.11*	31.76*	—	30.14*	30.02*	28.15*	27.75*	—
	Proposed	<b>37.27</b>	<b>36.63</b>	<b>35.68</b>	<b>36.64</b>	<b>36.69</b>	<b>37.49</b>	<b>35.43</b>	<b>35.58</b>	<b>29.70</b>	<b>30.85</b>	<b>35.20</b>
$\sigma = 0.005$	PnP-FBS	31.16*	30.35*	30.22*	30.04*	31.67*	—	30.07*	29.95*	28.12*	27.71*	—
	Proposed	<b>34.04</b>	<b>33.46</b>	<b>32.95</b>	<b>33.37</b>	<b>34.01</b>	<b>34.46</b>	<b>32.71</b>	<b>32.70</b>	<b>28.87</b>	<b>29.33</b>	<b>32.59</b>
$\sigma = 0.01$	PnP-FBS	30.58	29.98*	29.95*	29.69*	31.24	31.26	29.76*	29.60*	28.00*	27.56*	29.76
	Proposed	<b>30.97</b>	<b>30.52</b>	<b>30.50</b>	<b>30.37</b>	<b>31.55</b>	<b>31.74</b>	<b>30.20</b>	<b>30.09</b>	<b>28.16</b>	<b>27.85</b>	<b>30.20</b>
$\sigma = 0.02$	PnP-FBS	<b>28.26</b>	<b>27.92</b>	<b>28.35</b>	<b>27.66</b>	29.34	29.27	<b>28.18</b>	<b>27.94</b>	<b>27.43</b>	<b>26.80</b>	<b>28.12</b>
	Proposed	28.15	27.70	28.14	27.41	<b>29.36</b>	<b>29.28</b>	28.15	27.73	27.07	26.39	27.94
$\sigma = 0.04$	PnP-FBS	25.55	24.96	25.99	24.34	26.98	26.43	25.79	25.09	<b>26.41</b>	25.18	25.67
	Proposed	<b>26.00</b>	<b>25.73</b>	<b>26.29</b>	<b>25.58</b>	<b>27.05</b>	<b>26.80</b>	<b>26.13</b>	<b>25.94</b>	26.18	<b>25.42</b>	<b>26.11</b>

\* As the algorithm did not converge for  $\lambda$  values in the range  $0.8\lambda_{\text{opt}}$  to  $1.25\lambda_{\text{opt}}$ , we present the results for  $\lambda = 1.99$ .

1) *Versatility of a Once-Trained Denoiser*: We investigate the versatility of the once-trained denoiser with the fixed value of  $\sigma_J$  in different algorithms. Since the theoretical convergence guarantees in this work are built upon PnP-FBS, let us confirm that the versatility is enhanced by extending PnP-FBS to PnP-PDS.

As discussed in Section III-C1, the proposed method handles  $\ell_2$  data-fidelity constraint, whereas PnP-FBS handles  $\ell_2$  data-fidelity term. Following (18), we set  $\varepsilon$  in (15) as  $\varepsilon = \alpha\varepsilon_{\text{opt}}$  where  $\varepsilon_{\text{opt}} := \sigma\sqrt{K}$  and evaluated the performance over  $\alpha \in [0.8, 1.25]$ . For PnP-FBS, we followed the heuristic in (19) and set  $\lambda = \zeta\lambda_{\text{opt}}$  where  $\lambda_{\text{opt}} := \sigma_J/(2\sigma\|h\|)$  and  $\|h\|$  is the Frobenius norm of the blur kernel  $h$ . We evaluated the performance over  $\zeta \in [0.8, 1.25]$ . When the algorithm diverged within this range, we instead used  $\lambda = 1.99$ .

Table III shows the variation in restoration performance across different noise levels and blur kernels. At low noise levels and for kernels with small Frobenius norms, PnP-FBS diverges with  $\lambda$  close to  $\lambda_{\text{opt}}$  as discussed in Section III-C1. For example, in the case of  $\sigma = 0.01$  and kernel (b), we obtain  $\lambda_{\text{opt}} \approx 2.587$  because  $\sigma_J = 0.01$  and  $2\sigma\|h\| \approx 0.0039$ . However, the convergence condition (20) requires  $\lambda < 2$ , forcing the use of suboptimal parameter choices. This discrepancy stems from the dependence of  $\lambda_{\text{opt}}$  on  $\sigma_J$ .

In contrast, the proposed method exhibits reasonable performance across all noise levels. This consistency demonstrates the high versatility of the given denoiser, thanks to the decoupling of  $\varepsilon_{\text{opt}}$  and  $\sigma_J$  (see also the discussion in Section III-C1). Moreover, we note that  $\varepsilon_{\text{opt}}$  is independent of the blur kernels, which enables stable performance across all kernels under a consistent parameter choice.

TABLE IV: The Values of  $\alpha$  Used in Each Setting for the Proposed Method.

Noise level $\sigma$	0.0025	0.005	0.01	0.02	0.04
Deblurring	0.82	0.86	0.92	0.96	1.00
Inpainting	0.90	0.82	0.82	1.00	1.00

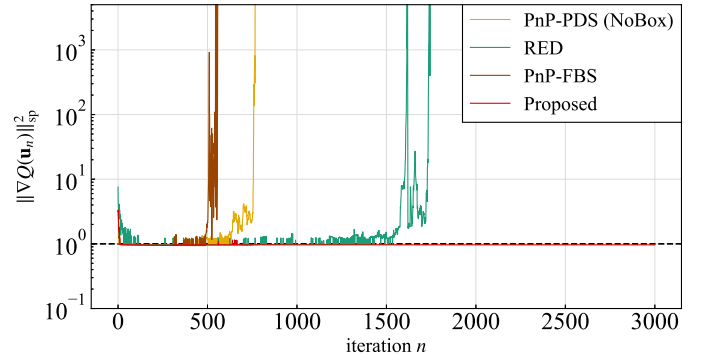


Fig. 3: The evolution of  $\|\nabla Q(\mathbf{u}_n)\|_{\text{sp}}^2$  of PnP-PDS (NoBox), PnP-FBS, RED, and the proposed method for inpainting at  $\sigma = 0.01$ . The values in each plot are computed as the mean over 7 images.

2) *Restoration Performance*: We compare the performance of the proposed method with other methods, namely PnP-FBS [18], PnP-PDS (Unstable) [26], PnP-PDS (NoBox) [27], TV [62], RED [45], and DRUNet [38]. For constrained formulations such as PnP-PDS (Unstable), PnP-PDS (NoBox), TV, and the proposed method, we set  $\varepsilon = \alpha\varepsilon_{\text{opt}}$  in the same way as in the previous section. In particular, we provide the values of  $\alpha$

TABLE V: The Average PSNR [dB] Values for Deblurring and Inpainting on the 7 Images From ImageNet. The Best Value is Highlighted in Bold and the Second Best Value is Underlined.

Noise level $\sigma$	Deblurring					Inpainting				
	0.0025	0.005	0.01	0.02	0.04	0.0025	0.005	0.01	0.02	0.04
PnP-FBS [18]	31.28	31.16	30.59	28.19	23.95	--	--	--	--	--
PnP-PDS (Unstable) [26]	23.79	23.67	23.78	24.60	23.35	33.65	33.36	32.79	31.51	29.58
PnP-PDS (NoBox) [27]	<u>37.23</u>	34.03	31.00	28.24	<b>26.00</b>	--	--	--	--	--
TV [62]	31.97	30.40	28.78	27.10	25.38	<u>34.19</u>	<u>33.90</u>	<u>33.22</u>	<u>31.90</u>	<u>29.85</u>
RED [45]	34.11	33.38	30.89	<b>28.30</b>	24.87	--	--	--	--	--
DRUNet [38]	32.60	31.80	28.54	--	--	33.21	32.68	31.36	26.71	--
DRUNet (8 iter) [38]	35.33	<b>34.50</b>	<b>31.74</b>	21.65	11.38	13.84	13.84	13.82	13.77	13.57
PnP-PDS (Proposed)	<b>37.24</b>	<u>34.04</u>	<u>31.01</u>	<u>28.26</u>	<b>26.00</b>	<b>35.61</b>	<b>35.25</b>	<b>34.43</b>	<b>33.04</b>	<b>31.00</b>

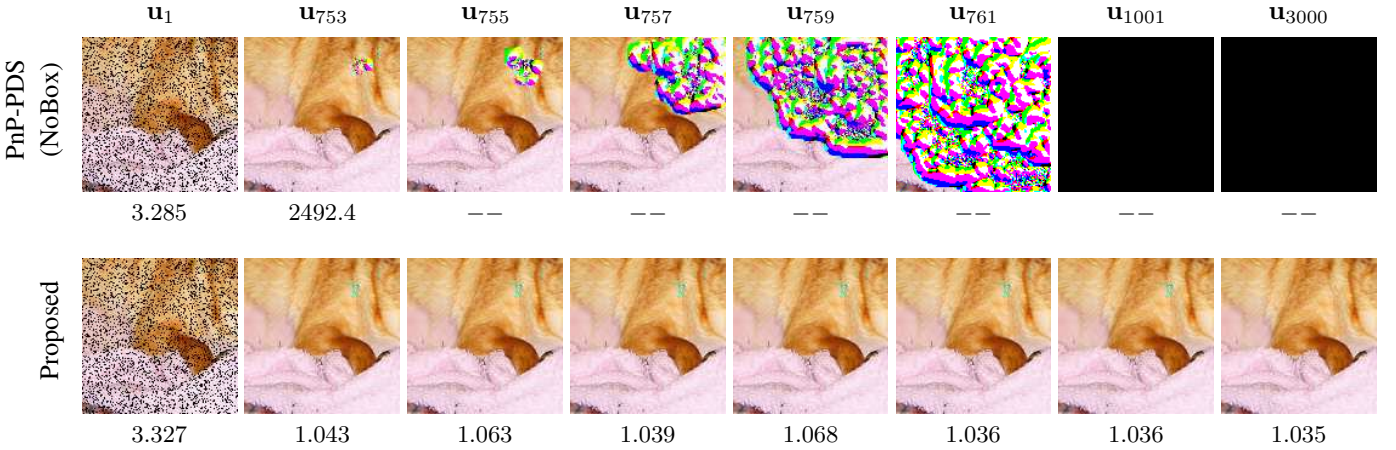


Fig. 4: Visualization of  $\mathbf{u}_n$  of PnP-PDS (NoBox) and the proposed method, along with the corresponding values of  $\|\nabla Q(\mathbf{u}_n)\|_{\text{sp}}^2$  for each  $\mathbf{u}_n$ . For visualization, each pixel value is clipped to the range  $[0, 1]$ . The symbol “--” indicates that the value of  $\|\nabla Q(\mathbf{u}_n)\|_{\text{sp}}^2$  could not be computed. Since  $\mathbf{u}_{1001}$  and  $\mathbf{u}_{3000}$  of PnP-PDS (NoBox) contained NaN values in Python, they could not be visualized.

for the proposed method in Table IV. For existing methods, we adopted the recommended values if parameter settings were suggested by the authors. Otherwise, we performed a linear search over the balancing parameter.

Table V summarizes the average PSNR values over seven images, obtained with the optimal parameters for each method. The proposed method demonstrates superior performance compared to PnP-FBS, even though both methods employ the same denoiser for regularization. This is attributed to the difference in the versatility of the denoiser, as mentioned in Section IV-B1. Moreover, we observe that DRUNet (8 iter) achieves higher PSNR values than the proposed method in deblurring at  $\sigma = 0.005$  and  $\sigma = 0.01$ . However, the proposed method outperforms DRUNet for all cases in larger iterations because of the difference in stability of the algorithms, which we further examine later with their convergence curves. This instability can be attributed to the lack of a theoretical convergence guarantee, and this observation is consistent with the literature, for instance in [69].

In Table V, we also note that PnP-FBS, PnP-PDS (NoBox), and RED fail to converge, and thus PSNR values cannot be obtained for the inpainting. To clarify this point, recall that the denoiser was trained with the penalty in (7), which was

designed to promote nonexpansiveness for low-noise images sampled according to (8). During the PnP iterations, however, the inputs to the denoiser can become highly noisy and may have characteristics that differ from those of the training images. In such cases, the denoiser may no longer maintain its firm nonexpansiveness, potentially amplifying instability through successive iterations.

For numerical verification, we present the evolution of  $\|\nabla Q(\mathbf{u}_n)\|_{\text{sp}}^2$  at each iteration in Fig. 3. Although PnP-PDS (NoBox), PnP-FBS, RED, and the proposed method employ the same denoiser, the values of  $\|\nabla Q(\mathbf{u}_n)\|_{\text{sp}}^2$  in PnP-PDS (NoBox), PnP-FBS, and RED exhibit unstable behavior and tend to diverge. In contrast, the proposed method consistently maintains values of  $\|\nabla Q(\mathbf{u}_n)\|_{\text{sp}}^2$  that satisfy the condition given in (6). In Fig. 4, we further compare the visualization of  $\mathbf{u}_n$  obtained by PnP-PDS (NoBox) in a diverging case with that of the proposed method. For PnP-PDS (NoBox), erratic regions appear in the image, and the surrounding areas are affected in the proceeding iterations, leading to divergence of  $\mathbf{u}_n$ . Although similar irregular regions are observed at  $\mathbf{u}_{753}$  in the proposed method,  $\mathbf{u}_n$  remains stable and those regions are effectively removed in the subsequent iterations.

To study the impact of enforcing firm nonexpansiveness on

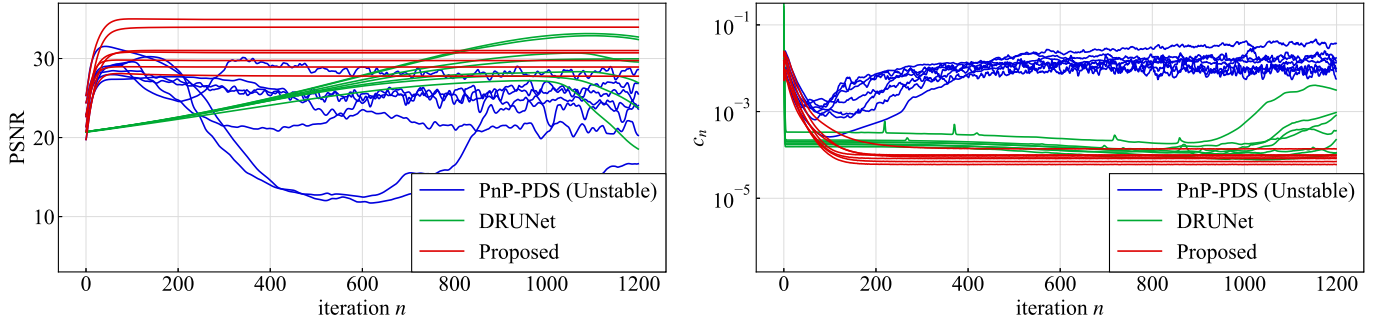


Fig. 5: The evolution of PSNR [dB] and  $c_n$  of PnP-PDS (Unstable), DRUNet, and the proposed method over iterations for deblurring at  $\sigma = 0.01$ . Each plot in the graph corresponds to each image extracted from ImageNet [68].

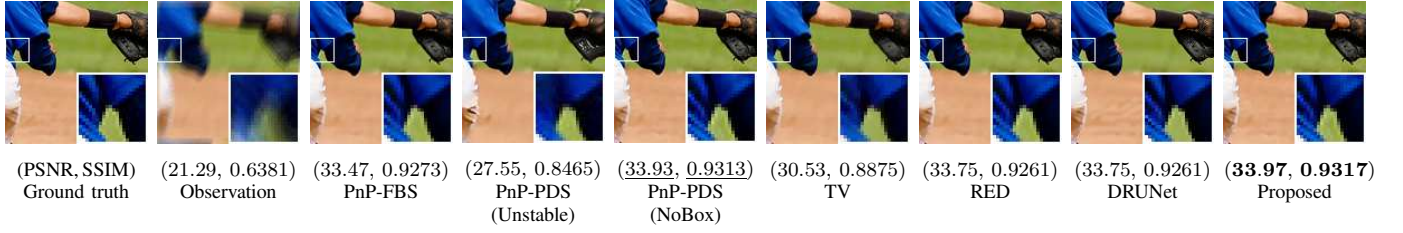


Fig. 6: Visual results of image restoration on ImageNet [68] for deblurring at  $\sigma = 0.01$ . The best results are indicated in bold, and the second-best results are underlined.

the denoiser, we compare the stability of the proposed method with that of PnP-PDS (Unstable) and DRUNet, whose denoisers are not guaranteed to be firmly nonexpansive. Fig. 5 shows the evolution of PSNR and the update rate  $c_n$  for the deblurring task with  $\sigma = 0.01$ . PnP-PDS (Unstable) exhibits unstable behavior throughout the iterations, and DRUNet shows similar behavior, becoming unstable in the later iterations. In contrast, the proposed method converges stably, with  $c_n$  decreasing to approximately  $10^{-5}$ .

In Fig. 6, we provide the visual results for deblurring at  $\sigma = 0.01$ , accompanied by PSNR (left) and SSIM (right). The images obtained by PnP-FBS and TV are overly-smoothed, resulting in a loss of fine detail. For the PnP methods without theoretical convergence guarantees, namely PnP-PDS (Unstable) and DRUNet, strong artifacts are generated because of the unstable behavior of the algorithm (see for example the area above the glove and the texture on the ground in the image, respectively). While RED achieves high PSNR and SSIM, there are some areas where the brightness appears unnaturally altered compared to the original image. The proposed method achieves the highest PSNR and SSIM for this image, presenting the most natural appearance.

### C. Deblurring/Inpainting under Poisson Noise

Let us now present the experiments on deblurring/inpainting under Poisson noise. To vary the noise intensity, we considered six different values for the scaling coefficients  $\eta$  between 1 and 200. Smaller values of  $\eta$  correspond to stronger noise, while larger values indicate weaker noise. We compare the performance of the proposed method with other methods applicable to this scenario, specifically PnP-ADMM [25], PnP-PDS (Unstable) [26], PnP-PDS (NoBox) [27], RED [45],

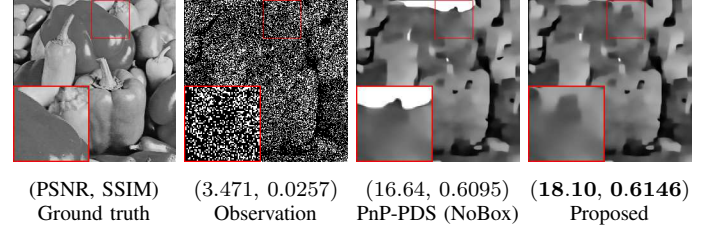


Fig. 7: Visual results of PnP-PDS (NoBox) and the proposed method for inpainting with  $\eta = 1$ . The higher PSNR and SSIM values are highlighted in bold.

and BPG [51]. For methods whose parameter settings are suggested by the authors, we followed their recommendations. For the others, we performed a linear search over the balancing parameters between the data-fidelity and regularization terms. In particular, we report the best values of  $\lambda$  for the proposed method in Table VI.

The average PSNR values over three images are presented in Table VII. The proposed method shows high restoration performance across all settings and outperforms the other methods with higher PSNR values. Especially, we observe that, at high noise levels (i.e.,  $\eta \in \{1, 2\}$ ), the proposed method outperforms PnP-PDS (NoBox). This indicates that the box constraint not only enhances stability as shown in Section IV-B, but also contributes positively to performance under severely degraded conditions. In Fig. 7, we compare the visual results in such cases. It can be observed that PnP-PDS (NoBox) tends to generate unpleasant artifacts when the noise level becomes high, whereas the proposed method effectively suppresses such artifacts.

TABLE VI: The Values of  $\lambda$  Used in Each Setting for the Proposed Method.

Scaling coefficient $\eta$	1	2	10	50	100	200
Deblurring	$2.00 \times 10^{-3}$	$2.00 \times 10^{-3}$	$1.50 \times 10^{-3}$	$1.25 \times 10^{-3}$	$1.25 \times 10^{-3}$	$1.00 \times 10^{-3}$
Inpainting	$1.25 \times 10^{-3}$	$1.25 \times 10^{-3}$	$1.00 \times 10^{-3}$	$7.50 \times 10^{-4}$	$5.00 \times 10^{-4}$	$5.00 \times 10^{-4}$

TABLE VII: The Average PSNR [dB] Values for Deblurring and Inpainting on the Three Images From Set3 [18]. The Best Value is Highlighted in Bold, and the Second Best Value is Underlined.

Scaling coefficient $\eta$	Deblurring						Inpainting					
	1	2	10	50	100	200	1	2	10	50	100	200
PnP-ADMM [25]	10.00	7.58	<b>22.95</b>	25.16	26.28	26.55	7.74	9.04	25.83	11.45	27.87	28.10
PnP-PDS (Unstable) [26]	7.07	13.86	21.06	23.98	25.18	26.51	5.38	8.25	20.29	<b>29.75</b>	<b>31.90</b>	<b>33.43</b>
PnP-PDS (NoBox) [27]	<u>18.85</u>	20.08	22.85	<b>25.17</b>	<u>26.42</u>	<b>27.77</b>	<u>18.09</u>	20.47	<u>25.95</u>	<b>29.75</b>	<u>31.21</u>	<u>32.74</u>
BPG [51]	5.41	6.86	16.27	21.76	26.36	27.45	6.52	7.85	10.19	11.84	12.06	12.18
RED [45]	18.64	<u>20.12</u>	<u>22.91</u>	25.13	26.39	27.72	18.02	<u>20.75</u>	25.88	29.71	31.19	32.69
PnP-PDS (Proposed)	<b>19.04</b>	<b>20.19</b>	22.87	<b>25.17</b>	<b>26.44</b>	<b>27.77</b>	<b>18.59</b>	<b>21.16</b>	<b>25.96</b>	<b>29.75</b>	<u>31.21</u>	<u>32.74</u>

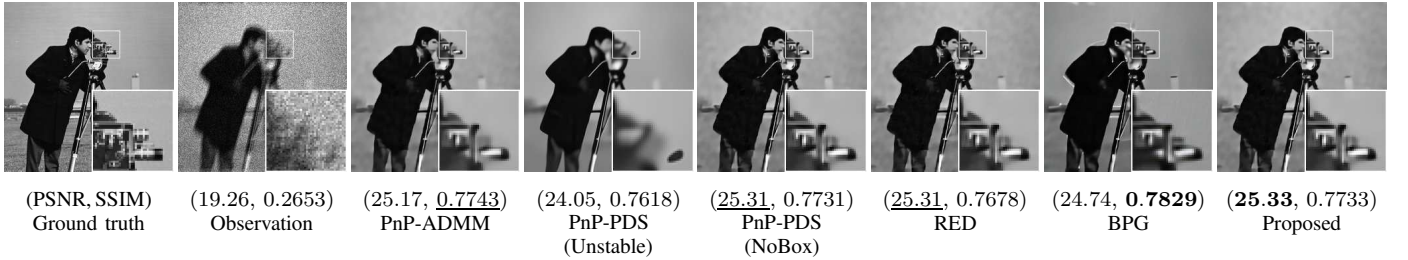
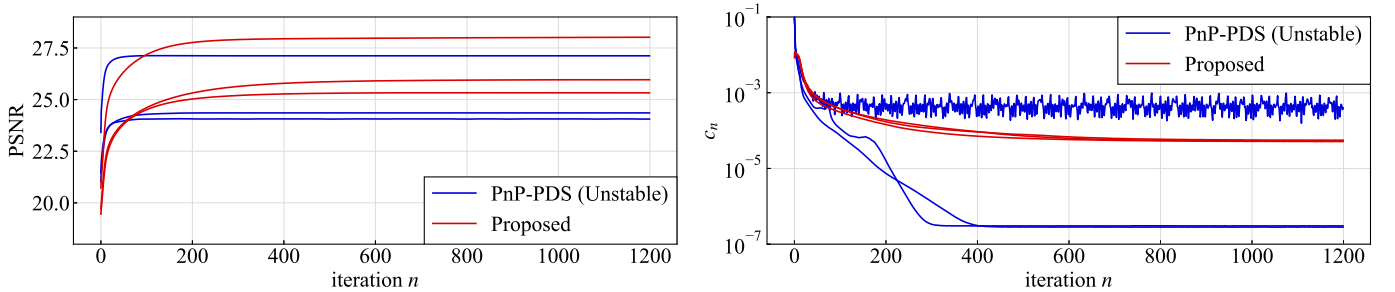
Fig. 8: Visual results of deblurring with  $\eta = 100$ . The best results are indicated in bold, and the second-best results are underlined.Fig. 9: The evolution of PSNR [dB] and  $c_n$  of PnP-PDS (Unstable) and the proposed method over iterations for deblurring at  $\eta = 100$ . Each plot in the graph corresponds to each image extracted from Set3 [18].

TABLE VIII: The Average CPU Time [s] per Iteration Across Scaling Coefficients on the Three Images From Set3 [18]. The Better Result is Highlighted in Bold.

	Deblurring	Inpainting
PnP-ADMM [25]	1.525	0.5558
PnP-PDS (Proposed)	<b>0.5972</b>	<b>0.3153</b>

*Remark 3* (Comparison with Clamping of Denoiser Output). An alternative approach to enforcing the box constraint is to clamp the denoiser output to  $[0, 1]$  after each iteration. This simple post-processing step can help stabilize the algorithm, and using it instead of the box constraint achieves almost the same performance as the proposed method in all set-

tings. On the other hand, the proposed formulation explicitly incorporates the box constraint into the monotone inclusion formulation. This allows for a clearer characterization of the optimality condition and improves the interpretability of the solution.

We investigate the computational efficiency of the proposed method, particularly in comparison with PnP-ADMM. Table VIII provides the CPU times per iteration for PnP-ADMM and the proposed method. As discussed in Section III-C2, we need to perform inner iterations for PnP-ADMM, leading to relatively high computational costs. The proposed method reduces CPU times to approximately half of those of PnP-ADMM in all settings, since it solves the problem in (16) without matrix inversions.

To evaluate stability, we present the PSNR and the update rate  $c_n$  at each iteration in the proposed method and PnP-PDS (Unstable) in Fig. 9. For PnP-PDS (Unstable),  $c_n$  behaves unstably depending on the image. In contrast, the proposed method shows a consistent decrease in  $c_n$  over iterations, along with stable PSNR values.

Fig. 8 shows the visual results of the deblurring at  $\eta = 100$ . PnP-PDS (Unstable) loses the details of the original images, while PnP-ADMM and RED show slight changes in brightness or fine shapes. Although BPG preserves the structure of the image well, it also produced several visually unpleasant artifacts (e.g., above the man's shoulder and head). In contrast, the proposed method recovers fine details, yielding a visually natural image.

#### D. Influence of the Prior on the Proposed Method

We study how the choice of prior influences the performance of the proposed method. As a representative case, we focus on how the noise level  $\sigma_J$  assumed during the training of the Gaussian denoiser affects reconstruction performance of deblurring/inpainting under Gaussian noise.

We trained denoisers with four fixed noise levels, specifically  $\sigma_J \in \{0.0075, 0.01, 0.05, 0.1\}$  where  $\sigma_J$  denotes the standard deviation of the Gaussian noise added to inputs during denoiser training (i.e., in (7),  $\mathbf{x}_\ell$  is corrupted with Gaussian noise of standard deviation  $\sigma_J$ ). In addition, we trained a mixed-noise-level model, where  $\sigma_J$  was randomly sampled from the range  $[0, 0.1]$  for each image. In all cases, the values of  $\tau$  and  $\xi$  in (7) were manually tuned. For more details, we follow the training procedure described in [18].

The performance comparison is shown in Fig. 10. The noise level  $\sigma$  of the observation was selected from seven values between 0.005 and 0.1. For each fixed  $\sigma$ , the proposed method exhibits nearly identical performance across all values of  $\sigma_J$ , with the maximum PSNR and SSIM differences being 1.215 dB (in deblurring at  $\sigma = 0.005$ ) and 0.021 (in deblurring at  $\sigma = 0.025$ ), respectively. These results indicate that the proposed method can effectively leverage the prior of denoisers trained with a wide range of  $\sigma_J$ . Furthermore, it achieves similarly competitive performance even when using a denoiser trained with randomly varying noise levels  $\sigma_J \in [0, 0.1]$ , suggesting that its regularization effect is not highly sensitive to the specific training noise level in the proposed method.

Finally, we note that the consistent performance regardless of  $\sigma_J$  and  $\sigma$  can be attributed to expressing the data-fidelity as a constraint. By formulating it as a constraint, the consistency between the reconstruction and the observation can be explicitly controlled via the parameter  $\varepsilon$ . This helps prevent overly aggressive denoising that could arise when  $\sigma_J$  is much larger than  $\sigma$ .

## V. CONCLUSION

In this paper, we have proposed a general PnP-PDS method with a theoretical convergence guarantee under realistic assumptions in real-world settings, supported by extensive experimental results. First, we established the convergent PnP-PDS

using a firmly nonexpansive denoiser and theoretically characterized its solution set. Then, we showed that it efficiently solves various image restoration problems involving nonsmooth data-fidelity terms and additional hard constraints without the need to compute matrix inversions or perform inner iterations. Finally, through numerical experiments on deblurring/inpainting under Gaussian noise and deblurring/inpainting under Poisson noise, we demonstrated that the proposed PnP-PDS outperforms existing methods with stable convergence behavior. We believe that the proposed PnP-PDS holds great potential for a wide range of applications including biomedical imaging, astronomical imaging, and remote sensing.

## APPENDIX

### PROOFS OF PROPOSITION 3.1 AND PROPOSITION 3.2

We start by introducing several key facts, followed by the proofs.

*Fact 1* ([13, Lemma 4.4]). Let  $B_1 : \mathcal{H} \rightarrow 2^{\mathcal{H}}$  be a maximally monotone operator, and  $B_2 : \mathcal{H} \rightarrow \mathcal{H}$  be a  $\kappa$ -cocoercive operator for some  $\kappa \in (0, \infty)$ . Suppose that  $\text{zer}(B_1 + B_2)$  is nonempty, and the following conditions hold:

- (i)  $\gamma \in (0, 2\kappa)$ ,
- (ii)  $\rho_n \in (0, \delta)$ ,
- (iii)  $\sum_{n \in \mathbb{N}} \rho_n(\delta - \rho_n) = \infty$ ,

where  $\delta := 2 - \gamma(2\kappa)^{-1}$ . Consider the sequence  $\{\mathbf{s}_n\}_{n \in \mathbb{N}}$  generated by the following update:

$$\mathbf{s}_{n+1} = \rho_n((\text{Id} + \gamma B_1)^{-1}(\text{Id} - \gamma B_2)(\mathbf{s}_n)) + (1 - \rho_n)\mathbf{s}_n. \quad (24)$$

The sequence  $\{\mathbf{s}_n\}_{n \in \mathbb{N}}$  generated by (24) weakly converges to  $\hat{\mathbf{s}}$  s.t.  $\hat{\mathbf{s}} \in \text{zer}(B_1 + B_2)$ .

*Fact 2* ([13, Lemma 4.2]). Let  $B : \mathcal{H} \rightarrow 2^{\mathcal{H}}$  be a maximally monotone operator. Suppose that  $\text{zer}(B)$  is nonempty, and the following conditions hold:

- (i)  $\rho_n \in (0, 2)$ ,
- (ii)  $\sum_{n \in \mathbb{N}} \rho_n(2 - \rho_n) = \infty$ .

Consider the sequence  $\{\mathbf{s}_n\}_{n \in \mathbb{N}}$  generated by the following update:

$$\mathbf{s}_{n+1} = \rho_n((\text{Id} + B)^{-1}(\mathbf{s}_n)) + (1 - \rho_n)\mathbf{s}_n. \quad (25)$$

The sequence  $\{\mathbf{s}_n\}_{n \in \mathbb{N}}$  generated by (25) weakly converges to  $\hat{\mathbf{s}}$  s.t.  $\hat{\mathbf{s}} \in \text{zer}(B)$ .

*Fact 3* ([13, Lemma 4.5]). Let  $J : \mathcal{H} \rightarrow \mathbb{R}$  be a convex differentiable function. If  $\kappa \nabla J$  is nonexpansive for some  $\kappa \in (0, \infty)$ , then  $\kappa \nabla J$  becomes firmly nonexpansive.

*Proof of Proposition 3.1.* We follow the proof structure as in [13, Theorem 3.1 for Algorithm 3.1]. Consider the real Hilbert space  $\mathcal{Z}_I = \mathcal{X} \times \mathcal{Y}$  equipped with the inner product defined as  $\langle \mathbf{z}, \mathbf{z}' \rangle_I := \langle \mathbf{x}, \mathbf{x}' \rangle + \langle \mathbf{y}, \mathbf{y}' \rangle$  where  $\mathbf{z} = (\mathbf{x}, \mathbf{y})$ ,  $\mathbf{z}' = (\mathbf{x}', \mathbf{y}')$ . Now, we define  $P$  as

$$P : \begin{pmatrix} \mathbf{x} \\ \mathbf{y} \end{pmatrix} \rightarrow \begin{pmatrix} \text{Id}/\gamma_1 & -\mathbf{L}^* \\ -\mathbf{L} & \text{Id}/\gamma_2 \end{pmatrix} \begin{pmatrix} \mathbf{x} \\ \mathbf{y} \end{pmatrix}.$$

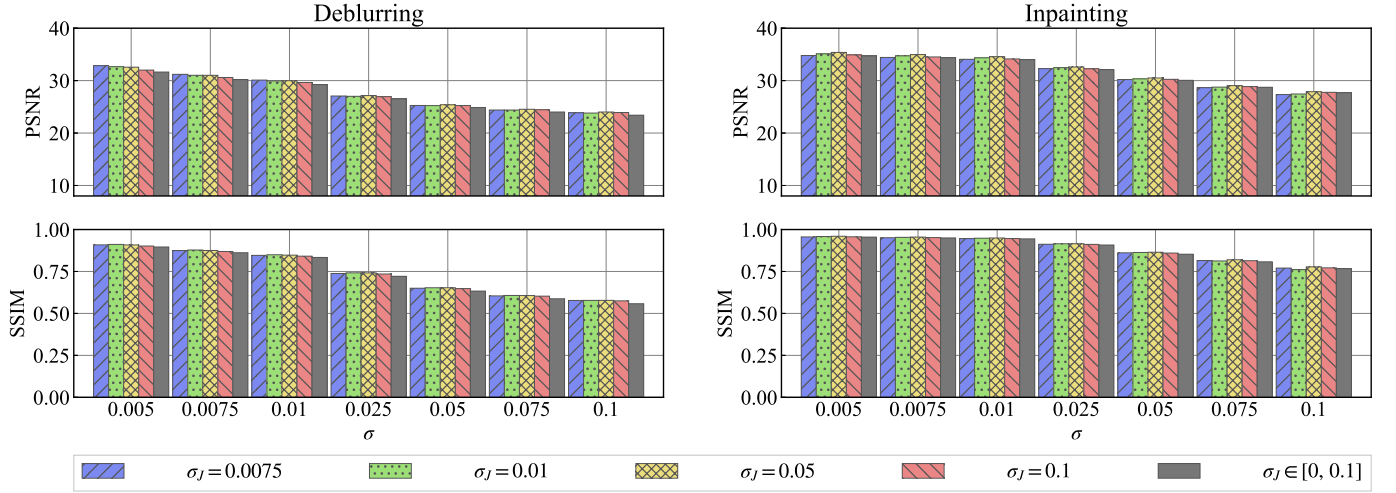


Fig. 10: The influence of the variation in  $\sigma_J$  on the restoration performance for the proposed method. The plotted values are the average of seven images from ImageNet [68].

Then, as shown in [13],  $P$  is bounded, self-adjoint, and strictly positive. Hence, we obtain the real Hilbert space  $\mathcal{Z}_P = \mathcal{X} \times \mathcal{Y}$  equipped with the following inner product:

$$\langle \mathbf{z}, \mathbf{z}' \rangle_P := \langle \mathbf{z}, P\mathbf{z}' \rangle_I,$$

Since convergence in  $\mathcal{Z}_I$  and  $\mathcal{Z}_P$  are equivalent [13, Theorem 3.1 for Algorithm 3.1], we show that the sequence generated by the iteration in (9) weakly converges to the solution to (10) in  $\mathcal{Z}_P$ .

For simple representation of Algorithm (9), let us define operators  $T_1$  and  $T_2$  as follows:

$$T_1(\mathbf{z}_n) := \begin{pmatrix} \gamma_1^{-1} A_J(\mathbf{x}_n) + \mathbf{L}^* \mathbf{y}_n \\ -\mathbf{L}\mathbf{x}_n + \partial h^*(\mathbf{y}_n) \end{pmatrix}, T_2(\mathbf{z}_n) := \begin{pmatrix} \nabla f(\mathbf{x}_n) \\ 0 \end{pmatrix},$$

where  $\mathbf{z}_n := (\mathbf{x}_n^\top, \mathbf{y}_n^\top)^\top$ . Then, we rewrite Algorithm (9) as

$$\mathbf{z}_{n+1} = \rho_n (\text{Id} + M_1)^{-1} (\text{Id} - M_2) \mathbf{z}_n + (1 - \rho_n) \mathbf{z}_n, \quad (26)$$

where  $M_1 := P^{-1} \circ T_1$  and  $M_2 := P^{-1} \circ T_2$ . Since the update in (26) has the same form as (24) with  $\gamma = 1$ , it is sufficient to demonstrate that each assumption in Fact 1 is satisfied. Hereunder, we define  $\kappa$  as follows:

$$\kappa := \frac{1}{\beta} \left( \frac{1}{\gamma_1} - \gamma_2 \|\mathbf{L}\|_{\text{op}}^2 \right). \quad (27)$$

- We see that the set-valued operator  $A_J = J^{-1} - \text{Id}$  is maximally monotone from [55, Proposition 23.8 (iii)]. Hence,  $\gamma_1^{-1} A_J$  is also maximally monotone [55, Proposition 20.22]. The operator  $\partial h^*$  is maximally monotone from [55, Theorem 20.48, Proposition 20.22, and Corollary 16.30]. Therefore, the operator  $(\mathbf{x}, \mathbf{y}) \mapsto \gamma_1^{-1} A_J(\mathbf{x}) \times \partial h^*(\mathbf{y})$  is maximally monotone in  $\mathcal{Z}_I$  from [55, Proposition 20.23]. Furthermore, according to [55, Example 20.35],  $(\mathbf{x}, \mathbf{y}) \mapsto \mathbf{L}^* \mathbf{y} \times -\mathbf{L}\mathbf{x}$  is maximally monotone and has a full domain. Therefore,  $T_1$  becomes a maximally monotone operator on  $\mathcal{Z}_I$  [55, Corollary 25.5 (i)]. Since  $P$  is injective,  $M_1 = P^{-1} \circ T_1$  is maximally monotone on  $\mathcal{Z}_P$ .

- From Fact 3 and the computation presented in [13, Theorem 3.1 for Algorithm 3.1],  $M_2$  is cocoercive, i.e.,  $\kappa M_2$  is firmly nonexpansive in  $\mathcal{Z}_P$ .
- We set  $\gamma = 1$ . Since  $0 < 2\kappa < 1$  from (i) of Proposition 3.1, the assumption (i) of Fact 1 is satisfied.
- The assumption (ii) of Fact 1 is satisfied from (27).
- The set of solutions to (10) coincides  $\text{zer}(T_1 + T_2) = \text{zer}(M_1 + M_2)$ , and this set is nonempty by the assumption.

Therefore, by applying Fact 1, we have that  $\mathbf{z}_n$  weakly converges to  $\hat{\mathbf{z}} \in \text{zer}(T_1 + T_2)$  on  $\mathcal{Z}_P$ , which implies  $\{\mathbf{x}_n, \mathbf{y}_n\}_{n \in \mathbb{N}}$  weakly converges to a solution to (10).  $\square$

*Proof of Proposition 3.2.* Following the definition of  $P$  as in Proposition 3.1,  $P$  is bounded, linear, and self-adjoint. Since condition (i) does not include equality,  $P$  is strictly positive. Therefore, same as the case in Proposition 3.1, we can define a Hilbert space  $\mathcal{Z}_P$  equipped with the inner product  $\langle \cdot, \cdot \rangle_P$ . Since  $f(\mathbf{x}) = 0$ , we have  $T_2 = 0$ , and the update in (26) becomes

$$\mathbf{z}_{n+1} = \rho_n (\text{Id} + M_1)^{-1} \mathbf{z}_n + (1 - \rho_n) \mathbf{z}_n. \quad (28)$$

The update in (28) is the same form as Fact 2 by setting  $B = M_1$ , so we need to show that the assumptions of Fact 2 are satisfied. Same as the proof of Proposition 3.1,  $M_1$  becomes a maximally monotone operator, and from the assumption,  $\text{zer}(T_1 + T_2) = \text{zer}(T_1) = \text{zer}(M_1)$  is nonempty. Furthermore, the assumption on  $\rho_n$  in Proposition 3.1 is the same as in Fact 2. Therefore, by Fact 2,  $\mathbf{z}_n$  weakly converges to  $\text{zer}(T_1 + T_2)$ , which means  $\{\mathbf{x}_n, \mathbf{y}_n\}$  weakly converges to the solution to (10).  $\square$

## REFERENCES

- [1] L. I. Rudin, S. Osher, and E. Fatemi, "Nonlinear total variation based noise removal algorithms," *Physica D*, vol. 60, pp. 259–268, 1992.
- [2] K. Bredies, K. Kunisch, and T. Pock, "Total generalized variation," *SIAM J. Imag. Sci.*, vol. 3, no. 3, p. 492–526, 2010.
- [3] S. Ono and I. Yamada, "Decorrelated vectorial total variation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2014, pp. 4090–4097.

- [4] S. Lefkimmiatis, A. Roussos, P. Maragos, and M. Unser, "Structure tensor total variation," *SIAM J. Imag. Sci.*, vol. 8, no. 2, pp. 1090–1122, 2015.
- [5] L. Condat, "Discrete total variation: New definition and minimization," *SIAM J. Imag. Sci.*, vol. 10, no. 3, pp. 1258–1290, 2017.
- [6] P. L. Combettes and J. C. Pesquet, *Proximal Splitting Methods in Signal Processing*. Springer New York, 2011, pp. 185–212.
- [7] N. Parikh and S. Boyd, "Proximal algorithms," *Found. Trends Optim.*, vol. 1, no. 3, pp. 127–239, 2014.
- [8] L. Condat, D. Kitahara, A. Contreras, and A. Hirabayashi, "Proximal splitting algorithms for convex optimization: A tour of recent advances, with new twists," *SIAM Rev.*, vol. 65, no. 2, pp. 375–435, 2023.
- [9] P. L. Combettes and V. R. Wajs, "Signal recovery by proximal forward-backward splitting," *Multiscale Modeling & Simulation*, vol. 4, no. 4, pp. 1168–1200, 2005.
- [10] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, "Distributed optimization and statistical learning via the alternating direction method of multipliers," *Found. Trends Mach. Learn.*, vol. 3, pp. 1–122, 2011.
- [11] A. Chambolle and T. Pock, "A first-order primal-dual algorithm for convex problems with applications to imaging," *J. Math. Imag. Vis.*, vol. 40, no. 1, pp. 120–145, 2010.
- [12] P. L. Combettes and J. C. Pesquet, "Primal-dual splitting algorithm for solving inclusions with mixtures of composite, lipschitzian, and parallel-sum type monotone operators," *Set-Valued Var. Anal.*, vol. 20, no. 2, p. 307–330, 2012.
- [13] L. Condat, "A primal-dual splitting method for convex optimization involving lipschitzian, proximable and linear composite terms," *J. Opt. Theory Appl.*, vol. 158, no. 2, pp. 460–479, 2013.
- [14] B. C. Vu, "A splitting algorithm for dual monotone inclusions involving cocoercive operators," *Adv. Comput. Math.*, vol. 38, no. 3, p. 667–681, 2013.
- [15] P. L. Combettes, L. Condat, J. C. Pesquet, and B. Vu, "A forward-backward view of some primal-dual optimization methods in image recovery," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, 2014, pp. 4141–4145.
- [16] L. Condat, "A generic proximal algorithm for convex optimization—application to total variation minimization," *IEEE Signal Process. Lett.*, vol. 21, no. 8, pp. 985–989, 2014.
- [17] Y. Sun, B. Wohlberg, and U. Kamilov, "An online plug-and-play algorithm for regularized image reconstruction," *IEEE Trans. Comput. Imag.*, vol. 5, no. 3, pp. 395–408, 2019.
- [18] J. C. Pesquet, A. Repetti, M. Terris, and Y. Wiaux, "Learning maximally monotone operators for image recovery," *SIAM J. Imag. Sci.*, vol. 14, no. 3, pp. 1206–1237, 2021.
- [19] A. Ebner and M. Haltmeier, "Plug-and-play image reconstruction is a convergent regularization method," *IEEE Trans. Image Process.*, vol. 33, p. 1476–1486, 2024.
- [20] S. V. Venkatakrishnan, C. A. Bouman, and B. Wohlberg, "Plug-and-play priors for model based reconstruction," in *Proc. IEEE Global Conf. Signal and Inf. Process.*, 2013, pp. 945–948.
- [21] A. Rond, R. Giryes, and M. Elad, "Poisson inverse problems by the plug-and-play scheme," *J. Vis. Comm. and Imag. Rep.*, vol. 41, pp. 96–108, 2016.
- [22] 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. Comput. Imag.*, vol. 2, no. 4, p. 408–423, 2016.
- [23] S. H. Chan, X. Wang, and O. A. Elgindy, "Plug-and-play ADMM for image restoration: Fixed-point convergence and applications," *IEEE Trans. Comput. Imag.*, vol. 3, no. 1, pp. 84–98, 2017.
- [24] P. Nair, R. G. Gavaskar, and K. N. Chaudhury, "Fixed-point and objective convergence of plug-and-play algorithms," *IEEE Trans. Comput. Imag.*, vol. 7, pp. 337–348, 2021.
- [25] Y. Sun, Z. Wu, X. Xu, B. Wohlberg, and U. S. Kamilov, "Scalable plug-and-play ADMM with convergence guarantees," *IEEE Trans. Comput. Imag.*, vol. 7, pp. 849–863, 2021.
- [26] S. Ono, "Primal-dual plug-and-play image restoration," *IEEE Signal Process. Lett.*, vol. 24, no. 8, pp. 1108–1112, 2017.
- [27] C. S. Garcia, M. Larchevêque, S. O'Sullivan, M. V. Waerebeke, R. Thomson, A. Repetti, and J.-C. Pesquet, "A primal-dual data-driven method for computational optical imaging with a photonic lantern," *PNAS Nexus*, vol. 3, no. 4, p. 164, 2024.
- [28] A. Danielyan, V. Katkovnik, and K. Egiazarian, "BM3D frames and variational image deblurring," *IEEE Trans. Image Process.*, vol. 21, no. 4, pp. 1715–1728, 2012.
- [29] J. Zhang, D. Zhao, and W. Gao, "Group-based sparse representation for image restoration," *IEEE Trans. Image Process.*, vol. 23, no. 8, pp. 3336–3351, 2014.
- [30] K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian, "Image denoising by sparse 3-D transform-domain collaborative filtering," *IEEE Trans. Image Process.*, vol. 16, no. 8, pp. 2080–2095, 2007.
- [31] A. Buades, B. Coll, and J.-M. Morel, "Non-local means denoising," *Imag. Process. On Line*, vol. 1, pp. 208–212, 2011.
- [32] Y. Chen and T. Pock, "Trainable nonlinear reaction diffusion: A flexible framework for fast and effective image restoration," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 6, pp. 1256–1272, 2017.
- [33] 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, p. 3142–3155, 2017.
- [34] Z. Wu, Y. Sun, A. Matlock, J. Liu, L. Tian, and U. Kamilov, "SIMBA: Scalable inversion in optical tomography using deep denoising priors," *IEEE J. Sel. Topics Signal Process.*, vol. 14, no. 6, pp. 1163–1175, 2020.
- [35] R. Ahmad, C. A. Bouman, G. T. Buzzard, S. Chan, S. Liu, E. T. Reehorst, and P. Schniter, "Plug-and-play methods for magnetic resonance imaging: Using denoisers for image recovery," *IEEE Signal Process. Mag.*, vol. 37, no. 1, pp. 105–116, 2020.
- [36] C. Pellizzari, M. Spencer, and C. Bouman, "Coherent plug-and-play: Digital holographic imaging through atmospheric turbulence using model-based iterative reconstruction and convolutional neural networks," *IEEE Trans. Comput. Imag.*, vol. 6, pp. 1–1, 12 2020.
- [37] U. S. Kamilov, C. A. Bouman, G. T. Buzzard, and B. Wohlberg, "Plug-and-play methods for integrating physical and learned models in computational imaging," *IEEE Signal Process. Mag.*, vol. 40, no. 1, pp. 85–97, 2023.
- [38] K. Zhang, Y. Li, W. Zuo, L. Zhang, L. Van Gool, and R. Timofte, "Plug-and-play image restoration with deep denoiser prior," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 44, no. 10, p. 6360–6376, 2022.
- [39] M. Duff, N. D. F. Campbell, and M. J. Ehrhardt, "Regularising inverse problems with generative machine learning models," *J. Math. Imag. Vis.*, vol. 66, pp. 37–56, 2024.
- [40] D. P. Kingma and M. Welling, "An introduction to variational autoencoders," *Found. Trends Mach. Learn.*, vol. 12, no. 4, p. 307–392, 2019.
- [41] M. Arjovsky, S. Chintala, and L. Bottou, "Wasserstein generative adversarial networks," in *Proc. Int. Conf. Mach. Learn. (ICML)*, ser. Proc. Mach. Learn. Research, 2017, pp. 214–223.
- [42] P. Dhariwal and A. Nichol, "Diffusion models beat gans on image synthesis," in *Proc. Conf. Neural Inf. Process. Sys. (NeurIPS)*, 2021, p. 8780–8794.
- [43] Y. Zhu, K. Zhang, J. Liang, J. Cao, B. Wen, R. Timofte, and L. V. Gool, "Denoising diffusion models for plug-and-play image restoration," in *IEEE Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, 2023, pp. 1219–1229.
- [44] E. Ryu, J. Liu, S. Wang, X. Chen, Z. Wang, and W. Yin, "Plug-and-play methods provably converge with properly trained denoisers," in *Proc. Int. Conf. Mach. Learn. (ICML)*, 2019, pp. 5546–5557.
- [45] Y. Romano, M. Elad, and P. Milanfar, "The little engine that could: regularization by denoising (RED)," *SIAM J. Imag. Sci.*, vol. 10, no. 4, p. 1804–1844, 2017.
- [46] R. Cohen, M. Elad, and P. Milanfar, "Regularization by denoising via fixed-point projection (RED-PRO)," *SIAM J. Imag. Sci.*, vol. 14, no. 3, pp. 1374–1406, 2021.
- [47] R. Cohen, Y. Blau, D. Freedman, and E. Rivlin, "It has potential: Gradient-driven denoisers for convergent solutions to inverse problems," in *Proc. Conf. Neural Inf. Process. Sys. (NeurIPS)*, 2021, pp. 18 152–18 164.
- [48] S. Hurault, A. Leclaire, and N. Papadakis, "Gradient step denoiser for convergent plug-and-play," in *Int. Conf. Learn. Represent.*, 2022.
- [49] S. Hurault, A. Leclaire, and N. Papadakis, "Proximal denoiser for convergent plug-and-play optimization with nonconvex regularization," in *Proc. Int. Conf. Mach. Learn. (ICML)*, 2022, pp. 9483–9505.
- [50] H. Y. Tan, S. Mukherjee, J. Tang, and C.-B. Schönlieb, "Provably convergent plug-and-play quasi-newton methods," *SIAM J. Imag. Sci.*, vol. 17, no. 2, pp. 785–819, 2024.
- [51] S. Hurault, U. Kamilov, A. Leclaire, and N. Papadakis, "Convergent Bregman plug-and-play image restoration for poisson inverse problems," in *Adv. Neural Inf. Process. Sys.*, vol. 36, 2023, pp. 27 251–27 280.
- [52] E. T. Reehorst and P. Schniter, "Regularization by denoising: Clarifications and new interpretations," *IEEE Trans. Comput. Imag.*, vol. 5, no. 1, pp. 52–67, 2019.
- [53] J. Hertrich, S. Neumayer, and G. Steidl, "Convolutional proximal neural networks and plug-and-play algorithms," *Linear Algebra Appl.*, vol. 631, pp. 203–234, 2021.

- [54] Y. Suzuki, R. Isono, and S. Ono, "A convergent primal-dual deep plug-and-play algorithm for constrained image restoration," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, 2024, pp. 9541–9545.
- [55] H. H. Bauschke and P. L. Combettes, *Correction to: Convex Analysis and Monotone Operator Theory in Hilbert Spaces*, 2nd ed. Springer Int. Publishing, 2017.
- [56] J. J. Moreau, "Fonctions convexes duales et points proximaux dans un espace hilbertien," *Comptes Rendus de l'Academie Bulg. des Sci.*, vol. 255, p. 2897–2899, 1962.
- [57] P. Combettes and N. N. Reyes, "Moreau's decomposition in banach spaces," *Math. Program.*, vol. 139, no. 1-2, pp. 103–114, 2013.
- [58] G. H. Golub and C. F. V. Loan, *Matrix computations*. JHU press, 2013.
- [59] R. Bhatia and F. Kittaneh, "Norm inequalities for partitioned operators and an application," *Math. Ann.*, vol. 287, pp. 719–726, 1990.
- [60] G. Chierchia, N. Pustelnik, J. Pesquet, and B. Pesquet, "Epigraphical projection and proximal tools for solving constrained convex optimization problems," *Signal, Image and Video Process.*, vol. 9, no. 8, pp. 1737–1749, 2015.
- [61] S. Ono, "Efficient constrained signal reconstruction by randomized epigraphical projection," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, 2019, pp. 4993–4997.
- [62] M. Afonso, J. Bioucas-Dias, and M. Figueiredo, "An augmented Lagrangian approach to the constrained optimization formulation of imaging inverse problems," *IEEE Trans. Image Process.*, vol. 20, no. 3, pp. 681–95, 2011.
- [63] S. Ono and I. Yamada, "Signal recovery with certain involved convex data-fidelity constraints," *IEEE Trans. Signal Process.*, vol. 63, no. 22, pp. 6149–6163, 2015.
- [64] S. Ono, " $L_0$  gradient projection," *IEEE Trans. Image Process.*, vol. 26, no. 4, pp. 1554–1564, 2017.
- [65] P. L. Combettes and J.-C. Pesquet, "A Douglas–Rachford splitting approach to nonsmooth convex variational signal recovery," *IEEE J. Sel. Topics Signal Process.*, vol. 1, no. 4, pp. 564–574, 2007.
- [66] A. Levin, Y. Weiss, F. Durand, and W. T. Freeman, "Understanding and evaluating blind deconvolution algorithms," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2009, pp. 1964–1971.
- [67] C. Bertocchi, E. Chouzenoux, M. C. Corbineau, J. C. Pesquet, and M. Prato, "Deep unfolding of a proximal interior point method for image restoration," *Inverse Prob.*, vol. 36, no. 3, p. 034005, 2020.
- [68] J. Deng, W. Dong, R. Socher, L. J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2009, pp. 248–255.
- [69] M. Terris, T. Moreau, N. Pustelnik, and J. Tachella, "Equivariant plug-and-play image reconstruction," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2024, pp. 25 255–25 264.



**Shunsuke Ono** (S'11–M'15–SM'23) received a B.E. degree in Computer Science in 2010 and M.E. and Ph.D. degrees in Communications and Computer Engineering in 2012 and 2014 from the Tokyo Institute of Technology, respectively. From 2012 to 2014, he was a Research Fellow (DC1) of the Japan Society for the Promotion of Science (JSPS). He was an Assistant, then an Associate Professor with Tokyo Institute of Technology (TokyoTech), Tokyo, Japan, from 2014 to 2024. From 2016 to 2020, he was a Researcher of Precursory Research for Embryonic Science and Technology (PRESTO), Japan Science and Technology Agency (JST), Tokyo, Japan. Currently, he is an Associate Professor with Institute of Science Tokyo (Science Tokyo), Tokyo, Japan. His research interests include signal processing, image analysis, optimization, remote sensing, and measurement informatics. He has served as an Associate Editor for IEEE TRANSACTIONS ON SIGNAL AND INFORMATION PROCESSING OVER NETWORKS (2019–2024). Dr. Ono was a recipient of the Young Researchers' Award and the Excellent Paper Award from the IEICE in 2013 and 2014, respectively, the Outstanding Student Journal Paper Award and the Young Author Best Paper Award from the IEEE SPS Japan Chapter in 2014 and 2020, respectively, and the Best Paper Award in APSIPA ASC 2024. He also received the Funai Research Award in 2017, the Ando Incentive Prize in 2021, the MEXT Young Scientists' Award in 2022, the IEEE SPS Outstanding Editorial Board Member Award in 2023, and the KDDI Foundation Award in 2025.



**Yodai Suzuki** (S'24) received a B.E. degree in Information and Computer Science in 2024 from the Tokyo Institute of Technology. He is currently pursuing a master's degree with the Department of Computer Science at the Institute of Science Tokyo. His current research interests include signal and image processing, mathematical optimization, neural networks, and convex analysis. He received the Young Researchers' Award from the IEICE and the PCSJ/IMPS Student Paper Award in 2025.



**Ryoosuke Isono** (S'23) received B.E. and M.E. degrees in Information and Computer Science in 2022 from the Osaka University and from the Tokyo Institute of Technology, respectively. He is currently pursuing a Ph.D. degree with the Department of Computer Science at the Institute of Science Tokyo. His current research interests include signal and image processing, mathematical optimization, and remote sensing. He received the PCSJ/IMPS Best Poster Award in 2024.