Plug-and-Play image restoration with Stochastic deNOising REgularization

Marien Renaud [1]   **Jean Prost** [1 2]   **Arthur Leclaire** [3]   **Nicolas Papadakis** [1]

Abstract

Plug-and-Play (PnP) algorithms are a class of iterative algorithms that address image inverse problems by combining a physical model and a deep neural network for regularization. Even if they produce impressive image restoration results, these algorithms rely on a non-standard use of a denoiser on images that are less and less noisy along the iterations, which contrasts with recent algorithms based on Diffusion Models (DM), where the denoiser is applied only on re-noised images. We propose a new PnP framework, called Stochastic deNOising REgularization (SNORE), which applies the denoiser only on images with noise of the adequate level. It is based on an explicit stochastic regularization, which leads to a stochastic gradient descent algorithm to solve ill-posed inverse problems. A convergence analysis of this algorithm and its annealing extension is provided. Experimentally, we prove that SNORE is competitive with respect to state-of-the-art methods on deblurring and inpainting tasks, both quantitatively and qualitatively.

1. Introduction

Many imaging problems can be formulated as *inverse problems* seeking to recover high-quality images $\mathbf{x}^*$ from their low-quality observations $\mathbf{y}$ by solving a problem of the form

$$\mathbf{x}^* \in \underset{\mathbf{x} \in \mathbb{R}^d}{\arg\min} \, \mathcal{F}(\mathbf{x}, \mathbf{y}) + \alpha \mathcal{R}(\mathbf{x}), \tag{1}$$

where $\mathcal{F}$ measures the fidelity to the degraded observation $\mathbf{y}$ and $\mathcal{R}$ is a regularization term weighted by a parameter $\alpha > 0$.

The regularization is crucial to complete the missing information of the observation by bringing prior knowledge on the high-quality image. Choosing a relevant prior $\mathcal{R}$ in Problem (1) is a long-standing (Rudin et al., 1992; Mallat, 1999; Zoran & Weiss, 2011) and challenging task and recent approaches explore deep learning techniques that learn a prior from a database of clean images (Zhang et al., 2017; Lunz et al., 2018; Laumont et al., 2023).

Problem (1) can be addressed with proximal splitting algorithms (Boyd et al., 2011) which are first-order optimization algorithms based on the recursive application of gradient-descent and/or proximal operators of functions $\mathcal{F}$ and $\mathcal{R}$.

The Plug-and-Play (PnP) framework (Venkatakrishnan et al., 2013) consists in replacing, within a proximal splitting algorithm, the proximal step on the regularization $\mathcal{R}$ by a denoising operation; and it allows to use implicit regularization priors encoded by pre-trained image denoisers. Similarly, following the Regularization by Denoising (RED) framework (Romano et al., 2017), a gradient-descent step on the regularization can be substituted by a learned denoiser. It has been observed that plugging a pre-trained state-of-the-art deep denoiser is essential for achieving the best quality results in many imaging contexts (Metzler et al., 2018; Ryu et al., 2019; Hurault et al., 2022b; Renaud et al., 2024; Ulondu-Mendes et al., 2023).

In another line of works, inverse problems solvers based on denoising diffusion models (DDM) (Ho et al., 2020; Song et al., 2020) have demonstrated their ability to provide high-quality restoration even for severely ill-posed problems where a large amount of information is missing (Chung et al., 2023; Song et al., 2023). DDM and PnP both rely on deep denoisers to implicitly model the prior distribution, and they decouple prior and data-fidelity terms in order to provide flexible solvers. However, while the theoretical properties of PnP algorithms regarding convergence have been studied in depth (Sreehari et al., 2016; Sun et al., 2019b; Gavaskar & Chaudhury, 2019; Ryu et al., 2019; Xu et al., 2020; Cohen et al., 2021; Hurault et al., 2022a), the theoretical properties of restoration algorithms based on DDM remain poorly understood. Indeed, DDM methods rely on heuristics to approximate the score of the intractable likelihood. To the best of our knowledge, the impact of the approximation error on the generated samples remains to be quantified.

[1]Univ. Bordeaux, CNRS, INRIA, Bordeaux INP, IMB, UMR 5251, F-33400 Talence, France [2]Université Paris-Cité, CNRS, MAP5, France [3]LTCI, Télécom Paris, IP Paris. Correspondence to: Marien Renaud <marien.renaud@math.u-bordeaux.fr>.

In this work, we seek to develop an inverse problem solver that inherits the superior restoration quality of DDM methods, while satisfying the theoretical guarantees of convergence that are met for some PnP algorithms. Our key observation is that, contrary to the PnP framework, in diffusion based methods the denoiser is applied on a *noisy version of the image* at each iteration of the sampling process. We postulate that applying the denoiser to noisy images is a main ingredient to the impressive performance of diffusion-based samplers, as it avoids a domain shift relative to the data on which the denoiser was trained.

This motivates us to propose SNORE (Stochastic deNOising REgularization), a stochastic PnP algorithm, which differs from classical PnP schemes by injecting noise to the input of the denoiser at each iteration. SNORE minimizes a classical variational objective, where the regularization term is defined as the average value of the smoothed log prior on noisy version of the image of interest, and can be viewed as a relaxed version of the usual negative log-prior.

Contributions. **(a)** We propose a new explicit regularization leading to a novel PnP framework, named *Stochastic deNOising Regularization (SNORE)*, in which the denoiser is applied on a noisy version of the image at each iteration. **(b)** We show that SNORE can be optimized by a stochastic gradient-descent algorithm (Algorithm 2). We prove that this algorithm converges with the exact MMSE denoiser (Proposition 3.3) and we bound the error with an inexact MMSE denoiser (Proposition 3.5). **(c)** With a critical point analysis (Proposition 3.2), we motivate the practical use of an annealed algorithm (Algorithm 3). Finally, we demonstrate the efficiency of SNORE to solve inverse problems.

2. Stochastic deNOising REgularization (SNORE)

In this section, we propose a new stochastic regularization, SNORE (Equation 5), that can be used for PnP restoration in such a way that the denoising step now applies to a noisy version of the current image.

We first recall the RED regularization (Section 2.1) and the corresponding PnP algorithm (Algorithm 1). Then we introduce the SNORE regularization leading to Algorithm 2. We also propose an Annealing SNORE algorithm (Algorithm 3) inspired by annealed importance sampling (Neal, 2001). Finally, we discuss the positioning of our method in relation to existing related works (Section 2.3).

2.1. Background

Bayesian inverse problem An inverse problem formulated as in Problem (1) has a general Bayesian interpretation. From the observation $\mathbf{y} \in \mathbb{R}^q$ (typically $q < d$), we can restore the image by computing the Maximum A Posteriori

(MAP) estimator $\hat{\mathbf{x}}$ defined by

$$\hat{\mathbf{x}} = \arg\max_{\mathbf{x} \in \mathbb{R}^d} p(\mathbf{x}|\mathbf{y}) = \arg\min_{\mathbf{x} \in \mathbb{R}^d} -\log p(\mathbf{x}|\mathbf{y})$$
$$= \arg\min_{\mathbf{x} \in \mathbb{R}^d} \underbrace{-\log p(\mathbf{y}|\mathbf{x})}_{=\mathcal{F}(\mathbf{x}, \mathbf{y})} \underbrace{-\log p(\mathbf{x})}_{=\mathcal{R}(\mathbf{x})}.$$

Thus the data-fidelity $\mathcal{F}$ is related to the image forward model, and the regularization $\mathcal{R}$ reflects the adopted prior model on images (which can be improper). Adding a weighting parameter $\alpha > 0$ (see Problem 1) is equivalent to adding a temperature parameter on the prior p, which becomes p^α.

Data Fidelity The forward model is supposed to have a known form

$$\mathbf{y} = \mathcal{A}(\mathbf{x}) + \mathbf{n},$$

with the degradation operator $\mathcal{A} : \mathbb{R}^d \mapsto \mathbb{R}^q$, and the noise $\mathbf{n} \sim \mathcal{N}(0, \sigma_\mathbf{y}^2 \mathbf{I}_q)$ where $\sigma_\mathbf{y} > 0$. Then $\mathcal{F}(\mathbf{x}, \mathbf{y}) = \frac{1}{\sigma_\mathbf{y}^2}\|\mathcal{A}(\mathbf{x}) - \mathbf{y}\|^2$.

Deep Learning regularization With a Bayesian interpretation, the regularization $\mathcal{R} = -\log p$ defines a model on the data. Recently, Deep Neural Networks (DNN) have proved their effectiveness in learning a realistic model from a database of observations. The RED framework (Romano et al., 2017) uses the performance of DNN for image restoration. It consists in adopting a prior regularization $\mathcal{R}$ whose gradient $\nabla\mathcal{R}$ is given by a pre-trained denoiser. This implicit relation relies on the regularization defined by

$$\mathcal{R}(\mathbf{x}) \approx \mathcal{P}_\sigma(\mathbf{x}) := -\log p_\sigma(\mathbf{x}), \quad (2)$$

where p_σ is the convolution $p * \mathcal{N}_\sigma$ between p and $\mathcal{N}_\sigma = \mathcal{N}(0, \sigma^2 \mathbf{I}_d)$. The relation between denoising and regularization is made explicit with Tweedie's formula (Efron, 2011):

$$\nabla\mathcal{P}_\sigma(\mathbf{x}) = -\frac{1}{\sigma^2}\left(D_\sigma^\star(\mathbf{x}) - \mathbf{x}\right), \quad (3)$$

where $D_\sigma^\star$ is the Minimum Mean Square Error (MMSE) denoiser is defined by

$$D_\sigma^\star(\tilde{\mathbf{x}}) := \mathbb{E}[\mathbf{x}|\tilde{\mathbf{x}}] = \int_{\mathbb{R}^d} \mathbf{x} p_{\mathbf{x}|\tilde{\mathbf{x}}}(\mathbf{x}|\tilde{\mathbf{x}}) d\mathbf{x}, \quad (4)$$

for $\tilde{\mathbf{x}} = \mathbf{x} + \boldsymbol{\epsilon}$ with $\mathbf{x} \sim p(\mathbf{x}), \boldsymbol{\epsilon} \sim \mathcal{N}(0, \sigma^2 \mathbf{I}_d)$.

In practice, we do not have access to the exact MMSE denoiser $D_\sigma^\star$, but only to a deep denoiser D_σ that is trained to approximate the MMSE $D_\sigma^\star$. Then a gradient descent scheme (Algorithm 1), as described by Reehorst & Schniter (2019), can be run to obtain an approximate solution of Problem (1).

Algorithm 1 involves the computation of $D_\sigma(\mathbf{x}_k)$, in which the denoiser is applied to an image iterate $\mathbf{x}_k$ that is not necessarily noisy. As a denoiser is trained to denoise images with noise, the application of D_σ to images that are out of the training domain might be irrelevant. To bypass this issue, we propose a new regularization.

Algorithm 1 RED (Romano et al., 2017)

1: **Param.:** $\mathbf{x}_0 \in \mathbb{R}^d$, $\sigma > 0$, $\alpha > 0$, $\delta > 0$, $N \in \mathbb{N}$
2: **Input:** degraded image $\mathbf{y}$
3: **Output:** restored image $\mathbf{x}_N$
4: **for** $k = 0, 1, \ldots, N - 1$ **do**
5: $\quad \mathbf{x}_{k+1} \leftarrow \mathbf{x}_k - \delta \nabla \mathcal{F}(\mathbf{x}_k, \mathbf{y}) - \frac{\alpha \delta}{\sigma^2} (\mathbf{x}_k - D_\sigma(\mathbf{x}_k))$
6: **end for**

2.2. SNORE regularization

We propose the SNORE regularization $\mathcal{R}_\sigma$, whose gradient applies the MMSE denoiser on noisy images. This new regularization is defined by

$$\mathcal{R}_\sigma(\mathbf{x}) = -\mathbb{E}_{\tilde{\mathbf{x}} \sim p_\sigma(\tilde{\mathbf{x}}|\mathbf{x})} \left(\log p_\sigma(\tilde{\mathbf{x}}) \right) \tag{5}$$

$$\nabla_{\mathbf{x}} \mathcal{R}_\sigma(\mathbf{x}) = -\frac{1}{\sigma^2} \left(\mathbb{E}_{\tilde{\mathbf{x}} \sim p_\sigma(\tilde{\mathbf{x}}|\mathbf{x})} \left(D_\sigma^\star(\tilde{\mathbf{x}}) \right) - \mathbf{x} \right). \tag{6}$$

Minimizing $\mathcal{R}_\sigma(\mathbf{x})$ is equivalent to maximizing $\mathbb{E}_{\tilde{\mathbf{x}} \sim p_\sigma(\tilde{\mathbf{x}}|\mathbf{x})} \left(\log p_\sigma(\tilde{\mathbf{x}}) \right)$. The last quantity is maximum if noisy versions of $\mathbf{x}$ are highly probable in the noisy prior distribution, $p_\sigma(\mathbf{x})$. In other words: *An image looks clean if its noisy versions look as noisy images.*

SNORE regularization can be seen as a relaxation of the classical PnP regularization $-\log(p * \mathcal{N}_\sigma)$, following the idea of Scarvelis et al. (2023).

$$\mathcal{R}_\sigma(\mathbf{x}) = - \left(\log(p * \mathcal{N}_\sigma) * \mathcal{N}_\sigma \right)(\mathbf{x}). \tag{7}$$

In Appendix C, we prove that $\mathcal{R}_\sigma$ provides the same minimum than $\mathcal{P}_\sigma$ if the prior is Gaussian. We also detail the case of Gaussian Mixture prior, with the convergence analysis of $\nabla \mathcal{P}_\sigma$ and $\nabla \mathcal{R}_\sigma$ to $-\nabla \log p$ when $\sigma \to 0$ and with a 1D illustration of the difference between $\nabla \mathcal{R}_\sigma$ and $-\nabla \log p_\sigma$.

Interpretation of the SNORE regularization We first underline that $\mathcal{R}_\sigma$ can be re-written as

$$\begin{aligned}\mathcal{R}_\sigma(\mathbf{x}) &= -\mathbb{E}_{\tilde{\mathbf{x}} \sim p_\sigma(\tilde{\mathbf{x}}|\mathbf{x})} \left(\log p_\sigma(\tilde{\mathbf{x}}) \right) \\ &= \mathcal{KL}(p_\sigma(\tilde{\mathbf{x}}|\mathbf{x}) \| p_\sigma(\tilde{\mathbf{x}})) - \mathbb{E}_{\tilde{\mathbf{x}} \sim p_\sigma(\tilde{\mathbf{x}}|\mathbf{x})} \left(\log p_\sigma(\tilde{\mathbf{x}}|\mathbf{x}) \right) \\ &= \mathcal{KL}(p_\sigma(\tilde{\mathbf{x}}|\mathbf{x}) \| p_\sigma(\tilde{\mathbf{x}})) + C, \end{aligned} \tag{8}$$

where we introduced the Kullback-Leibler divergence $\mathcal{KL}(\mu \| \nu) := \int_{\mathbb{R}^d} \log \left(\frac{d\mu}{d\nu} \right) d\mu$ and the constant

$$C = -\mathbb{E}_{\tilde{\mathbf{x}} \sim p_\sigma(\tilde{\mathbf{x}}|\mathbf{x})} \left(\log p_\sigma(\tilde{\mathbf{x}}|\mathbf{x}) \right) = \frac{d}{2} \left(1 + \log \left(2\pi\sigma^2 \right) \right).$$

Hence the potential $\mathcal{R}_\sigma(\mathbf{x})$ has the same optimization profile than $\mathcal{KL}(p_\sigma(\tilde{\mathbf{x}}|\mathbf{x}) \| p_\sigma(\tilde{\mathbf{x}}))$. This last quantity leads to another interpretation. Minimizing $\mathcal{R}_\sigma(\mathbf{x})$ is equivalent to *find the Gaussian mode $p_\sigma(\tilde{\mathbf{x}}|\mathbf{x})$ of standard deviation σ that best approximates the noisy prior distribution $p_\sigma(\tilde{\mathbf{x}})$ in terms of KL divergence.*

Optimization algorithms With SNORE regularization, we solve the following optimization problem to restore an image

$$\arg \min_{\mathbf{x} \in \mathbb{R}^d} \mathcal{J}(\mathbf{x}) := \mathcal{F}(\mathbf{x}, \mathbf{y}) + \alpha \mathcal{R}_\sigma(\mathbf{x}). \tag{9}$$

Due to the formulation of $\mathcal{R}_\sigma$ as an expectation, we implement[1] a stochastic gradient descent algorithm (Algorithm 2) to solve Problem (9).

Algorithm 2 SNORE

1: **Param.:** init. $\mathbf{x}_0 \in \mathbb{R}^d$, $\sigma > 0$, $\alpha > 0$, $\delta > 0$, $N \in \mathbb{N}$
2: **Input:** degraded image $\mathbf{y}$
3: **Output:** restored image $\mathbf{x}_N$
4: **for** $k = 0, 1, \ldots, N - 1$ **do**
5: $\quad \boldsymbol{\epsilon} \leftarrow \mathcal{N}(0, \mathbf{I}_d)$
6: $\quad \tilde{\mathbf{x}}_k \leftarrow \mathbf{x}_k + \sigma\boldsymbol{\epsilon}$
7: $\quad \mathbf{x}_{k+1} \leftarrow \mathbf{x}_k - \delta \nabla \mathcal{F}(\mathbf{x}_k, \mathbf{y}) - \frac{\alpha \delta}{\sigma^2} (\mathbf{x}_k - D_\sigma(\tilde{\mathbf{x}}_k))$
8: **end for**

Algorithm 3 Annealed SNORE

1: **Param.:** init. $\mathbf{x}_0 \in \mathbb{R}^d$, $\delta > 0$, annealing schedule $m \in \mathbb{N}$, $\sigma_0 > \cdots > \sigma_{m-1} \approx 0$, $\alpha_0, \ldots, \alpha_{m-1} > 0$, $N_0, \ldots, N_{m-1} \in \mathbb{N}$
2: **Input:** degraded image $\mathbf{y}$
3: **Output:** restored image $\mathbf{x}_{N_{m-1}}$
4: **for** $i = 0, 1, \ldots, m - 1$ **do**
5: $\quad$ **for** $k = 0, 1, \ldots, N_i - 1$ **do**
6: $\quad \quad \boldsymbol{\epsilon} \leftarrow \mathcal{N}(0, \mathbf{I}_d)$
7: $\quad \quad \tilde{\mathbf{x}}_k \leftarrow \mathbf{x}_k + \sigma_i\boldsymbol{\epsilon}$
8: $\quad \quad \mathbf{x}_{k+1} \leftarrow \mathbf{x}_k - \delta \nabla \mathcal{F}(\mathbf{x}_k, \mathbf{y}) - \frac{\alpha_i \delta}{\sigma_i^2} (\mathbf{x}_k - D_{\sigma_i}(\tilde{\mathbf{x}}_k))$
9: $\quad$ **end for**
10: **end for**

Inspired by annealed importance sampling (Neal, 2001) and the recent use of such a decreasing of σ in diffusion model (Sun et al., 2023), we also develop an Annealed SNORE Algorithm (Algorithm 3). This algorithm, which proves more efficient in practice, is supported by a critical point analysis (Proposition 3.2).

2.3. Related Works

Other Stochastic Plug-and-Play algorithms In the existing literature, stochastic versions of Plug-and-Play have already been proposed. Most of these works intend to accelerate the computation by a stochastic mini-batch approximation on the data-fidelity (Tang & Davies, 2020) or the regularization (Sun et al., 2019a). On the other hand, SNORE

---

[1]Note that a possible stochastic gradient of $\mathcal{R}_\sigma$ can be $\frac{1}{\sigma^2} (\tilde{\mathbf{x}} - D_\sigma(\tilde{\mathbf{x}}))$. We choose to only add noise in the denoiser to reduce the residual noise on the image.

does not aim at accelerating PnP algorithms but it proposes a stochastic improvement of PnP by injecting noise inside the classical PnP regularization.

Laumont et al. (2023) propose to run a stochastic gradient descent algorithm (PnP SGD) with the PnP regularization. Contrary to PnP SGD, SNORE injects the noise inside the denoiser only, and not in the data-fidelity term. Moreover, in SNORE, the standard deviation of the injected noise is fixed (for fixed σ).

Another line of works target image restoration by sampling the posterior law instead of solving Problem 1.. This can be done within a PnP framework using the Tweedie formula (Equation 3) to compute a Langevin dynamic (Laumont et al., 2022; Renaud et al., 2024) or a Gibbs sampling (Coeurdoux et al., 2023; Bouman & Buzzard, 2023).

Link with diffusion based method Denoising diffusion models (DDM) are a class of generative models that can generate images by gradually transforming noise into data with deep denoising networks (Ho et al., 2020; Song et al., 2020). A key feature of DDMs is that, for the adequate weighting schedule w_t, the log-likelihood of the generative model $p_\theta(\mathbf{x})$ is lower-bounded by a (negative) mixture of denoising losses at different noise levels (Ho et al., 2020; Song et al., 2021):

$$\log p_\theta(\mathbf{x}) \geq \underbrace{-\mathbb{E}_{t,\tilde{\mathbf{x}}\sim p_\sigma(\tilde{\mathbf{x}}|\mathbf{x})}\left[w_t\|\mathbf{x}-D_{\sigma_t}(\tilde{\mathbf{x}})\|^2\right]}_{\mathcal{L}(\mathbf{x})} \quad (10)$$

In order to use DDMs for regularizing inverse problems, several works have proposed to replace the intractable log-likelihood by its lower-bound (10) (Poole et al., 2022; Wang et al., 2023; Feng & Bouman, 2023; Mardani et al., 2023). In particular, the gradient of the lower-bound is:

$$\nabla_\mathbf{x}\mathcal{L}(\mathbf{x}) = \mathbb{E}_{t,\tilde{\mathbf{x}}\sim p_\sigma(\tilde{\mathbf{x}}|\mathbf{x})}\left[\frac{w_t}{2}J_{D_{\sigma_t}}^\top(\mathbf{x}-D_{\sigma_t}(\tilde{\mathbf{x}}))^2\right] \quad (11)$$

where $J_{D_{\sigma_t}}$ is the Jacobian matrix of the denoiser with respect to the input. Hence, the gradient of DDMs lower-bound (11) has a similar formulation than the gradient of our regularization function (6), with the difference that it includes the Jacobian of the denoiser, and it is averaged over multiple noise levels σ_t. It has been found that removing the Jacobian in (11) yields better results in practice (Poole et al., 2022). Several works propose theoretical justifications to omit the Jacobian matrix (Wang et al., 2023; Mardani et al., 2023), by assuming that the denoising network provides the exact score of some prior function. Although we rely on a similar assumption, our theoretical analysis in Section 3.3 also covers the case of an imperfect denoiser.

A different approach for solving inverse problem with a DDM prior is to guide the generative process of an unconditional DDM to generate images consistent with an observation $\mathbf{y}$ (Song et al., 2022; Kawar et al., 2022; Chung et al.,

2022; 2023; Song et al., 2023; Luther & Seung, 2023; Zhu et al., 2023). Notice that Luther & Seung (2023) proposed an algorithm similar to Algorithm 3 but do not provide strong theoretical motivation or analysis. Such reverse diffusion processes involve a gradual decrease of the strength of the denoising network, analogous to our annealing procedure. However, those approaches aim at sampling from the posterior distribution of the inverse problem, whereas we adopt a (stochastic) optimization perspective. Despite their impressive practical results, DDM guided rely on heuristics to approximate the intractable likelihood model on noisy data. The impact of the approximation error on the distribution of generated samples remains to be quantified.

3. Convergence Analysis

In this section, we provide a theoretical analysis of our regularization SNORE and a convergence analysis of the associated algorithm. Problem (9) is non-convex due to our regularization. Hence, in the best-case scenario, one can only expect a convergence of the algorithm towards a critical point of the target functional $\mathcal{J}$, defined in relation (9). Note that all global and local minima of $\mathcal{J}$ are critical points of $\mathcal{J}$. Based on the existing literature (Doucet & Tadic, 2017; Laumont et al., 2023), we analyze our stochastic gradient descent in this challenging non-convex context.

We first show (Section 3.1) that our regularization $\mathcal{R}_\sigma$ is a relevant approximation of the ideal regularization $-\log p$. Then we analyze the asymptotic behavior of the critical points of Problem (9) which motivates the annealing Algorithm 3. Next we prove, at fixed σ, the convergence of Algorithm 2 to a critical point of Problem (9) in the case of using the exact MMSE denoiser $D_\sigma^\star$ (Section 3.2). In Section 3.3, we quantify the error of Algorithm 2 with an inexact denoiser D_σ. Proofs can be found in Appendix D. A discussion on technical assumptions is given in Appendix E.

3.1. Asymptotics of critical points when $\sigma \to 0$

Inspired by Laumont et al. (2023, Proposition 1), we first demonstrate that $\nabla\mathcal{R}_\sigma$ converges uniformly to $-\nabla\log p$ on every compact when $\sigma \to 0$. This type of result requires technical assumptions, such as $\nabla\log p$ to be defined everywhere and smooth.

Assumption 3.1. (a) The prior distribution $p \in C^1(\mathbb{R}^d,]0, +\infty[)$ with $\|p\|_\infty + \|\nabla p\|_\infty < +\infty$. **(b)** The prior score is sub-polynomial, there exist $A \in \mathbb{R}^+$ and $q \in \mathbb{N}$ such that $\forall\mathbf{x} \in \mathbb{R}^d, \|\nabla\log p(\mathbf{x})\| \leq A(1 + \|\mathbf{x}\|^q)$.

Assumption 3.1(a) ensures that the prior is smooth, non-degenerate and Lipschitz.

Assumption 3.2. The noisy prior score is sub-polynomial, there exist $B \in \mathbb{R}^+$, $\beta \in \mathbb{R}$ and $r \in \mathbb{N}$ such that $\forall\mathbf{x} \in \mathbb{R}^d$, $\|\nabla\log p_\sigma(\mathbf{x})\| \leq B\sigma^\beta(1 + \|\mathbf{x}\|^r)$.

Under the so-called manifold hypothesis (De Bortoli, 2022) (see Assumption 3.5), Assumption 3.2 is verified with $r = 1$ and $\beta = -2$. Assumption 3.2 has also been proved with $r = 1$ and $\beta = 0$ in (De Bortoli et al., 2021), under the Assumption 3.1(b) with $q = 1$ and the fact that there exist $m_0 > 0$ and $d_0 \geq 0$ such that $\forall \mathbf{x} \in \mathbb{R}^d$, $\langle \nabla \log p(\mathbf{x}), \mathbf{x} \rangle \leq -m_0 \|\mathbf{x}\|^2 + d_0 \|\mathbf{x}\|$.

Proposition 3.1. *Under Assumptions 3.1-3.2, for* $\mathbf{K}$ *a compact of* $\mathbb{R}^d$, $\nabla \mathcal{R}_\sigma$ *converges uniformly to* $-\nabla \log p$ *on* $\mathbf{K}$,

$$\lim_{\sigma \to 0} \sup_{\mathbf{K}} \|\nabla \mathcal{R}_\sigma + \nabla \log p\| = 0.$$

Proposition 3.1 proves that our score $\nabla \mathcal{R}_\sigma$ is close to the ideal score $-\nabla \log p$ when $\sigma \to 0$. With this uniform approximation result, we are now able to study the behavior of the critical points of our optimization problem when $\sigma \to 0$. For $\mathbf{K}$ a compact of $\mathbb{R}^d$, we define $\mathbf{S}_{\sigma,\mathbf{K}} = \{\mathbf{x} \in \mathbf{K}|\nabla \mathcal{F}(\mathbf{x}, \mathbf{y}) + \alpha \nabla \mathcal{R}_\sigma(\mathbf{x}) = 0\}$, the set of critical points of Problem (9) in $\mathbf{K}$. In order to study the behavior of the critical points set $\mathbf{S}_{\sigma,\mathbf{K}}$ when $\sigma \to 0$, we define below the notion of limit for sets $\mathbf{S}_{\sigma,\mathbf{K}}$ when $\sigma \to 0$. To do so, we first introduce cluster points of sets.

Definition. For a sequence of sets $(\mathbf{S}_k)_{k \in \mathbb{N}} \in (\mathbb{R}^d)^{\mathbb{N}}$, $\mathbf{z}$ is called a cluster point of these sets if any neighborhood of $\mathbf{z}$ is visited infinitely often by $(\mathbf{S}_k)$, i.e. $\forall \epsilon > 0$, $\forall k_0 \in \mathbb{N}$, there exist $k \geq k_0$, and $\mathbf{x}_k \in \mathbf{S}_k$ such that $\|\mathbf{x}_k - \mathbf{z}\| \leq \epsilon$.

We now apply this definition of cluster points of sets for a decreasing sequence of $\sigma > 0$. For $\mathbf{E} = \{(\sigma_n)_{n \in \mathbb{N}} \in (\mathbb{R})^{\mathbb{N}} | \forall n \in \mathbb{N}, \sigma_n > 0, \sigma_n$ decreases to $0\}$, and for $\boldsymbol{\sigma} = (\sigma_n)_{n \in \mathbb{N}} \in \mathbf{E}$, we define the cluster points of the sequence of set $(\mathbf{S}_{\sigma_n,\mathbf{K}})_{n \in \mathbb{N}}$ by $\mathbf{S}_{\boldsymbol{\sigma},\mathbf{K}} = \{\mathbf{x} \in \mathbf{K}|\forall \epsilon > 0, \forall m \in \mathbb{N}, \exists n \geq m, \mathbf{z}_n \in \mathbf{S}_{\sigma_n,\mathbf{K}}, \|\mathbf{x} - \mathbf{z}_n\| \leq \epsilon\}$. Finally, we can define a limit of sets $\mathbf{S}_{\sigma,\mathbf{K}}$ for a continuous $\sigma \to 0$ with $\mathbf{S}_\mathbf{K} = \cup_{\boldsymbol{\sigma} \in \mathbf{E}} \mathbf{S}_{\boldsymbol{\sigma},\mathbf{K}}$. Our target points are the critical points of Problem (1), $\mathbf{S}_\mathbf{K}^\star = \{\mathbf{x} \in \mathbf{K}|\nabla \mathcal{F}(\mathbf{x}) + \alpha \nabla \mathcal{R}(\mathbf{x}) = 0\}$ where $\mathcal{R} = -\log p$. The following proposition finally establishes that the limit of set $\mathbf{S}_{\sigma,\mathbf{K}}$ (in the sense of cluster point) is included in the targeted points.

Proposition 3.2. *Under Assumptions 3.1-3.2, for* $\mathbf{K}$ *a compact subset of* $\mathbb{R}^d$, *we have*

$$\mathbf{S}_\mathbf{K} \subseteq \mathbf{S}_\mathbf{K}^\star.$$

Proposition 3.2 means that a sequence of computed critical points with $\sigma > 0$ has all its cluster points in $\mathbf{S}_\mathbf{K}^\star$, the set of critical points of the ideal optimization problem (1). This result suggests that the annealed algorithm may converge to a point of $\mathbf{S}_\mathbf{K}^\star$. In fact annealing (Neal, 2001) consists in successively approximating critical points of $\mathbf{S}_{\sigma_i,\mathbf{K}}$ for a decreasing sequence $\sigma_0 > \cdots > \sigma_{m-1} \approx 0$. Proposition 3.2 thus motivates Algorithm 3, which will be proved efficient in practice.

3.2. Unbiased algorithm analysis

In this section, we prove the convergence of the SNORE Algorithm 2 run with the exact MMSE denoiser $D_\sigma^\star$. With this denoiser, an iteration of the algorithm is computed by

$$\mathbf{x}_{k+1} = \mathbf{x}_k - \delta_k \nabla \mathcal{F}(\mathbf{x}_k, \mathbf{y}) - \alpha \delta_k \nabla \log p_\sigma(\mathbf{x}_k + \zeta_k), \tag{12}$$

with $\zeta_k \sim \mathcal{N}(0, \sigma^2 \mathbf{I}_d)$ and $(\delta_k)_{k \in \mathbb{N}} \in (\mathbb{R}^+)^{\mathbb{N}}$ the decreasing sequence of step-sizes. Algorithm 2 is a stochastic descent algorithm that solves Problem (9).

One can note that the stochastic gradient estimation is unbiased. Indeed, by defining

$$f(\mathbf{x}, \zeta) = \nabla \mathcal{F}(\mathbf{x}, \mathbf{y}) + \alpha \nabla \log p_\sigma(\mathbf{x} + \zeta),$$

we verify that

$$\mathbb{E}_{\zeta \sim \mathcal{N}(0, \sigma^2 \mathbf{I}_d)}(f(\mathbf{x}, \zeta)) = \nabla \mathcal{F}(\mathbf{x}, \mathbf{y}) + \alpha \nabla \mathcal{R}_\sigma(\mathbf{x}).$$

We make our convergence analysis based on previous studies on stochastic gradient algorithm (Benaïm, 1999, Corollary 6.7), (Metivier & Priouret, 1984, Section II-D) or (Doucet & Tadic, 2017). An assumption on the step-size decrease is required to ensure convergence.

Assumption 3.3. *The step-size decreases to zero but not too fast:* $\sum_{k=0}^{+\infty} \delta_k = +\infty$ *and* $\sum_{k=0}^{+\infty} \delta_k^2 < +\infty$.

Assumption 3.3 guides the choice of the step-size rule to ensure convergence, for instance $\delta_k = \frac{\delta}{k^a}$ with $a \in]\frac{1}{2}, 1]$.

Assumption 3.4. *The data-fidelity term* $\mathcal{F}_y : \mathbf{x} \in \mathbb{R}^d \mapsto \mathcal{F}(\mathbf{x}, \mathbf{y})$ *is* $\mathcal{C}^\infty$.

This assumption is typically verified for a data-fidelity term $\mathcal{F}(\mathbf{x}, \mathbf{y}) = \frac{1}{2\sigma_y^2}\|\mathbf{y} - \mathcal{A}\mathbf{x}\|^2$ associated to a linear inverse problem with additive white Gaussian noise.

We define the set of realizations where the sequence is bounded in the compact $\mathbf{K}$ by

$$\Lambda_\mathbf{K} = \bigcap_{k \in \mathbb{N}} \{\mathbf{x}_k \in \mathbf{K}\},$$

and the distance of a point to a set by $d(\mathbf{x}, \mathbf{S}) = \inf_{\mathbf{y} \in \mathbf{S}} \|\mathbf{x} - \mathbf{y}\|$, with $\mathbf{x} \in \mathbb{R}^d$ and $\mathbf{S} \subset \mathbb{R}^d$. The fact that we restrict to realizations in $\Lambda_\mathbf{K}$ will be referred to as the "boundedness assumption".

Proposition 3.3. *Under Assumptions 3.2-3.4, almost surely on* $\Lambda_\mathbf{K}$, *we have*

$$\lim_{k \to +\infty} d(\mathbf{x}_k, \mathbf{S}_{\sigma,\mathbf{K}}) = 0,$$

$$\lim_{k \to +\infty} \|\nabla \mathcal{J}(\mathbf{x}_k)\| = 0,$$

and $(\mathcal{J}(\mathbf{x}_k))_{k \in \mathbb{N}}$ *converges to a value of* $\mathcal{J}(\mathbf{S}_{\sigma,\mathbf{K}})$.

Proposition 3.3 proves that Algorithm 2 run with the exact MMSE denoiser (4) converges to the set of critical points of Problem (9). This is a weak convergence in the sense that this does not give *a priori* that there is $\mathbf{x}^\star \in \mathbf{S}_{\sigma,\mathbf{K}}$ such that $\lim_{k\to+\infty} \|\mathbf{x}_k - \mathbf{x}^\star\| = 0$. Assuming that the sequence is bounded in $\Lambda_\mathbf{K}$ is standard in the stochastic gradient descent analysis (Benaïm, 2006; Castera et al., 2021). We discuss this assumption in Appendix H.

In the previous result, we do not assume that the prior p is smooth but we make Assumption 3.2 of a subpolynomial noisy score. This assumption is difficult to verify for a general prior distribution but can be verified in the case of the so-called manifold hypothesis (De Bortoli, 2022).

Assumption 3.5 (Manifold hypothesis). The prior p is supported on a compact set $\mathcal{M}$.

Assumption 3.5 is typically true for an image distribution with bounded pixel values.

Proposition 3.4. *Under Assumptions 3.3-3.5, almost surely on $\Lambda_\mathbf{K}$, we have*

$$\lim_{k\to+\infty} d(\mathbf{x}_k, \mathbf{S}_{\sigma,\mathbf{K}}) = 0,$$
$$\lim_{k\to+\infty} \|\nabla\mathcal{J}(\mathbf{x}_k)\| = 0,$$

and $(\mathcal{J}(\mathbf{x}_k))_{k\in\mathbb{N}}$ converges to a value of $\mathcal{J}(\mathbf{S}_{\sigma,\mathbf{K}})$.

Proposition 3.4 establishes the convergence of the SNORE algorithm to the critical points of Problem (9), under the three mild Assumptions 3.3-3.5.

3.3. Biased algorithm analysis

We now quantify the error of the SNORE algorithm (Algorithm 2) run with an inexact MMSE denoiser $D_\sigma \approx D_\sigma^\star$. Such a study is crucial as the algorithm is run in practice with a learned denoiser D_σ which is not exact. With this denoiser, an iteration of Algorithm 2 is computed by

$$\mathbf{x}_{k+1} = \mathbf{x}_k - \delta_k \nabla\mathcal{F}(\mathbf{x}_k, \mathbf{y}) - \frac{\alpha\delta_k}{\sigma^2}\left(\mathbf{x}_k - D_\sigma(\mathbf{x}_k + \zeta_k)\right). \tag{13}$$

It can be rewritten as

$$\mathbf{x}_{k+1} = \mathbf{x}_k - \delta_k(\nabla\mathcal{J}(x_k) + \xi_k), \tag{14}$$

where, by using Equation (6), the gradient perturbation writes as

$$\xi_k = \nabla\mathcal{F}(\mathbf{x}_k, \mathbf{y}) + \frac{\alpha}{\sigma^2}\left(\mathbf{x}_k - D_\sigma(\mathbf{x}_k + \zeta_k)\right) - \nabla\mathcal{J}(x_k)$$
$$= \frac{\alpha}{\sigma^2}\left(D_\sigma(\mathbf{x}_k + \zeta_k) - D_\sigma^*(x_k)\right).$$

This stochastic shift ξ_k is in general biased, i.e. $\mathbb{E}(\xi_k) \neq 0$.

Assumption 3.6. The learned denoiser D_σ is $\mathcal{C}^\infty$ and is a bounded approximation of the exact MMSE denoiser $D_\sigma^\star$, $\forall R > 0$, there exists $M(R) > 0$, such that $\forall \mathbf{x} \in \mathcal{B}(0, R)$, $\|D_\sigma(\mathbf{x}) - D_\sigma^\star(\mathbf{x})\| \leq M(R)$.

Assumption 3.6 is also made in (Laumont et al., 2023), and it can be ensured if the denoiser is learned with a specific loss (Laumont et al., 2022). Moreover, if the activation functions of the denoiser are $\mathcal{C}^\infty$ (in our case ELU), then the denoiser is $\mathcal{C}^\infty$.

Assumption 3.7. The exact MMSE denoiser and the learned denoiser are sublinear, there exists $C \geq 0$ such that $\forall \mathbf{x} \in \mathbb{R}^d$, $\|D_\sigma^\star(\mathbf{x})\| \leq \|\mathbf{x}\| + C\sigma$ and $\|D_\sigma(\mathbf{x})\| \leq \|\mathbf{x}\| + C\sigma$.

Assumption 3.7 is the stable condition on the denoiser D_σ in the sense that it is bounded in norm. As a practical example, a bounded denoiser (Chan et al., 2016, Definition 1) verifies Assumption 3.7.

Proposition 3.5. *Under Assumptions 3.3, 3.4, 3.6, 3.7, for $R > 0$ and $\mathbf{K} \subseteq \mathcal{B}(0, R)$ compact, almost surely on $\Lambda_\mathbf{K}$, there exists $M_\mathbf{K} \in]0, +\infty[$ such that*

$$\limsup_{k\to+\infty} \|\nabla\mathcal{J}(\mathbf{x}_k)\| \leq M_\mathbf{K}\eta^{\frac{1}{2}},$$
$$\limsup_{k\to+\infty} \mathcal{J}(\mathbf{x}_k) - \liminf_{k\to+\infty} \mathcal{J}(\mathbf{x}_k) \leq M_\mathbf{K}\eta,$$

with the bias $\eta = \limsup_{k\to+\infty}\|\mathbb{E}(\xi_k)\| \underset{\sigma\to 0}{\leq} \frac{\alpha}{\sigma^2}M(R) + o(\sigma)$.

The denoiser bias η has a similar bound than in (Laumont et al., 2023, Proposition 3). If the denoiser is well trained, $M(R) \approx 0$, i.e. the denoiser bias is small. Proposition 3.5 proves that the smaller is the denoiser bias, the closer the sequence $(x_k)_{k\in\mathbf{N}}$ is to critical points of Problem (9) (in terms of gradient norm). This statement generalizes the convergence result of Proposition 3.3 to the biased case.

4. Experiments

In this section, we show the performance of Annealing SNORE (Ann-SNORE) algorithm for image inverse problems, including deblurring and inpainting. Ann-SNORE is compared to several state-of-the-art image restoration methods. In Appendix F we give more details on our experiments and other results on various images and various inverse problems including deblurring, inpainting, super-resolution and despeckling. A study of Ann-SNORE sensitivity to its parameters and the randomness of the algorithm is also provided. The code used in these experiments can be found in https://github.com/Marien-RENAUD/SNORE.

In our experiment, we use a Gradient-Step denoiser (Hurault et al., 2022a) of the form

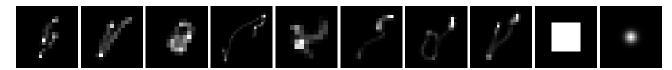$$D_\sigma = \mathrm{Id} - \nabla g_\sigma, \tag{15}$$

Figure 1. Kernels used for deblurring. As in (Zhang et al., 2017; 2021; Pesquet et al., 2021; Hurault et al., 2022a) we test the different methods on 8 real-world camera shake kernels proposed in (Levin et al., 2009) and on the uniform 9×9 kernel and the 25×25 Gaussian kernel with standard deviation 1.6 proposed in (Romano et al., 2017).
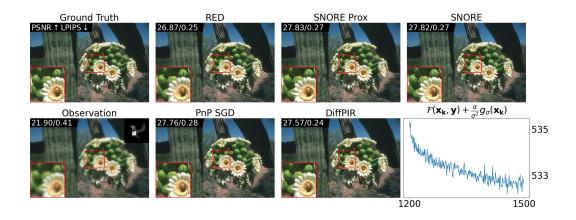

Figure 2. Deblurring with various methods of a motion blur kernel with input noise level $\sigma_{\mathbf{y}} = 10/255$ with a GS-denoiser trained on natural images. Note that Ann-SNORE produces a better perceptual reconstruction (BRISQUE). *Bottom-Right:* Decrease of the optimized function $\mathcal{J}$ (Equation (16)) along the stochastic gradient descent with the last parameters $(\alpha_{m-1}, \sigma_{m-1})$ of the annealing procedure.

where g_σ is a learned neural network. With this gradient-step denoiser, Hurault et al. (2022a) demonstrated that RED converges to a critical point of an explicit objective function of the form $\mathcal{F}(\mathbf{x}, \mathbf{y}) + \alpha g_\sigma(\mathbf{x})$. Furthermore, it is established that this objective function decreases throughout the algorithm. Using the Gradient-Step denoiser in the SNORE Algorithm 2 yields a stochastic gradient descent that minimizes the objective

$$\underset{\mathbf{x} \in \mathbb{R}^d}{\arg\min} \, \mathcal{J}(\mathbf{x}) := \mathbb{E}_{\tilde{\mathbf{x}} \sim p_\sigma(\tilde{\mathbf{x}}|\mathbf{x})} \left(\mathcal{F}(\mathbf{x}, \mathbf{y}) + \frac{\alpha}{\sigma^2} g_\sigma(\tilde{\mathbf{x}}) \right). \quad (16)$$

4.1. Deblurring

For image deblurring, the degradation operator is a convolution performed with circular boundary conditions. Therefore $\mathcal{F}(\mathbf{x}) = \frac{1}{2\sigma_{\mathbf{y}}^2} \|\mathbf{y} - \mathbf{A}\mathbf{x}\|^2$, where $\mathbf{A} = \mathbf{F}\boldsymbol{\Lambda}\mathbf{F}^*$, $\mathbf{F}$ is the orthogonal matrix of the discrete Fourier transform (and $\mathbf{F}^*$ its inverse) and $\boldsymbol{\Lambda}$ a diagonal matrix.

We make a gradual annealing, by keeping σ and α fixed for some iterations, to efficiently minimize Problem (16). We set 1500 iterations and $m = 16$ annealing levels. To ensure convergence, we run 300 iterations with the last parameters $\sigma_{m-1}, \alpha_{m-1}$. For all input noise levels $\sigma_{\mathbf{y}}$, we set $\sigma_0 = 1.8\sigma_{\mathbf{y}}$, $\sigma_{m-1} = 0.5\sigma_{\mathbf{y}}$, $\alpha_0 = 0.1\sigma_0^2\sigma_{\mathbf{y}}^{-2}$ and $\alpha_{m-1} = \sigma_{m-1}^2\sigma_{\mathbf{y}}^{-2}$. We initialize with the observation

$\mathbf{x}_0 = \mathbf{y}$ and use a fixed step-size, $\delta_k = \delta = 0.1$, as we observe that a decreasing δ_k leads to a slower convergence, as notice by Laumont et al. (2023).

We compare in Table 1 our method to RED restoration algorithm of (Romano et al., 2017) (see Algorithm 1) with a gradient-descent step on the data-fidelity, "RED Prox" method (Romano et al., 2017) with a proximal-descent step on the data-fidelity (see Algorithm 4), DiffPIR (Zhu et al., 2023) and PnP SGD (Laumont et al., 2023). As (Hurault et al., 2022a), we evaluate each method on 10 images from CBSD68 (Martin et al., 2001) and 10 blur kernels presented in Figure 1. The same denoiser trained by (Hurault et al., 2022a) on natural images is used for all methods to ensure a fair comparison. We compare two variants of the Ann-SNORE algorithm, with gradient step on the data-fidelity (Algorithm 3) or with a proximal step (Algorithm 5 in Appendix F). DiffPIR is used with $t_{\text{start}} < t_{\text{trained}}$ as suggested by the authors (Zhu et al., 2023). However, this diffusion method does not outperform in our experiments because our denoiser is not trained to tackle highly-noised images. An extra-parameter $\beta > 0$ is added to PnP SGD algorithm to increase the algorithm performances. On Table 1, results are presented with distorsion metrics (PSNR, SSIM) and perceptual metrics with reference (LPIPS) and without reference (BRISQUE).
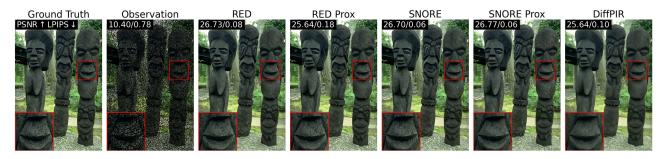
7

Figure 3. Inpainting with various methods on a random mask (with a proportion $p = 0.5$ of masked pixels) with a GS-denoiser trained on natural images. One can observe here the ability of Ann-SNORE to recover both sharp structures and textural content.

Noise level	Method	PSNR↑	SSIM↑	LPIPS↓	BRISQUE↓
	RED	29.82	0.84	0.17	<u>21.09</u>
	RED Prox	**30.64**	**0.87**	**0.15**	45.27
5/255	Ann-SNORE	<u>29.92</u>	<u>0.85</u>	0.17	28.24
	Ann-SNORE Prox	<u>29.92</u>	<u>0.85</u>	0.17	28.25
	DiffPIR	28.55	0.76	<u>0.16</u>	**17.87**
	PnP SGD	29.37	0.83	0.20	26.44
	RED	27.18	0.72	0.25	**20.16**
	RED Prox	**28.50**	**0.80**	0.23	51.41
10/255	Ann-SNORE	<u>27.91</u>	<u>0.78</u>	0.24	27.89
	Ann-SNORE Prox	<u>27.91</u>	<u>0.78</u>	**0.23**	27.97
	DiffPIR	27.47	0.74	0.24	<u>21.12</u>
	PnP SGD	27.61	0.75	0.26	25.74
	RED	24.03	0.54	0.43	**20.75**
	RED Prox	**26.31**	**0.71**	**0.31**	55.48
20/255	Ann-SNORE	25.61	0.66	0.32	28.88
	Ann-SNORE Prox	25.61	0.66	<u>0.32</u>	29.00
	DiffPIR	<u>26.05</u>	<u>0.69</u>	0.33	33.57
	PnP SGD	25.50	0.63	0.33	<u>22.67</u>

Table 1. Quantitative comparisons of image deblurring methods on CBSD10 with 10 different blur kernels (see Figure 1) and three different level of noise $\sigma_\mathbf{y} \in \{5, 10, 20\}/255$. Best and second best results are respectively displayed in bold and underlined.

On Table 1, we observe that Ann-SNORE has similar performance than other state-of-the-art methods. If we compare Ann-SNORE to RED Prox, which performs the best in terms of distortion, we observe that Ann-SNORE performs favorably in terms of perceptual metrics. However, Ann-SNORE remains slower than other methods for deblurring as it requires a sufficient number of annealing levels for the restored images to have a high visual quality (see discussion in Appendix F).

On Figure 2, we provide a qualitative comparison. Note that the Ann-SNORE algorithm provides a more realistic result than RED Prox with equivalent quantitative score (PSNR). The global decreasing behavior of the function $\mathcal{J}$ (16) empirically confirms that our algorithm minimizes this function.

4.2. Inpainting

For image inpainting, the degradation operator $\mathbf{A}$ is a diagonal matrix with coefficient in $\{0, 1\}$. No noise is added to the degraded observation, $\mathbf{y} = \mathbf{A}\mathbf{x}$. The proximal op-

Method	PSNR↑	SSIM↑	LPIPS↓	BRISQUE↓
RED	31.26	0.91	<u>0.07</u>	17.13
RED Prox	30.31	0.89	0.12	38.29
Ann-SNORE	<u>31.65</u>	<u>0.92</u>	**0.04**	<u>7.10</u>
Ann-SNORE Prox	**31.94**	**0.93**	**0.04**	8.35
DiffPIR	29.57	0.87	<u>0.07</u>	**4.17**

Table 2. Inpainting result for random missing pixel with probability $p = 0.5$ on CBSD68 dataset. Best and second best results are respectively displayed in bold and underlined.

erator of $\mathcal{F}$ is the orthogonal projection which imposes observed pixels values. This strict condition is relaxed in RED (Algorithm 1) and Ann-SNORE (Algorithm 3) by taking $\nabla\mathcal{F}(\mathbf{x}, \mathbf{y}) = \mathbf{A}(\mathbf{x} - \mathbf{y})$. We focus on random mask inpainting with a proportion of masked pixels $p = 0.5$.

For Ann-SNORE, we keep the same annealing scheme than for deblurring with 500 iterations. We set $\sigma_0 = 50/255$, $\sigma_{m-1} = 5/255$, $\alpha_0 = \alpha_{m-1} = 0.15$. The initialization is done with a modified version of the observation $\mathbf{y}$, where the masked pixels are set to the 0.5 value. We set a fixed step-size $\delta_k = \delta = 0.5$.

On Table 2, we compare Ann-SNORE to other methods for the inpainting task with $p = 0.5$ on CBSD68. A qualitative comparison is given in Figure 3. We observe that Ann-SNORE outperforms other approaches with better distortion and perceptual scores (PSNR, SSIM, LPIPS). Visually, we observe that Ann-SNORE succeeds to restore textures. Other qualitative results are given in Appendix F. Note that each method (except DiffPIR) is run with the same number of iterations. Hence, contrary to the deblurring task, inpainting with Ann-SNORE does not involve any additional computational load with respect to RED methods.

5. Conclusion

In this work, we introduce a stochastic denoising regularization (SNORE) for solving imaging inverse problems within the PnP framework. This regularization realizes the heuristic idea that *an image looks clean if its noisy versions look*

as noisy images (with the same noise level). Solving inverse problems with this regularization can be addressed with a provably-convergent stochastic optimization algorithm. The algorithm differs from standard PnP in the fact that the regularization step consists in denoising noised images, thus avoiding a distribution-shift from the data on which the denoiser is trained. It also draws a connection with recent diffusion-based approaches, which also involve noising-denoising steps, but included in a different global scheme that relates to backward diffusion. Experiments conducted on ill-posed inverse problems (deblurring, inpainting) show that SNORE attains state-of-the-art image restoration performance (in terms of full-reference and no-reference quality measures), at the expense of a computational cost which is, for now, larger than competing methods for deblurring. It would be interesting to determine whether the computational cost could be reduced by relying on other stochastic gradient-descent algorithms (e.g. ADAM (Kingma & Ba, 2014), INNA (Castera et al., 2021)), for which theoretical convergence has not been shown yet for functionals including regularizations such as RED or SNORE.

Impact Statement

The work presented in this paper addresses the highly ill-posed problem of restoring missing information within an image. This sensitivity to errors is particularly pronounced in scenarios involving post-processing algorithms such as segmentation, detection, or classification applied to the reconstructed image, where errors in the reconstruction process may propagate into erroneous decision-making based on the image data. This concern is particularly critical in the context of medical images. SNORE, functioning as a stochastic process reliant on a learned denoiser, inherently produces random fluctuations in its output. In Section F.2, we provide an analysis of SNORE uncertainty to its random seed and its initialization. These experiments suggest a robustness of our method. Subsequent research effort should focus on quantifying the errors associated with SNORE in order to confirm its utility as a reliable reconstruction algorithm.

References

Agustsson, E. and Timofte, R. Ntire 2017 challenge on single image super-resolution: Dataset and study. In *2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pp. 1122–1131, 2017. doi: 10.1109/CVPRW.2017.150.

Benaïm, M. Dynamics of stochastic approximation algorithms. In Azéma, J., Émery, M., Ledoux, M., and Yor, M. (eds.), *Séminaire de Probabilités XXXIII*, pp. 1–68, Berlin, Heidelberg, 1999. Springer Berlin Heidelberg. ISBN 978-3-540-48407-3.

Benaïm, M. Dynamics of stochastic approximation algorithms. In *Seminaire de probabilites XXXIII*, pp. 1–68. Springer, 2006.

Bouman, C. A. and Buzzard, G. T. Generative plug and play: Posterior sampling for inverse problems. In *2023 59th Annual Allerton Conference on Communication, Control, and Computing (Allerton)*, pp. 1–7, 2023. doi: 10.1109/ Allerton58177.2023.10313413.

Boyd, S., Parikh, N., Chu, E., Peleato, B., and Eckstein, J. *Distributed Optimization and Statistical Learning via the Alternating Direction Method of Multipliers*. Now Foundations and Trends, 2011.

Brown, B. C., Caterini, A. L., Ross, B. L., Cresswell, J. C., and Loaiza-Ganem, G. Verifying the union of manifolds hypothesis for image data. *arXiv preprint arXiv:2207.02862*, 2022.

Castera, C., Bolte, J., Févotte, C., and Pauwels, E. An inertial newton algorithm for deep learning. *Journal of Machine Learning Research*, 22(134):1–31, 2021.

Chan, R. H., Ho, C.-W., and Nikolova, M. Salt-and-pepper noise removal by median-type noise detectors and detail-preserving regularization. *IEEE Transactions on image processing*, 14(10):1479–1485, 2005.

Chan, S. H., Wang, X., and Elgendy, O. A. Plug-and-play admm for image restoration: Fixed-point convergence and applications. *IEEE Transactions on Computational Imaging*, 3(1):84–98, 2016.

Chan, T. F., Esedoglu, S., and Nikolova, M. Algorithms for finding global minimizers of image segmentation and denoising models. *SIAM Journal on Applied Mathematics*, 66(5):1632–1648, 2006.

Chung, H., Sim, B., Ryu, D., and Ye, J. C. Improving diffusion models for inverse problems using manifold constraints. *Advances in Neural Information Processing Systems*, 35:25683–25696, 2022.

Chung, H., Kim, J., Mccann, M. T., Klasky, M. L., and Ye, J. C. Diffusion posterior sampling for general noisy inverse problems. In *The Eleventh International Conference on Learning Representations*, 2023.

Coeurdoux, F., Dobigeon, N., and Chainais, P. Plug-and-play split gibbs sampler: embedding deep generative priors in bayesian inference. *arXiv preprint arXiv:2304.11134*, 2023.

Cohen, R., Elad, M., and Milanfar, P. Regularization by denoising via fixed-point projection (red-pro). *SIAM Journal on Imaging Sciences*, 14(3):1374–1406, 2021.

Dalsasso, E., Yang, X., Denis, L., Tupin, F., and Yang, W. Sar image despeckling by deep neural networks: from a pre-trained model to an end-to-end training strategy. *Remote Sensing*, 12(16):2636, August 2020. ISSN 2072-4292. doi: 10.3390/rs12162636. URL http://dx.doi.org/10.3390/rs12162636.

Davis, D. and Drusvyatskiy, D. Stochastic subgradient method converges at the rate $o(k^{-1/4})$ on weakly convex functions. *arXiv preprint arXiv:1802.02988*, 2018.

De Bortoli, V. Convergence of denoising diffusion models under the manifold hypothesis. *Transactions on Machine Learning Research*, 2022. ISSN 2835-8856. URL https://openreview.net/forum?id=MhK5aXo3gB. Expert Certification.

De Bortoli, V., Thornton, J., Heng, J., and Doucet, A. Diffusion schrödinger bridge with applications to score-based generative modeling. *Advances in Neural Information Processing Systems*, 34:17695–17709, 2021.

Deledalle, C.-A., Denis, L., Tabti, S., and Tupin, F. Mulog, or how to apply gaussian denoisers to multi-channel sar speckle reduction? *IEEE Transactions on Image Processing*, 26(9):4389–4403, 2017.

Donoho, D. L. and Johnstone, I. M. Ideal spatial adaptation by wavelet shrinkage. *Biometrika*, 81:425–455, 1994.

Doucet, A. and Tadic, V. Asymptotic bias of stochastic gradient search. *Annals of Applied Probability*, 27(6), 2017.

Efron, B. Tweedie's formula and selection bias. *Journal of the American Statistical Association*, 106(496):1602–1614, 2011.

Fefferman, C., Mitter, S., and Narayanan, H. Testing the manifold hypothesis. *Journal of the American Mathematical Society*, 29(4):983–1049, 2016.

Feng, B. and Bouman, K. Efficient bayesian computational imaging with a surrogate score-based prior. In *NeurIPS 2023 Workshop on Deep Learning and Inverse Problems*, 2023.

Fermanian, R., Le Pendu, M., and Guillemot, C. PnP-ReG: Learned regularizing gradient for plug-and-play gradient descent. *SIAM Journal on Imaging Sciences*, 16(2):585–613, 2023.

Gavaskar, R. G. and Chaudhury, K. N. On the proof of fixed-point convergence for plug-and-play admm. *IEEE Signal Processing Letters*, 26(12):1817–1821, 2019.

Ghadimi, S. and Lan, G. Stochastic first-and zeroth-order methods for nonconvex stochastic programming. *SIAM journal on optimization*, 23(4):2341–2368, 2013.

Goodman, J. W. Some fundamental properties of speckle. *Journal of the Optical Society of America*, 66:1145–1150, 1976.

Ho, J., Jain, A., and Abbeel, P. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020.

Huang, T., Dong, W., Xie, X., Shi, G., and Bai, X. Mixed noise removal via laplacian scale mixture modeling and nonlocal low-rank approximation. *IEEE Transactions on Image Processing*, 26(7):3171–3186, 2017.

Hurault, S., Leclaire, A., and Papadakis, N. Gradient step denoiser for convergent plug-and-play. In *International Conference on Learning Representations*, 2022a.

Hurault, S., Leclaire, A., and Papadakis, N. Proximal denoiser for convergent plug-and-play optimization with nonconvex regularization. In *International Conference on Machine Learning*, pp. 9483–9505. PMLR, 2022b.

Jalal, A., Arvinte, M., Daras, G., Price, E., Dimakis, A. G., and Tamir, J. Robust compressed sensing mri with deep generative priors. *Advances in Neural Information Processing Systems*, 34:14938–14954, 2021.

Kadkhodaie, Z. and Simoncelli, E. P. Solving linear inverse problems using the prior implicit in a denoiser. *arXiv preprint arXiv:2007.13640*, 2020.

Kawar, B., Elad, M., Ermon, S., and Song, J. Denoising diffusion restoration models. *Advances in Neural Information Processing Systems*, 35:23593–23606, 2022.

Kingma, D. P. and Ba, J. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

Laumont, R., De Bortoli, V., Almansa, A., Delon, J., Durmus, A., and Pereyra, M. Bayesian imaging using plug and play priors: When langevin meets tweedie. *SIAM Journal on Imaging Sciences*, 15(2):701–737, 2022.

Laumont, R., De Bortoli, V., Almansa, A., Delon, J., Durmus, A., and Pereyra, M. On maximum-a-posteriori estimation with plug & play priors and stochastic gradient descent. *Journal of Mathematical Imaging and Vision*, 65:140–163, 2023.

Lehtinen, J., Munkberg, J., Hasselgren, J., Laine, S., Karras, T., Aittala, M., and Aila, T. Noise2Noise: Learning image restoration without clean data. In Dy, J. and Krause, A. (eds.), *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pp. 2965–2974. PMLR, 2018.

Levin, A., Weiss, Y., Durand, F., and Freeman, W. T. Understanding and evaluating blind deconvolution algorithms. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1964–1971, 2009.

Lim, B., Son, S., Kim, H., Nah, S., and Mu Lee, K. Enhanced deep residual networks for single image super-resolution. In *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, pp. 136–144, 2017.

Lunz, S., Öktem, O., and Schönlieb, C.-B. Adversarial regularizers in inverse problems. *Advances in neural information processing systems*, 31, 2018.

Luther, K. and Seung, H. S. Ddgm: Solving inverse problems by diffusive denoising of gradient-based minimization. *arXiv preprint arXiv:2307.04946*, 2023.

Ma, K., Duanmu, Z., Wu, Q., Wang, Z., Yong, H., Li, H., and Zhang, L. Waterloo Exploration Database: New challenges for image quality assessment models. *IEEE Transactions on Image Processing*, 26(2):1004–1016, Feb. 2017.

Mallat, S. *A Wavelet Tour of Signal Processing*. ScienceDirect, 1999.

Mardani, M., Song, J., Kautz, J., and Vahdat, A. A variational perspective on solving inverse problems with diffusion models. *arXiv preprint arXiv:2305.04391*, 2023.

Martin, D., Fowlkes, C., Tal, D., and Malik, J. A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. In *IEEE International Conference on Computer Vision*, volume 2, pp. 416–423 vol.2, 2001.

Metivier, M. and Priouret, P. Applications of a kushner and clark lemma to general classes of stochastic algorithms. *IEEE Transactions on Information Theory*, 30(2):140–151, 1984.

Metzler, C. A., Schniter, P., Veeraraghavan, A., and Baraniuk, R. G. prDeep: Robust phase retrieval with a flexible deep network. In *Proc. 36th Int. Conf. Mach. Learn.*, pp. 3501–3510, 2018.

Mittal, A., Moorthy, A. K., and Bovik, A. C. No-reference image quality assessment in the spatial domain. *IEEE Transactions on Image Processing*, 21(12):4695–4708, 2012.

Neal, R. M. Annealed importance sampling. *Statistics and computing*, 11:125–139, 2001.

Nikolova, M. A variational approach to remove outliers and impulse noise. *Journal of Mathematical Imaging and Vision*, 20, 2004.

Nurminski, E. The quasigradient method for the solving of the nonlinear programming problems. *Cybernetics*, 9, 1973.

Pesquet, J.-C., Repetti, A., Terris, M., and Wiaux, Y. Learning maximally monotone operators for image recovery. *SIAM Journal on Imaging Sciences*, 14(3):1206–1237, 2021.

Poole, B., Jain, A., Barron, J. T., and Mildenhall, B. Dreamfusion: Text-to-3d using 2d diffusion. *arXiv preprint arXiv:2209.14988*, 2022.

Reehorst, E. T. and Schniter, P. Regularization by denoising: Clarifications and new interpretations. *IEEE Transactions on Computational Imaging*, 5(1):52–67, 2019. doi: 10.1109/TCI.2018.2880326.

Ren, M., Delbracio, M., Talebi, H., Gerig, G., and Milanfar, P. Multiscale structure guided diffusion for image deblurring. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 10721–10733, 2023.

Renaud, M., Liu, J., Bortoli, V. D., Almansa, A., and Kamilov, U. Plug-and-play posterior sampling under mismatched measurement and prior models. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=66arKkGiFy.

Romano, Y., Elad, M., and Milanfar, P. The little engine that could: Regularization by denoising (RED). *SIAM Journal on Imaging Sciences*, 10(4):1804–1844, 2017.

Rudin, L. I., Osher, S., and Fatemi, E. Nonlinear total variation based noise removal algorithms. *Physica D: Nonlinear Phenomena*, 60(1):259–268, 1992. ISSN 0167-2789.

Ryu, E. K., Liu, J., Wang, S., Chen, X., Wang, Z., and Yin, W. Plug-and-Play Methods Provably Converge with

Properly Trained Denoisers. In *International Conference on Machine Learning*, 2019.

Scarvelis, C., Borde, H. S. d. O., and Solomon, J. Closed-form diffusion models. *arXiv preprint arXiv:2310.12395*, 2023.

Shoushtari, S., Liu, J., Chandler, E. P., Asif, M. S., and Kamilov, U. S. Prior mismatch and adaptation in PnP-ADMM with a nonconvex convergence analysis. *arXiv preprint arXiv:2310.00133*, 2023.

Song, J., Vahdat, A., Mardani, M., and Kautz, J. Pseudoinverse-guided diffusion models for inverse problems. In *International Conference on Learning Representations*, 2023.

Song, Y. and Ermon, S. Generative modeling by estimating gradients of the data distribution. *Advances in neural information processing systems*, 32, 2019.

Song, Y., Sohl-Dickstein, J., Kingma, D. P., Kumar, A., Ermon, S., and Poole, B. Score-based generative modeling through stochastic differential equations. *arXiv preprint arXiv:2011.13456*, 2020.

Song, Y., Durkan, C., Murray, I., and Ermon, S. Maximum likelihood training of score-based diffusion models. *Advances in Neural Information Processing Systems*, 34: 1415–1428, 2021.

Song, Y., Shen, L., Xing, L., and Ermon, S. Solving inverse problems in medical imaging with score-based generative models. In *International Conference on Learning Representations*, 2022.

Sreehari, S., Venkatakrishnan, S. V., Wohlberg, B., Buzzard, G. T., Drummy, L. F., Simmons, J. P., and Bouman, C. A. Plug-and-play priors for bright field electron tomography and sparse interpolation. *IEEE Transactions on Computational Imaging*, 2(4):408–423, 2016.

Sun, Y., Liu, J., and Kamilov, U. Block coordinate regularization by denoising. *Advances in Neural Information Processing Systems*, 32, 2019a.

Sun, Y., Wohlberg, B., and Kamilov, U. S. An online plug-and-play algorithm for regularized image reconstruction. *IEEE Transactions on Computational Imaging*, 5(3):395–408, 2019b.

Sun, Y., Wu, Z., Chen, Y., Feng, B. T., and Bouman, K. L. Provable probabilistic imaging using score-based generative priors. *arXiv preprint arXiv:2310.10835*, 2023.

Tang, J. and Davies, M. A fast stochastic plug-and-play admm for imaging inverse problems. *arXiv preprint arXiv:2006.11630*, 2020.

Ulondu-Mendes, C., Denis, L., Deledalle, C.-A., and Tupin, F. Robustness to spatially-correlated speckle in Plug-and-Play PolSAR despeckling. working paper or preprint, 2023. URL https://telecom-paris.hal.science/hal-04159195.

Venkatakrishnan, S. V., Bouman, C. A., and Wohlberg, B. Plug-and-play priors for model based reconstruction. In *IEEE Global Conference on Signal and Information Processing*, pp. 945–948, 2013.

Wang, H., Du, X., Li, J., Yeh, R. A., and Shakhnarovich, G. Score jacobian chaining: Lifting pretrained 2d diffusion models for 3d generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 12619–12629, 2023.

Xu, X., Sun, Y., Liu, J., Wohlberg, B., and Kamilov, U. S. Provable convergence of plug-and-play priors with MMSE denoisers. *IEEE Signal Processing Letters*, 27: 1280–1284, 2020.

Zhang, K., Zuo, W., Gu, S., and Zhang, L. Learning deep CNN denoiser prior for image restoration. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 3929–3938, 2017.

Zhang, K., Li, Y., Zuo, W., Zhang, L., Van Gool, L., and Timofte, R. Plug-and-play image restoration with deep denoiser prior. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(10):6360–6376, 2021.

Zhao, N., Wei, Q., Basarab, A., Dobigeon, N., Kouame, D., and Tourneret, J.-Y. Fast single image super-resolution. *arXiv preprint arXiv:1510.00143*, 2015.

Zhu, Y., Zhang, K., Liang, J., Cao, J., Wen, B., Timofte, R., and Van Gool, L. Denoising diffusion models for plug-and-play image restoration. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, pp. 1219–1229, 2023.

Zoran, D. and Weiss, Y. From learning models of natural image patches to whole image restoration. In *2011 International Conference on Computer Vision*, pp. 479–486, 2011.

Supplementary Material

Our analysis of SNORE is based on stochastic gradient-descent theory and our implementation on the Gradient-Step PnP code (Hurault et al., 2022a) as well as the Python library DeepInverse for comparisons. In Supplement A, we provide a reproducibility statement for this work. In Supplement B, we present in more details some related works. In Supplement C, SNORE regularization is detailed in simple cases such as Gaussian or Gaussian Mixture priors. In Supplement D, we provide the proofs of the different propositions presented in Section 3. In Supplement E, we give more explanations about our technical assumptions. In Supplement F, we provide more details on our experimental setting and present additional numerical results. In Supplement H, we discuss the boundedness hypothesis and perspectives of generalization.

A. Reproducibility Statement

Anonymous source code is given in supplementary material. It contains a README.md file that explains step by step how to run the algorithm and replicate the results of the paper. Moreover, the pseudocode of SNORE algorithm is given in Algorithm 3 and every comparing methods pseudocodes are given in Appendix F. All parameters setting is detailed in Appendix F. The used datasets and the denoiser weights (Hurault et al., 2022a) are given in the supplementary materials. Theoretical results presented in Section 3 are proved in the appendices.

B. Related works

First of all, the convergence analysis of RED (Algorithm 1) and RED Prox (Algorithm 4) algorithms have been made in the literature (Fermanian et al., 2023; Hurault et al., 2022a). However, to our knowledge, no convergence analysis have be developed for DiffPIR (Zhu et al., 2023).

In the existing literature, some stochastic versions of Plug-and-Play have already been proposed.

Tang & Davies (2020) propose to accelerate the computation of PnP-ADMM algorithms especially when the data-fidelity is heavy to compute by using a mini-batch approximation of the data-fidelity. In the context of gradient descent, this would lead to a computed sequence defined by $\mathbf{x}_{k+1} = \mathbf{x}_k - \delta_k \tilde{\nabla}\mathcal{F}(\mathbf{x}_k, \mathbf{y}) - \frac{\alpha\delta_k}{\sigma^2}(\mathbf{x}_k - D_\sigma(\mathbf{x}_k))$ with $\tilde{\nabla}\mathcal{F}$ a random batch approximation of $\nabla\mathcal{F}$.

Sun et al. (2019a) propose a similar idea based on a batch of random indices, which are optimized at each step. Take $U_i \in \mathbb{R}^{n \times n_i}$ such that $\sum_{i=1}^b n_i = n$ and $\sum_{i=1}^b U_i U_i^T = \mathbf{I}_n$ then $x_{k+1} = x_k - \delta U_{i_k}^T \nabla\mathcal{J}(x_k)$, with i_k a random index in $[1, b]$ and $\mathcal{J} = \mathcal{F} + \alpha\mathcal{R}$ the functional to minimize. On can remark that the two previous works propose an acceleration of PnP based on random batches but no noise is added inside the process. SNORE does not aim at accelerating PnP algorithms but it rather proposes a stochastic improvement of PnP by injecting noise inside the classical PnP regularization.

Laumont et al. (2023) propose to run a stochastic gradient descent (SGD) algorithm with the PnP regularization. Thus the computed sequence, PnP SGD, is defined by $\mathbf{x}_{k+1} = \mathbf{x}_k - \delta_k\nabla\mathcal{F}(\mathbf{x}_k, \mathbf{y}) - \frac{\alpha\delta_k}{\sigma^2}(\mathbf{x}_k - D_\sigma(\mathbf{x}_k)) + \delta_k z_{k+1}$ with $z_{k+1} \sim \mathcal{N}(0, \mathbf{I}_d)$. The step-size decreases with time so the additive noise standard deviation also reduces with time. Unlike the above mentioned methods, PnP SGD is not an acceleration of RED (Romano et al., 2017) but an another type of algorithm to minimize the same objective function, $\mathcal{F}(\cdot, \mathbf{y}) + \alpha\mathcal{P}_\sigma$. SNORE has two main differences with this algorithm. First the noise is only injected inside the denoiser, and never in the data-fidelity term. Second, the standard deviation of the injected noise is fixed during the time (for fixed σ). Luther & Seung (2023) proposed an algorithm similar to Algorithm 3 run with a blind denoiser but do not provide strong theoretical motivation or analysis.

The authors of Kadkhodaie & Simoncelli (2020) propose a method to solve linear inverse problems based on a modification of the Monte Carlo Markov Chain (MCMC) of Langevin Algorithm (Song & Ermon, 2019). Their coarse-to-fine stochastic ascent method is defined by (in our notation) $\mathbf{x}_{k+1} = \mathbf{x}_k + \delta_k\left((\text{Id} - A^T A)(D(\mathbf{x}_k) - \mathbf{x}_k) + A^T(\mathbf{y} - A\mathbf{x}_k)\right) + \gamma_k\mathbf{z}_k$, with $\delta_k = \frac{\delta_0 k}{1 + \delta_0(k-1)}$ the step-size, D a blind denoiser, $\mathbf{z}_k \sim \mathcal{N}(0, \text{Id})$, $\gamma_k = \left((1 - \beta\delta_k)^2 - (1 - \delta_k)^2\right)\sigma_k$, $\beta \in [0, 1]$ and σ_k an estimation of the noise level of $\mathbf{x}_k$. In this method, a blind denoiser D is used instead of a denoiser D_σ which takes the noise level as an input and a non-exact estimation of the noise level of $\mathbf{x}_k$ is made at each iteration. Moreover the parameter β is chosen to make the added noise level γ_k decrease through the iterations, compared to σ that is adapted to the denoiser D_σ in SNORE. This noise $\gamma_k\mathbf{z}_k$ is added to the entire image and not only to the regularization such as in SNORE. Finally, the authors of Kadkhodaie & Simoncelli (2020) do not provide a theoretical analysis of the proposed algorithm, and depending on the noise schedule, the algorithm may perform posterior sampling or stochastic optimization.

In section 2.3, we separate stochastic version of PnP and Diffusion Model (DM) for clarity, but this separation is not strict. We recall that PnP is using a denoiser inside an optimization algorithm and DM generate data by simulating a reverse diffusion process. However, many recent works do not fit exactly inside these categories. Jalal et al. (2021); Sun et al. (2023) develop a restoration method by running a Langevin dynamics (instead of a reverse diffusion process) with a score-matching network. Other posterior sampling algorithms can be simulated with Langevin Dynamics run with a PnP approximation of the score (Laumont et al., 2022; Renaud et al., 2024). Another line of work, it to run Gibbs sampling (Coeurdoux et al., 2023; Bouman & Buzzard, 2023) with a diffusion model approximation of the score. Our new regularization SNORE is part of this field between PnP and DM.

C. SNORE in simple cases

In this section, we detail the behavior of the SNORE regularization in simple cases (Gaussian prior and Gaussian Mixture prior) in order to develop our intuition on this regularization. We observe that this regularization is equivalent to the PnP regularization with a Gaussian prior. Then, in the Gaussian Mixture case, we demonstrate that the gradient of the SNORE regularization converges to the gradient of the ideal regularization $-\nabla \log p$ on every compact when $\sigma \to 0$ and we exhibit the speed of convergence. Finally, we simulate SNORE regularization for 1D distribution and compare it to the PnP regularization.

C.1. Gaussian Prior

In order to understand the behavior of Algorithm 2 compared to other algorithms (such as Algorithm 1) we make the computation in a very simple case where the prior is a Gaussian distribution.

We suppose that $p(\mathbf{x}) = \mathcal{N}(\mathbf{x}; \mu, \boldsymbol{\Sigma})$. Then $p_\sigma(\tilde{\mathbf{x}}) = \mathcal{N}(\tilde{\mathbf{x}}; \mu, \boldsymbol{\Sigma} + \sigma^2 \mathbf{I})$.

The score of these distributions has the following expression

$$-\nabla \log p(\mathbf{x}) = \boldsymbol{\Sigma}^{-1}(\mathbf{x} - \mu)$$
$$-\nabla \log p_\sigma(\tilde{\mathbf{x}}) = (\boldsymbol{\Sigma} + \sigma^2 \mathbf{I}_d)^{-1}(\tilde{\mathbf{x}} - \mu).$$

Moreover, the SNORE regularization defined in Equation (5) can be computed in closed form

$$\mathcal{R}_\sigma(\mathbf{x}) = -\mathbb{E}_{\tilde{\mathbf{x}} \sim p_\sigma(\tilde{\mathbf{x}}|\mathbf{x})} \left(\log p_\sigma(\tilde{\mathbf{x}}) \right) = \frac{1}{2} \mathbb{E}_{\tilde{\mathbf{x}} \sim p_\sigma(\tilde{\mathbf{x}}|\mathbf{x})} \left((\tilde{\mathbf{x}} - \mu)^T (\boldsymbol{\Sigma} + \sigma^2 \mathbf{I}_d)^{-1} (\tilde{\mathbf{x}} - \mu) \right)$$

$$= \frac{1}{2} \int_{\mathbb{R}^d} (\tilde{\mathbf{x}} - \mu)^T (\boldsymbol{\Sigma} + \sigma^2 \mathbf{I}_d)^{-1} (\tilde{\mathbf{x}} - \mu) \mathcal{N}(\tilde{\mathbf{x}}; \mathbf{x}, \sigma^2 \mathbf{I}_d) d\tilde{\mathbf{x}}$$

$$= \frac{1}{2} \int_{\mathbb{R}^d} (\tilde{\mathbf{x}} - \mathbf{x})^T (\boldsymbol{\Sigma} + \sigma^2 \mathbf{I}_d)^{-1} (\tilde{\mathbf{x}} - \mathbf{x}) \mathcal{N}(\tilde{\mathbf{x}}; \mathbf{x}, \sigma^2 \mathbf{I}_d) d\tilde{\mathbf{x}} + \frac{1}{2} \int_{\mathbb{R}^d} (\mathbf{x} - \mu)^T (\boldsymbol{\Sigma} + \sigma^2 \mathbf{I}_d)^{-1} (\mathbf{x} - \mu) \mathcal{N}(\tilde{\mathbf{x}}; \mathbf{x}, \sigma^2 \mathbf{I}_d) d\tilde{\mathbf{x}}$$

$$+ \int_{\mathbb{R}^d} (\tilde{\mathbf{x}} - \mathbf{x})^T (\boldsymbol{\Sigma} + \sigma^2 \mathbf{I}_d)^{-1} (\mathbf{x} - \mu) \mathcal{N}(\tilde{\mathbf{x}}; \mathbf{x}, \sigma^2 \mathbf{I}_d) d\tilde{\mathbf{x}}$$

$$= \sigma^2 Tr \left((\boldsymbol{\Sigma} + \sigma^2 \mathbf{I}_d)^{-1} \right) + \frac{1}{2} (\mathbf{x} - \mu)^T (\boldsymbol{\Sigma} + \sigma^2 \mathbf{I}_d)^{-1} (\mathbf{x} - \mu).$$

The gradient of the SNORE regularization is thus

$$\nabla \mathcal{R}_\sigma(\mathbf{x}) = (\boldsymbol{\Sigma} + \sigma^2 \mathbf{I}_d)^{-1}(\mathbf{x} - \mu),$$

and we have that with a Gaussian prior, $\nabla \mathcal{R}_\sigma(\mathbf{x}) = \nabla \mathcal{P}_\sigma(\mathbf{x}) = -\nabla \log p_\sigma(\mathbf{x})$, which makes SNORE equivalent to traditional PnP.

C.2. Gaussian Mixture prior

In this section, we study the behavior of the SNORE regularization in the case of a Gaussian Mixture prior. First, we remember the convergence of $-\nabla \log p_\sigma$ to $-\nabla \log p$ and we compute the speed of convergence of $\nabla \mathcal{R}_\sigma$ to $-\nabla \log p$ when $\sigma \to 0$. Then, simulations are run in 1D to give more intuition.

Let us suppose that the prior p is a Gaussian Mixture Model,

$$p(\mathbf{x}) = \sum_{i=1}^{p} \pi_i \mathcal{N}(\mathbf{x}; \mu_i, \mathbf{\Sigma}_i),$$

with $\pi_i \geq 0$ and $\sum_{i=1}^{p} \pi_i = 1$.

C.2.1. THEORETICAL CONSIDERATIONS

The scores of the prior distribution and the noisy prior distributions have the closed forms

$$\nabla \log p(\mathbf{x}) = -\frac{\sum_{i=1}^{p} \pi_i \mathbf{\Sigma}_i^{-1}(\mathbf{x} - \mu_i)\mathcal{N}(\mathbf{x}; \mu_i, \mathbf{\Sigma}_i)}{\sum_{i=1}^{p} \pi_i \mathcal{N}(\mathbf{x}; \mu_i, \mathbf{\Sigma}_i)}$$

$$\nabla \log p_\sigma(\mathbf{x}) = -\frac{\sum_{i=1}^{p} \pi_i (\mathbf{\Sigma}_i + \sigma^2 \mathbf{I}_d)^{-1}(\mathbf{x} - \mu_i)\mathcal{N}(\mathbf{x}; \mu_i, \mathbf{\Sigma}_i + \sigma^2 \mathbf{I}_d)}{\sum_{i=1}^{p} \pi_i \mathcal{N}(\mathbf{x}; \mu_i, \mathbf{\Sigma}_i + \sigma^2 \mathbf{I}_d)}. \tag{17}$$

First we study the behavior of this noisy prior distribution score when $\sigma \to 0$.

Proposition C.1. *If the prior p is a Gaussian Mixture Model, then there exist $a_p \in \mathbb{R}^+$, $\alpha_0 > 0$, such that $\forall \mathbf{x} \in \mathbb{R}^d$, $\forall \sigma \in]0, \sigma_0]$:*

$$\|\nabla \log p_\sigma(\mathbf{x}) - \nabla \log p(\mathbf{x})\| \leq \sigma^2 a_p(\|\mathbf{x}\|^3 + 1). \tag{18}$$

Proof. We make a series expansion of $\nabla \log p_\sigma(\mathbf{x})$ when σ goes to 0. Recall that

$$\mathcal{N}(\mathbf{x}; \mu_i, \mathbf{\Sigma}_i + \sigma^2 \mathbf{I}_d) = \frac{1}{\sqrt{\det(2\pi(\mathbf{\Sigma}_i + \sigma^2 \mathbf{I}_d))}} \exp\left(-\frac{1}{2}(\mathbf{x} - \mu_i)^T (\mathbf{\Sigma}_i + \sigma^2 \mathbf{I}_d)^{-1}(\mathbf{x} - \mu_i)\right).$$

It is known that $\det(\mathbf{\Sigma}_i + \sigma^2 \mathbf{I}_d) = \det(\mathbf{\Sigma}_i) + \sigma^2 \text{Tr}(\text{Com}(\mathbf{\Sigma}_i)^T) + \mathcal{O}(\sigma^4)$, with Tr the trace of the matrix and $\text{Com}(\mathbf{\Sigma}_i)$ the comatrix of the matrix $\mathbf{\Sigma}_i$. Then we have

$$\mathcal{N}(\mathbf{x}; \mu_i, \mathbf{\Sigma}_i + \sigma^2 \mathbf{I}_d) = \frac{1}{\sqrt{\det(2\pi(\mathbf{\Sigma}_i + \sigma^2 \mathbf{I}_d))}} \exp\left(-\frac{1}{2}(\mathbf{x} - \mu_i)^T (\mathbf{\Sigma}_i + \sigma^2 \mathbf{I}_d)^{-1}(\mathbf{x} - \mu_i)\right)$$

$$= \frac{1}{\sqrt{\det(2\pi\mathbf{\Sigma}_i)}} \left(1 - \sigma^2 \frac{\text{Tr}(\text{Com}(\mathbf{\Sigma}_i))}{2\det(\mathbf{\Sigma}_i)} + \mathcal{O}(\sigma^4)\right)$$

$$\exp\left(-\frac{1}{2}(\mathbf{x} - \mu_i)^T (\mathbf{\Sigma}_i)^{-1}(\mathbf{x} - \mu_i) + \frac{\sigma^2}{2}(\mathbf{x} - \mu_i)^T (\mathbf{\Sigma}_i)^{-2}(\mathbf{x} - \mu_i) + \mathcal{O}(\sigma^4)\right)$$

$$= \mathcal{N}(\mathbf{x}; \mu_i, \mathbf{\Sigma}_i) \left(1 - \sigma^2 \frac{\text{Tr}(\text{Com}(\mathbf{\Sigma}_i))}{2\det(\mathbf{\Sigma}_i)} + \mathcal{O}(\sigma^4)\right) \left(1 + \frac{\sigma^2}{2}(\mathbf{x} - \mu_i)^T (\mathbf{\Sigma}_i)^{-2}(\mathbf{x} - \mu_i) + \mathcal{O}(\sigma^4)\right)$$

$$= \mathcal{N}(\mathbf{x}; \mu_i, \mathbf{\Sigma}_i) \left(1 + \frac{\sigma^2}{2}\left((\mathbf{x} - \mu_i)^T (\mathbf{\Sigma}_i)^{-2}(\mathbf{x} - \mu_i) - \frac{\text{Tr}(\text{Com}(\mathbf{\Sigma}_i))}{\det(\mathbf{\Sigma}_i)}\right) + \mathcal{O}(\sigma^4)\right).$$

We define $u_i(\mathbf{x}) = \frac{1}{2}\left((\mathbf{x} - \mu_i)^T (\mathbf{\Sigma}_i)^{-2}(\mathbf{x} - \mu_i) - \frac{\text{Tr}(\text{Com}(\mathbf{\Sigma}_i))}{\det(\mathbf{\Sigma}_i)}\right)$. For the score approximation, we get

$$\nabla \log p_\sigma(\mathbf{x}) = -\frac{\sum_{i=1}^{p} \pi_i (\mathbf{\Sigma}_i + \sigma^2 \mathbf{I}_d)^{-1}(\mathbf{x} - \mu_i)\mathcal{N}(\mathbf{x}; \mu_i, \mathbf{\Sigma}_i + \sigma^2 \mathbf{I}_d)}{\sum_{i=1}^{p} \pi_i \mathcal{N}(\mathbf{x}; \mu_i, \mathbf{\Sigma}_i + \sigma^2 \mathbf{I}_d)}$$

$$= -\frac{\sum_{i=1}^{p} \pi_i (\mathbf{\Sigma}_i^{-1} + \sigma^2 \mathbf{\Sigma}_i^{-2} + \mathcal{O}(\sigma^4))(\mathbf{x} - \mu_i)\mathcal{N}(\mathbf{x}; \mu_i, \mathbf{\Sigma}_i)\left(1 + \sigma^2 u_i(\mathbf{x}) + \mathcal{O}(\sigma^4)\right)}{\sum_{i=1}^{p} \pi_i \mathcal{N}(\mathbf{x}; \mu_i, \mathbf{\Sigma}_i)\left(1 + \sigma^2 u_i(\mathbf{x}) + \mathcal{O}(\sigma^4)\right)}$$

$$= -\sigma^2 \left(\frac{\sum_{i=1}^{p} \pi_i \left(\mathbf{\Sigma}_i^{-1}(\mathbf{x} - \mu_i)u_i(\mathbf{x}) + \mathbf{\Sigma}_i^{-2}(\mathbf{x} - \mu_i)\right)\mathcal{N}(\mathbf{x}; \mu_i, \mathbf{\Sigma}_i)}{\sum_{i=1}^{p} \pi_i \mathcal{N}(\mathbf{x}; \mu_i, \mathbf{\Sigma}_i)}\right)$$

$$- \frac{\nabla p(\mathbf{x})}{p(\mathbf{x})\left(1 + \frac{\sigma^2}{p(\mathbf{x})}\sum_{i=1}^{p} \pi_i \mathcal{N}(\mathbf{x}; \mu_i, \mathbf{\Sigma}_i)u_i(\mathbf{x})\right)} + \mathcal{O}(\sigma^4)$$

$$= \nabla \log p(\mathbf{x}) - \sigma^2 c_p(\mathbf{x}) + \mathcal{O}(\sigma^4),$$

with

$$c_p(\mathbf{x}) = \frac{\sum_{i=1}^{p} \pi_i \left(\mathbf{\Sigma}_i^{-1}(\mathbf{x} - \mu_i)u_i(\mathbf{x}) + \mathbf{\Sigma}_i^{-2}(\mathbf{x} - \mu_i)\right) \mathcal{N}(\mathbf{x}; \mu_i, \mathbf{\Sigma}_i)}{p(\mathbf{x})} - \frac{\nabla p(\mathbf{x})}{p^2(\mathbf{x})} \sum_{i=1}^{p} \pi_i \mathcal{N}(\mathbf{x}; \mu_i, \mathbf{\Sigma}_i)u_i(\mathbf{x}).$$

This demonstrates the following inequality

$$\nabla \log p_\sigma(\mathbf{x}) \underset{\sigma \to 0}{=} \nabla \log p(\mathbf{x}) - \sigma^2 c_p(\mathbf{x}) + \mathcal{O}(\sigma^4), \tag{19}$$

from which we can deduce a pointwise convergence of $\nabla \log p_\sigma$ to $\nabla \log p$ with speed σ^2. Indeed, there exist $c_p : \mathbb{R}^d \mapsto \mathbb{R}$ and $\sigma_0 > 0$, such that for $\sigma \in]0, \sigma_0]$,

$$\|\nabla \log p_\sigma(\mathbf{x}) - \nabla \log p(\mathbf{x})\| \leq 2\sigma^2 \|c_p(\mathbf{x})\|, \tag{20}$$

Next, one can remark that $\forall j \in [1, p]$,

$$\frac{\pi_j \mathcal{N}(\mathbf{x}; \mu_j, \mathbf{\Sigma}_j)}{p(\mathbf{x})} = \frac{\pi_j \mathcal{N}(\mathbf{x}; \mu_j, \mathbf{\Sigma}_j)}{\sum_{i=1}^{p} \pi_i \mathcal{N}(\mathbf{x}; \mu_i, \mathbf{\Sigma}_i)} = \frac{1}{1 + \frac{\sum_{i \neq j} \pi_i \mathcal{N}(\mathbf{x}; \mu_i, \mathbf{\Sigma}_i)}{\pi_j \mathcal{N}(\mathbf{x}; \mu_j, \mathbf{\Sigma}_j)}} \leq 1.$$

As a consequence, there exists $\alpha_p \geq 0$ such that

$$\|c_p(\mathbf{x})\| \leq \alpha_p(\|\mathbf{x}\|^3 + 1) \tag{21}$$

By combining (20) and (21), we obtain that there exist a_p and $\alpha_0 > 0$, such that $\forall \mathbf{x} \in \mathbb{R}^d, \forall \sigma \in]0, \sigma_0]$,

$$\|\nabla \log p_\sigma(\mathbf{x}) - \nabla \log p(\mathbf{x})\| \leq \sigma^2 a_p(\|\mathbf{x}\|^3 + 1). \tag{22}$$

$\square$

A similar work can be done to evaluate the approximation done by our regularization $\mathcal{R}_\sigma$.

Proposition C.2. *If the prior p is a Gaussian Mixture Model, then there exist $\sigma_0 > 0$ and $e_p \in \mathbb{R}^+$ such that $\forall \mathbf{x} \in \mathbb{R}^d$ and $\sigma \in]0, \sigma_0]$,*

$$\|\nabla \mathcal{R}_\sigma(\mathbf{x}) + \nabla \log p(\mathbf{x})\| \leq \sigma e_p(\|\mathbf{x}\|^2 + 1). \tag{23}$$

Proof. To emphasize the dependence of $\mathcal{R}$ in σ, we will denote it as $\mathcal{R}_\sigma$ in the following computations. The behavior of our regularization gradient $\nabla \mathcal{R}_\sigma(\mathbf{x}) = -\mathbb{E}_{\tilde{\mathbf{x}} \sim p_\sigma(\tilde{\mathbf{x}}|\mathbf{x})} (\nabla \log p_\sigma(\tilde{\mathbf{x}}))$ is

$$
\begin{aligned}
\|\nabla \mathcal{R}_\sigma(\mathbf{x}) + \nabla \log p(\mathbf{x})\| &= \|\mathbb{E}_{\tilde{\mathbf{x}} \sim p_\sigma(\tilde{\mathbf{x}}|\mathbf{x})} (\nabla \log p_\sigma(\tilde{\mathbf{x}})) - \nabla \log p(\mathbf{x})\| \\
&\leq \|\mathbb{E}_{\tilde{\mathbf{x}} \sim p_\sigma(\tilde{\mathbf{x}}|\mathbf{x})} (\nabla \log p_\sigma(\tilde{\mathbf{x}}) - \nabla \log p(\tilde{\mathbf{x}})) \| + \|\mathbb{E}_{\tilde{\mathbf{x}} \sim p_\sigma(\tilde{\mathbf{x}}|\mathbf{x})} (\nabla \log p(\tilde{\mathbf{x}})) - \nabla \log p(\mathbf{x})\| \\
&\leq \sigma^2 \mathbb{E}_{\tilde{\mathbf{x}} \sim p_\sigma(\tilde{\mathbf{x}}|\mathbf{x})} (c_p(\tilde{\mathbf{x}})) + \|\mathbb{E}_{\tilde{\mathbf{x}} \sim p_\sigma(\tilde{\mathbf{x}}|\mathbf{x})} (\nabla \log p(\tilde{\mathbf{x}})) - \nabla \log p(\mathbf{x})\| \\
&\leq \sigma^2 a_p \mathbb{E}_{\tilde{\mathbf{x}} \sim p_\sigma(\tilde{\mathbf{x}}|\mathbf{x})} (\|\tilde{\mathbf{x}}\|^3 + 1) + \|\mathbb{E}_{\tilde{\mathbf{x}} \sim p_\sigma(\tilde{\mathbf{x}}|\mathbf{x})} (\nabla \log p(\tilde{\mathbf{x}})) - \nabla \log p(\mathbf{x})\|.
\end{aligned}
$$

By using that for $a, b \in \mathbb{R}^+$, $(a + b)^3 \leq 4(a^3 + b^3)$, we get the following inequalities $\mathbb{E}_{\tilde{\mathbf{x}} \sim p_\sigma(\tilde{\mathbf{x}}|\mathbf{x})} (\|\tilde{\mathbf{x}}\|^3) \leq \mathbb{E}_{\tilde{\mathbf{x}} \sim p_\sigma(\tilde{\mathbf{x}}|\mathbf{x})} \left((\|\tilde{\mathbf{x}} - \mathbf{x}\| + \|\mathbf{x}\|)^3\right) \leq 4\mathbb{E}_{\tilde{\mathbf{x}} \sim p_\sigma(\tilde{\mathbf{x}}|\mathbf{x})} (\|\tilde{\mathbf{x}} - \mathbf{x}\|^3 + \|\mathbf{x}\|^3) = 8d\sigma^3 \sqrt{\frac{2}{\pi}} + 4\|\mathbf{x}\|^3$. This leads to the following inequality

$$\|\nabla \mathcal{R}_\sigma(\mathbf{x}) + \nabla \log p(\mathbf{x})\| \leq \sigma^2 a_p \left(8d\sigma^3 \sqrt{\frac{2}{\pi}} + 4\|\mathbf{x}\|^3 + 1\right) + \|\mathbb{E}_{\tilde{\mathbf{x}} \sim p_\sigma(\tilde{\mathbf{x}}|\mathbf{x})} (\nabla \log p(\tilde{\mathbf{x}})) - \nabla \log p(\mathbf{x})\|. \tag{24}$$

As $\nabla \log p$ is $\mathcal{C}^2$ in $\mathbf{x}$, by the Taylor theorem, there exists $r > 0$ such that $\forall \mathbf{y}, \mathbf{z} \in \mathcal{B}(\mathbf{x}, r)$

$$\|\nabla \log p(\mathbf{y}) - \nabla \log p(\mathbf{z})\| \leq M_{\mathbf{x}} \|\mathbf{y} - \mathbf{z}\|,$$

with $M_{\mathbf{x}} = 2\|\nabla^2 \log p(\mathbf{x})\|$ and $\mathcal{B}(\mathbf{x}, r) = \{\mathbf{y} \in \mathbb{R}^d | \|\mathbf{y} - \mathbf{x}\| \leq r\}$.

So we have

$$\|\mathbb{E}_{\tilde{\mathbf{x}} \sim p_\sigma(\tilde{\mathbf{x}}|\mathbf{x})} (\nabla \log p(\tilde{\mathbf{x}})) - \nabla \log p(\mathbf{x})\| = \| \int_{\mathbb{R}^d} \nabla \log p(\tilde{\mathbf{x}}) \mathcal{N}(\tilde{\mathbf{x}}; \mathbf{x}, \sigma^2 \mathbf{I}_d) d\tilde{\mathbf{x}} - \nabla \log p(\mathbf{x})\|$$

$$\leq \| \int_{\mathcal{B}(\mathbf{x}, r)} (\nabla \log p(\tilde{\mathbf{x}}) - \nabla \log p(\mathbf{x})) \mathcal{N}(\tilde{\mathbf{x}}; \mathbf{x}, \sigma^2 \mathbf{I}_d) d\tilde{\mathbf{x}} \| + \| \int_{\mathbb{R}^d \setminus \mathcal{B}(\mathbf{x}, r)} (\nabla \log p(\tilde{\mathbf{x}}) - \nabla \log p(\mathbf{x})) \mathcal{N}(\tilde{\mathbf{x}}; \mathbf{x}, \sigma^2 \mathbf{I}_d) d\tilde{\mathbf{x}} \|$$

$$\leq M_{\mathbf{x}} \int_{\mathcal{B}(\mathbf{x}, r)} \|\tilde{\mathbf{x}} - \mathbf{x}\| \mathcal{N}(\tilde{\mathbf{x}}; \mathbf{x}, \sigma^2 \mathbf{I}_d) d\tilde{\mathbf{x}} + \| \int_{\mathbb{R}^d \setminus \mathcal{B}(\mathbf{x}, r)} (\nabla \log p(\tilde{\mathbf{x}}) - \nabla \log p(\mathbf{x})) \mathcal{N}(\tilde{\mathbf{x}}; \mathbf{x}, \sigma^2 \mathbf{I}_d) d\tilde{\mathbf{x}} \|$$

$$\leq M_{\mathbf{x}} \sigma \int_{\mathbb{R}^d} \|\tilde{\mathbf{x}}\| \mathcal{N}(\tilde{\mathbf{x}}; 0, \mathbf{I}_d) d\tilde{\mathbf{x}} + \| \int_{\mathbb{R}^d \setminus \mathcal{B}(\mathbf{x}, r)} (\nabla \log p(\tilde{\mathbf{x}}) - \nabla \log p(\mathbf{x})) \mathcal{N}(\tilde{\mathbf{x}}; \mathbf{x}, \sigma^2 \mathbf{I}_d) d\tilde{\mathbf{x}} \|. \tag{25}$$

One can notice that $\nabla \log p$ is sub-linear, so there exists $b_p \geq 0$ such that $\forall \mathbf{x} \in \mathbb{R}^d, \|\nabla \log p(\mathbf{x})\| \leq b_p(\|\mathbf{x}\| + 1)$. Moreover, $\int_{\mathbb{R}^d} \|\tilde{\mathbf{x}}\| \mathcal{N}(\tilde{\mathbf{x}}; 0, \mathbf{I}_d) d\tilde{\mathbf{x}} \leq \sqrt{\int_{\mathbb{R}^d} \|\tilde{\mathbf{x}}\|^2 \mathcal{N}(\tilde{\mathbf{x}}; 0, \mathbf{I}_d) d\tilde{\mathbf{x}}} = \sqrt{d}$. Combining this property with relation (25), we get

$$\|\mathbb{E}_{\tilde{\mathbf{x}} \sim p_\sigma(\tilde{\mathbf{x}}|\mathbf{x})} (\nabla \log p(\tilde{\mathbf{x}})) - \nabla \log p(\mathbf{x})\| \leq M_{\mathbf{x}} \sqrt{d} \sigma + \int_{\mathbb{R}^d \setminus \mathcal{B}(\mathbf{x}, r)} (b_p \|\tilde{\mathbf{x}}\| + b_p \|\mathbf{x}\| + 2) \mathcal{N}(\tilde{\mathbf{x}}; \mathbf{x}, \sigma^2 \mathbf{I}_d) d\tilde{\mathbf{x}}$$

$$\leq M_{\mathbf{x}} \sqrt{d} \sigma + \int_{\mathbb{R}^d \setminus \mathcal{B}(\mathbf{x}, r)} (b_p \|\tilde{\mathbf{x}}\| + b_p \|\mathbf{x}\| + 2) \mathcal{N}(\tilde{\mathbf{x}}; \mathbf{x}, \sigma^2 \mathbf{I}_d) d\tilde{\mathbf{x}}.$$

Moreover on the set $\mathbb{R}^d \setminus \mathcal{B}(\mathbf{x}, r)$, the inequality $\|\mathbf{x} - \tilde{\mathbf{x}}\|^2 \geq \frac{r^2}{2} + \frac{\|\mathbf{x} - \tilde{\mathbf{x}}\|^2}{2}$ holds. This leads to $\mathcal{N}(\tilde{\mathbf{x}}; \mathbf{x}, \sigma^2 \mathbf{I}_d) \leq 2^{\frac{d}{2}} \exp\left(-\frac{r^2}{2\sigma^2}\right) \mathcal{N}(\tilde{\mathbf{x}}; \mathbf{x}, 2\sigma^2 \mathbf{I}_d)$. Injecting this relation into the previous computation, we obtain

$$\|\mathbb{E}_{\tilde{\mathbf{x}} \sim p_\sigma(\tilde{\mathbf{x}}|\mathbf{x})} (\nabla \log p(\tilde{\mathbf{x}})) - \nabla \log p(\mathbf{x})\| \leq M_{\mathbf{x}} \sqrt{d} \sigma + 2^{\frac{d}{2}} \exp\left(-\frac{r^2}{2\sigma^2}\right) \int_{\mathbb{R}^d \setminus \mathcal{B}(\mathbf{x}, r)} (b_p \|\tilde{\mathbf{x}}\| + b_p \|\mathbf{x}\| + 2) \mathcal{N}(\tilde{\mathbf{x}}; \mathbf{x}, 2\sigma^2 \mathbf{I}_d) d\tilde{\mathbf{x}}$$

$$\leq M_{\mathbf{x}} \sqrt{d} \sigma + 2^{\frac{d}{2}} \exp\left(-\frac{r^2}{2\sigma^2}\right) \int_{\mathbb{R}^d} (b_p \|\tilde{\mathbf{x}}\| + b_p \|\mathbf{x}\| + 2) \mathcal{N}(\tilde{\mathbf{x}}; \mathbf{x}, 2\sigma^2 \mathbf{I}_d) d\tilde{\mathbf{x}}$$

$$\leq M_{\mathbf{x}} \sqrt{d} \sigma + 2^{\frac{d}{2}} \exp\left(-\frac{r^2}{2\sigma^2}\right) (b_p \sqrt{\|\mathbf{x}\|^2 + b_p \sqrt{d} \sigma^2} + b_p \|\mathbf{x}\| + 2)$$

$$\leq M_{\mathbf{x}} \sqrt{d} \sigma + \mathcal{O}(\sigma^4)$$

$$\leq e_p(\|\mathbf{x}\|^2 + 1) \sqrt{d} \sigma + \mathcal{O}(\sigma^4).$$

The last inequality holds because there exists $e_p \in \mathbb{R}^+$ such that $M_{\mathbf{x}} = 2\|\nabla^2 \log p(\mathbf{x})\| \leq e_p(\|\mathbf{x}\|^2 + 1)$.

Combining the above inequalities with Equation (24), we have that

$$\|\nabla \mathcal{R}_\sigma(\mathbf{x}) + \nabla \log p(\mathbf{x})\| \leq \sigma^2 a_p \left(8d\sigma^3 \sqrt{\frac{2}{\pi}} + 4\|\mathbf{x}\|^3 + 1\right) + e_p(\|\mathbf{x}\|^2 + 1)\sqrt{d} \sigma + \mathcal{O}(\sigma^4)$$

$$\leq \sigma \tilde{e}_p(\|\mathbf{x}\|^2 + 1) + \sigma^2 a_p(4\|\mathbf{x}\|^3 + 1) + \mathcal{O}(\sigma^4),$$

with $\tilde{e}_p = e_p \sqrt{d}$.

By defining $d_p = 2\tilde{e}_p$, we have demonstrated that there exist $\sigma_0 > 0, d_p \geq 0$ such that $\forall \mathbf{x} \in \mathbb{R}^d$ and $\forall \sigma \in]0, \sigma_0]$,

$$\|\nabla \mathcal{R}_\sigma(\mathbf{x}) + \nabla \log p(\mathbf{x})\| \leq \sigma d_p(\|\mathbf{x}\|^2 + 1). \tag{26}$$

$\square$

Proposition C.2 shows that, in a case of a GMM prior, our regularization approximates the prior score with a pointwise speed of σ. This speed of approximation is slower than for the traditional PnP (Proposition C.1).

C.2.2. SIMULATIONS

In Figure 4, we display a non-trivial Gaussian Mixture distribution in 1D with three Gaussians. Using Equation (17), we display $-\log p_\sigma$. In order to show $\mathcal{R}_\sigma$, we use Equation (5) and the Euler method to approximate the integration on $\tilde{\mathbf{x}}$.
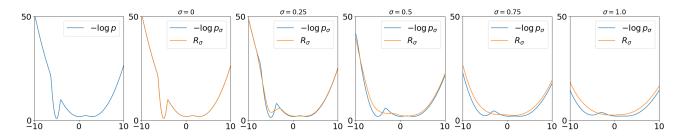

Figure 4. Leftmost: Score of the prior p. *Rightmost:* Values of $-\log p_\sigma$ and $\mathcal{R}_\sigma$ for five values of $\sigma > 0$.

First, we observe that both $-\log p_\sigma$ and $\mathcal{R}_\sigma$ converge to $-\log p$ when $\sigma \to 0$. We also see that $\mathcal{R}_\sigma$ seems to converge to $-\log p$ more slowly that $-\log p_\sigma$ as suggested by Proposition C.1 and Proposition C.2. Finally, $\mathcal{R}_\sigma$ seems to be "more convex" so easier to minimize and it has a similar minimum that $-\log p_\sigma$. This simulation suggests that $\mathcal{R}_\sigma$ is an easier potential to minimize.

D. Proofs of section 3

D.1. Proof of critical Point Analysis

First, we recall a part of the proof of (Laumont et al., 2023, Proposition 1) where it has been proved under some assumptions on the prior distribution that the posterior score approximation converges uniformly to the posterior score on every compact of $\mathbb{R}^d$.

Proposition D.1. *(Laumont et al., 2023, Proposition 1) Under Assumption 3.1, for $\mathbf{K}$ a compact of $\mathbb{R}^d$, $\nabla \log p_\sigma(\cdot|\mathbf{y})$ converges uniformly to $\nabla \log p(\cdot|\mathbf{y})$ on $\mathbf{K}$,*

$$\lim_{\sigma \to 0} \sup_{\mathbf{K}} \|\nabla \log p_\sigma(\cdot|\mathbf{y}) - \nabla \log p(\cdot|\mathbf{y})\| = 0.$$

Proof. By the Bayes' theorem, $\nabla \log p_\sigma(\cdot|\mathbf{y}) = \nabla \log p(\mathbf{y}|\cdot) + \nabla \log p_\sigma$. As a consequence, Proposition D.1 is equivalent to show that the score converges uniformly on $\mathbf{K}$.

For $f \in \mathrm{C}(\mathbb{R}^d, \mathbb{R}^p)$, with $p \in \mathbb{N}$ and $\|f\|_\infty < +\infty$, we define for $\mathbf{x} \in \mathbb{R}^d$

$$f_\sigma(\mathbf{x}) = (f * \mathcal{N}_\sigma)(\mathbf{x}) = \int_{\mathbb{R}^d} f(\tilde{\mathbf{x}} - \mathbf{x})\mathcal{N}(\mathbf{x}; 0, \sigma^2 \mathbf{I}_d)d\tilde{\mathbf{x}}.$$

For $\epsilon > 0$, there exists $R \in \mathbb{R}^+$ such that

$$\int_{\mathbb{R}^d \setminus \mathcal{B}(0,R)} \|f(\mathbf{x} - \sigma\tilde{\mathbf{x}}) - f(\mathbf{x})\|\mathcal{N}(\tilde{\mathbf{x}}; 0, \mathbf{I}_d)d\tilde{\mathbf{x}} \le 2\|f\|_\infty \int_{\mathbb{R}^d \setminus \mathcal{B}(0,R)} \mathcal{N}(\tilde{\mathbf{x}}; 0, \mathbf{I}_d)d\tilde{\mathbf{x}} \le \frac{\epsilon}{2}.$$

Then $\tilde{\mathbf{K}} = \mathbf{K} + \mathcal{B}(0, R)$ (Minkowski sum) is compact, so f is uniformly continuous on $\tilde{\mathbf{K}}$. There exists $\xi > 0$ such that

$$\forall \sigma \in [0, \xi], \forall \tilde{\mathbf{x}} \in \mathcal{B}(0, R), \forall \mathbf{x} \in \mathbf{K}, \quad \|f(\mathbf{x} - \sigma\tilde{\mathbf{x}}) - f(\mathbf{x})\| \le \frac{\epsilon}{2}.$$

We can deduce for $\mathbf{x} \in \mathbf{K}, \sigma \in [0, \xi]$ that

$$
\begin{aligned}
\|f_\sigma(\mathbf{x}) - f(\mathbf{x})\| &\leq \int_{\mathbb{R}^d} \|f(\mathbf{x} - \tilde{\mathbf{x}}) - f(\mathbf{x})\| \mathcal{N}(\tilde{\mathbf{x}}; 0, \sigma^2 \mathbf{I}_d) d\tilde{\mathbf{x}} \\
&\leq \int_{\mathbb{R}^d} \|f(\mathbf{x} - \sigma \tilde{\mathbf{x}}) - f(\mathbf{x})\| \mathcal{N}(\tilde{\mathbf{x}}; 0, \mathbf{I}_d) d\tilde{\mathbf{x}} \\
&\leq \int_{\mathcal{B}(0,R)} \|f(\mathbf{x} - \sigma \tilde{\mathbf{x}}) - f(\mathbf{x})\| \mathcal{N}(\tilde{\mathbf{x}}; 0, \mathbf{I}_d) d\tilde{\mathbf{x}} + \int_{\mathbb{R}^d \backslash \mathcal{B}(0,R)} \|f(\mathbf{x} - \sigma \tilde{\mathbf{x}}) - f(\mathbf{x})\| \mathcal{N}(\tilde{\mathbf{x}}; 0, \mathbf{I}_d) d\tilde{\mathbf{x}} \\
&\leq \epsilon.
\end{aligned}
$$

So we have the uniform convergence of f_σ to f on $\mathbf{K}$: $\forall \epsilon > 0$, there exists $\xi > 0$ such that $\forall \sigma \in [0, \xi]$

$$
\sup_{\mathbf{x} \in \mathbf{K}} \|f_\sigma(\mathbf{x}) - f(\mathbf{x})\| \leq \epsilon.
$$

Applying this result with $f = p$ and $f = \nabla p$ (because $\nabla p_\sigma = \nabla(p * \mathcal{N}_\sigma) = (\nabla p) * \mathcal{N}_\sigma$, we finally get

$$
\sup_{\mathbf{K}} \|\nabla \log p_\sigma - \nabla \log p\| = \sup_{\mathbf{K}} \left\| \frac{\nabla p_\sigma}{p_\sigma} - \frac{\nabla p}{p} \right\| = \sup_{\mathbf{K}} \left\| \frac{(\nabla p_\sigma - \nabla p)p + \nabla p(p - p_\sigma)}{p_\sigma p} \right\|.
$$

We define $m_p = \min\{\inf_{\mathbf{x} \in \mathbf{K}} p(\mathbf{x})\} > 0$, because $\sup_{\mathbf{K}} \|p - p_\sigma\| \to 0$, there exists $\sigma_1 > 0$ such that for $0 < \sigma \leq \sigma_1$, $\forall \mathbf{x} \in \mathbf{K}, p_\sigma(x) > \frac{m_p}{2}$. Thus for $0 < \sigma \leq \sigma_1$,

$$
\sup_{\mathbf{K}} \|\nabla \log p_\sigma - \nabla \log p\| \leq \frac{2(\sup_{\mathbf{K}} \|\nabla p_\sigma - \nabla p\|) \|p\|_\infty + \|\nabla p\|_\infty (\sup_{\mathbf{K}} \|p - p_\sigma\|)}{m_p^2} \to 0
$$

Thus we have prove Propositon D.1.

$\square$

D.1.1. PROOF OF PROPOSITION 3.1

We define $\mathbf{L} = \mathbf{K} + \mathcal{B}(0, 1)$. By Proposition D.1, $\nabla \log p_\sigma$ converges uniformly to $\nabla \log p$ on $\mathbf{L}$.

For $\epsilon > 0, \mathbf{x} \in \mathbf{K}$,

$$
\| -\mathcal{R}_\sigma(\mathbf{x}) - \nabla \log p(\mathbf{x})\| = \left\| \int_{\mathbb{R}^d} \nabla \log p_\sigma(\mathbf{x} + \zeta) \mathcal{N}(\zeta; 0, \sigma^2 \mathbf{I}_d) d\zeta - \nabla \log p(\mathbf{x}) \right\|
$$

$$
\leq \int_{\mathbb{R}^d} \|\nabla \log p_\sigma(\mathbf{x} + \zeta) - \nabla \log p(\mathbf{x})\| \mathcal{N}(\zeta; 0, \sigma^2 \mathbf{I}_d) d\zeta
$$

$$
\leq \int_{\mathbb{R}^d} \|\nabla \log p_\sigma(\mathbf{x} + \zeta) - \nabla \log p(\mathbf{x} + \zeta)\| \mathcal{N}(\zeta; 0, \sigma^2 \mathbf{I}_d) d\zeta + \int_{\mathbb{R}^d} \|\nabla \log p(\mathbf{x} + \zeta) - \nabla \log p(\mathbf{x})\| \mathcal{N}(\zeta; 0, \sigma^2 \mathbf{I}_d) d\zeta
$$

$$
\leq \int_{\mathcal{B}(0,1)} \|\nabla \log p_\sigma - \nabla \log p\|_{\infty, \mathbf{L}} \mathcal{N}(\zeta; 0, \sigma^2 \mathbf{I}_d) d\zeta + \int_{\mathbb{R}^d \backslash \mathcal{B}(0,1)} \left(\|\nabla \log p_\sigma(\mathbf{x} + \zeta)\| + \|\nabla \log p(\mathbf{x} + \zeta)\| \right) \mathcal{N}(\zeta; 0, \sigma^2 \mathbf{I}_d) d\zeta
$$

$$
+ \int_{\mathbb{R}^d} \|\nabla \log p(\mathbf{x} + \zeta) - \nabla \log p(\mathbf{x})\| \mathcal{N}(\zeta; 0, \sigma^2 \mathbf{I}_d) d\zeta.
$$

Because $\mathbf{L}$ is compact and $\nabla \log p$ is continuous on $\bar{\mathbf{K}}$, $\nabla \log p$ is uniformly continuous on $\mathbf{L}$. So there exists $1 \geq \mu > 0$ such that $\forall \mathbf{x}, \mathbf{y} \in \mathbf{L}$, if $\|\mathbf{x} - \mathbf{y}\| \leq \mu$,

$$
\|\nabla \log p(\mathbf{x}) - \nabla \log p(\mathbf{y})\| \leq \epsilon. \tag{27}
$$

Then by using Equation (27), Assumptions 3.1(**b**) and Assumptions 3.2, we have

$$\|\nabla \mathcal{R}_\sigma(\mathbf{x}) + \nabla \log p(\mathbf{x})\|$$

$$\leq (\sup_{\mathbf{L}} \|\nabla \log p_\sigma - \nabla \log p\|) \int_{\mathcal{B}(0,1)} \mathcal{N}(\zeta; 0, \sigma^2 \mathbf{I}_d) d\zeta$$

$$+ \int_{\mathbb{R}^d \setminus \mathcal{B}(0,1)} \left(B\sigma^\beta (1 + \|\mathbf{x} + \zeta\|^r) + A(1 + \|\mathbf{x} + \zeta\|^q) \right) \mathcal{N}(\zeta; 0, \sigma^2 \mathbf{I}_d) d\zeta$$

$$+ \epsilon \int_{\mathcal{B}(0,\mu)} \mathcal{N}(\zeta; 0, \sigma^2 \mathbf{I}_d) d\zeta + \int_{\mathbb{R}^d \setminus \mathcal{B}(0,\mu)} (A(1 + \|\mathbf{x} + \zeta\|^q) + A(1 + \|\mathbf{x}\|^q)) \mathcal{N}(\zeta; 0, \sigma^2 \mathbf{I}_d) d\zeta$$

$$\leq \sup_{\mathbf{L}} (\|\nabla \log p_\sigma - \nabla \log p\|) + \epsilon$$

$$+ \int_{\mathbb{R}^d \setminus \mathcal{B}(0,\mu)} \left(B\sigma^\beta (1 + \|\mathbf{x} + \zeta\|^r) + 2A(1 + \|\mathbf{x} + \zeta\|^q) + A(1 + \|\mathbf{x}\|^q) \right) \mathcal{N}(\zeta; 0, \sigma^2 \mathbf{I}_d) d\zeta,$$

where in the last inequality we have used that $\mu \leq 1$. Then $\|\mathbf{x} + \zeta\| \leq \|\mathbf{x}\| + \|\zeta\| \leq R + \|\zeta\|$, where $R = \sup_{\mathbf{x} \in \mathbf{K}} \|\mathbf{x}\| < +\infty$ because $\mathbf{K}$ is compact. Moreover on the set $\mathbb{R}^d \setminus \mathcal{B}(0, \mu)$, the inequality $\|\zeta\|^2 \geq \frac{\mu^2}{2} + \frac{\|\zeta\|^2}{2}$ holds. This leads to $\mathcal{N}(\zeta; 0, \sigma^2 \mathbf{I}_d) \leq 2^{\frac{d}{2}} \exp\left(-\frac{\mu^2}{2\sigma^2}\right) \mathcal{N}(\zeta; 0, 2\sigma^2 \mathbf{I}_d)$; and then to the following inequality

$$\|\nabla \mathcal{R}_\sigma(\mathbf{x}) + \nabla \log p(\mathbf{x})\| \leq \|\nabla \log p_\sigma - \nabla \log p\|_{\infty, \mathbf{L}} + \epsilon$$

$$+ \int_{\mathbb{R}^d \setminus \mathcal{B}(0,\mu)} \left(B\sigma^\beta (1 + (R + \|\zeta\|)^r) + 2A(1 + (R + \|\zeta\|)^q) + A(1 + R^q) \right) 2^{\frac{d}{2}} \exp\left(-\frac{\mu^2}{2\sigma^2}\right) \mathcal{N}(\zeta; 0, 2\sigma^2 \mathbf{I}_d) d\zeta$$

$$\leq \sup_{\mathbf{L}} (\|\nabla \log p_\sigma - \nabla \log p\|) + \epsilon + (\sigma^\beta C_{r,B,R} + C_{q,A,R}) \exp\left(-\frac{\mu^2}{2\sigma^2}\right),$$

with $C_{r,B,R}$ a constant depending on r, B and R; and $C_{q,A,R}$ a constant depending on q, A and R. By the uniform convergence of $\nabla \log p_\sigma$ to $\nabla \log p$ on $\mathbf{L}$, there exists $\sigma_0 > 0$ such that $\forall \sigma \leq \sigma_0, \sup_{\mathbf{L}} \|\nabla \log p_\sigma - \nabla \log p\| \leq \epsilon$. Then with the polynomial-exponential behavior, there exists, $\sigma_1 > 0$, such that $\forall \sigma \leq \sigma_1, (\sigma^\beta C_{r,B,R} + C_{q,A,R}) \exp\left(-\frac{\mu^2}{2\sigma^2}\right) \leq \epsilon$. Finally, for $\sigma \leq \min(\sigma_0, \sigma_1)$,

$$\|\nabla \mathcal{R}_\sigma(\mathbf{x}) + \nabla \log p(\mathbf{x})\| \leq 3\epsilon.$$

D.1.2. PROOF OF PROPOSITION 3.2

We will follow the same structure as the proof of (Laumont et al., 2023, Proposition 1) and use our Proposition 3.1. For $\mathbf{x} \in \mathbf{S_K}$, by definition there exist $(\sigma_n)_{n \in \mathbb{N}} \in \mathbf{E}$, $\mathbf{x}_n \in \mathbf{S}_{\sigma_n, \mathbf{K}}$ and $\phi : \mathbb{N} \to \mathbb{N}$ strictly increasing such that $\mathbf{x}_{\phi(n)} \to \mathbf{x}$. By Proposition 3.1, $\nabla \mathcal{R}_\sigma$ converges uniformly to $-\nabla \log p$ on $\mathbf{K}$. So $\nabla \mathcal{F}(\cdot, \mathbf{y}) + \alpha \nabla \mathcal{R}_\sigma$ converges uniformly to $\nabla \mathcal{F}(\cdot, \mathbf{y}) - \alpha \nabla \log p = \nabla \mathcal{J}$ on $\mathbf{K}$. Because $\forall n \in \mathbb{N}, \mathbf{x}_{\phi(n)} \in \mathbf{K}$, we have that $\nabla \mathcal{F}(\mathbf{x}_{\phi(n)}, \mathbf{y}) + \alpha \nabla \mathcal{R}_{\sigma_{\phi(n)}}(\mathbf{x}_{\phi(n)}) \to \nabla \mathcal{J}(\mathbf{x})$. On the other hand, as $\forall n \in \mathbb{N}$, because $\mathbf{x}_{\phi(n)} \in \mathbf{S}_{\sigma_{\phi(n)}, \mathbf{K}}$, $\nabla \mathcal{F}(\mathbf{x}_{\phi(n)}, \mathbf{y}) + \alpha \nabla \mathcal{R}_{\sigma_{\phi(n)}}(\mathbf{x}_{\phi(n)}) = 0$, we have $\nabla \mathcal{J}(\mathbf{x}) = 0$, which shows that $\mathbf{x} \in \mathbf{S_K^\star}$.

D.1.3. PROOF OF PROPOSITION 3.3

We will apply the result (Doucet & Tadic, 2017, Theorem 2.1, (ii)). The (Doucet & Tadic, 2017, Assumption 2.1) is verified by Assumption 3.3. First we define $\xi_k = \nabla \mathcal{F}(\mathbf{x}_k, \mathbf{y}) + \alpha \nabla \log p_\sigma(\mathbf{x}_k + \zeta_k) - \nabla \mathcal{J}_\sigma(x_k) = \alpha(\nabla \log p_\sigma(\mathbf{x}_k + \zeta_k) - \nabla \log p(\mathbf{x}_k))$, with $\alpha > 0, \zeta_k \sim \mathcal{N}(0, \sigma^2 \mathbf{I}_d)$. By definition we have $\mathbb{E}_{\zeta_k}(\xi_k) = 0$ and

$$\mathbf{x}_{k+1} = \mathbf{x}_k - \delta_k (\nabla \mathcal{J}_\sigma(x_k) + \xi_k).$$

We have, by using Assumption 3.2 and $R_{\mathbf{K}} = \sup_{\mathbf{x} \in \mathbf{K}} \|\mathbf{x}\|$

$$
\begin{aligned}
\mathbb{E}(\|\xi_k\|^2 |\mathbf{x}_k) &= \alpha^2 \mathbb{E}(\|\nabla \log p_\sigma(\mathbf{x}_k + \zeta_k) - \mathbb{E}_{\zeta \sim \mathcal{N}(0,\sigma^2 \mathbf{I}_d)} \left(\nabla \log p_\sigma(\mathbf{x}_k + \zeta)\right)\|^2 |\mathbf{x}_k) \\
&= \alpha^2 \left(\mathbb{E}(\|\nabla \log p_\sigma(\mathbf{x}_k + \zeta)\|^2 |\mathbf{x}_k) - \|\mathbb{E}\left(\nabla \log p_\sigma(\mathbf{x}_k + \zeta)\right)\|^2 |\mathbf{x}_k)\right) \\
&\leq \alpha^2 \mathbb{E}(\|\nabla \log p_\sigma(\mathbf{x}_k + \zeta)\|^2 |\mathbf{x}_k) \\
&\leq \alpha^2 \mathbb{E}(B^2 \sigma^{2\beta} \left(1 + \|\mathbf{x}_k + \zeta\|^r\right)^2 |\mathbf{x}_k) \\
&\leq \alpha^2 B^2 \sigma^{2\beta} \mathbb{E}((1 + (R_{\mathbf{K}} + \|\zeta\|)^r)^2 |\mathbf{x}_k).
\end{aligned}
$$

As $\mathbb{E}((1 + (R_{\mathbf{K}} + \|\zeta\|)^r)^2 |\mathbf{x}_k) = \mathbb{E}((1 + (R_{\mathbf{K}} + \|\zeta\|)^r)^2) = C_{r,\sigma,d,R_{\mathbf{K}}}$ is a constant that only depends on r, σ, d and $R_{\mathbf{K}}$, we get $\mathbb{E}(\|\xi_k\|^2) \leq \tilde{C}$, with $\tilde{C} < +\infty$, a constant of k. Then $\sum_{k \in \mathbb{N}} \delta_k^2 \mathbb{E}(\|\xi_k\|^2) \leq \tilde{C} \sum_{k \in \mathbb{N}} \delta_k^2 < +\infty$, by Assumption 3.3.

By following the classical Robbins-Monro (RM) situation (see for example Metivier & Priouret (1984, Part II-D)), we deduce from the Doob inequality (which holds because $\sum_{k=n}^l \xi_k$ is a martingale),

$$
\mathbb{E}(\sup_{n \leq l \leq m} \|\sum_{k=n}^l \delta_k \xi_k\|) \leq 4 \sum_{k=n}^m \delta_k^2 \mathbb{E}(\|\xi_k\|^2).
$$

By the monotone convergence theorem, this implies

$$
\mathbb{E}(\sup_{n \leq l} \|\sum_{k=n}^l \delta_k \xi_k\|) \leq 4 \sum_{k=n}^\infty \delta_k^2 \mathbb{E}(\|\xi_k\|^2) \leq 4 C_{r,\sigma,d,R_{\mathbf{K}}} \sum_{k=n}^\infty \delta_k^2.
$$

Thus the sequence $(\sup_{n \leq l} \|\sum_{k=n}^l \delta_k \xi_k\|)_n$ tends to zero in L^1 and also almost surely (because it is non-increasing). This allows us to conclude that our process verifies (Doucet & Tadic, 2017, Assumption 2.2) almost surely.

Also, p_σ is $\mathcal{C}^\infty$ by convolution with a Gaussian and then $\log p_\sigma$ is also $\mathcal{C}^\infty$. Again, by convolution, $\mathcal{R}_\sigma$ is clearly $\mathcal{C}^\infty$. By Assumption 3.4, $\mathcal{J} = \mathcal{F}(\cdot, \mathbf{y}) + \mathcal{R}_\sigma$ is $\mathcal{C}^\infty$. So (Doucet & Tadic, 2017, Assumption 2.b) is verified. We can thus apply (Doucet & Tadic, 2017, Theorem 2.1. (ii)) with $\eta = 0$ to our process, which concludes the proof.

D.1.4. PROOF OF PROPOSITION 3.4

We only need to notice that Assumption 3.5 implies Assumption 3.2 and apply Proposition 3.3. One can directly apply (De Bortoli, 2022, Lemma C.1) but we give a proof here for the sake of clarity. Observe that

$$
p_\sigma(\mathbf{x}) = (p \star \mathcal{N}_\sigma)(\mathbf{x}) = \int_{\mathbb{R}^d} p(\mathbf{y}) \mathcal{N}_\sigma(\mathbf{x} - \mathbf{y}) d\mathbf{y} = \int_{\mathcal{B}(0,R)} p(\mathbf{y}) \mathcal{N}_\sigma(\mathbf{x} - \mathbf{y}) d\mathbf{y}.
$$

By taking $R > 0$ such that $\mathcal{M} \subset \mathcal{B}(0, R)$, we get

$$
\begin{aligned}
\|\nabla \log p_\sigma(\mathbf{x})\| = \left\|\frac{\nabla p_\sigma(\mathbf{x})}{p_\sigma(\mathbf{x})}\right\| &= \frac{1}{p_\sigma(\mathbf{x})} \left\|\int_{\mathcal{B}(0,R)} p(\mathbf{y}) \frac{\mathbf{x} - \mathbf{y}}{\sigma^2} \mathcal{N}_\sigma(\mathbf{x} - \mathbf{y}) d\mathbf{y}\right\| \\
&\leq \frac{1}{p_\sigma(\mathbf{x})} \int_{\mathcal{B}(0,R)} p(\mathbf{y}) \frac{\|\mathbf{x} - \mathbf{y}\|}{\sigma^2} \mathcal{N}_\sigma(\mathbf{x} - \mathbf{y}) d\mathbf{y} \\
&\leq \frac{\|\mathbf{x}\| + R}{\sigma^2},
\end{aligned}
$$

so Assumption 3.2 holds with $B = \max(1, R)$, $\beta = -2$ and $r = 1$.

D.1.5. PROOF OF PROPOSITION 3.5

We can define η_k by $\eta_k = \mathbb{E}(\xi_k)$ and $\gamma_k = \xi_k - \mathbb{E}(\xi_k)$. So we have $\xi_k = \gamma_k + \eta_k$ and $\mathbb{E}(\gamma_k) = 0$. We will apply (Doucet & Tadic, 2017, Theorem 2.1. (ii)). Assumption 2.1. and 2.3.b. of this paper are already verified because of Assumption 3.3,

Assumption 3.4, and Assumption 3.6. Then, using the definition of ξ_k and Assumption 3.7, we have

$$\mathbb{E}(\|\gamma_k\|^2|\mathbf{x}_k) = \mathbb{E}(\|\xi_k - \mathbb{E}(\xi_k)\|^2|\mathbf{x}_k)$$

$$= \frac{\alpha^2}{\sigma^4}\mathbb{E}\left(\|D_\sigma(\mathbf{x}_k + \zeta_k) - \mathbf{x}_k - \zeta_k - \mathbb{E}_\zeta\left(D_\sigma(\mathbf{x}_k + \zeta) - \mathbf{x}_k - \zeta\right)\|^2|\mathbf{x}_k\right)$$

$$= \frac{\alpha^2}{\sigma^4}\mathbb{E}\left(\|D_\sigma(\mathbf{x}_k + \zeta) - \zeta - \mathbb{E}_\zeta\left(D_\sigma(\mathbf{x}_k + \zeta) - \zeta\right)\|^2|\mathbf{x}_k\right)$$

$$= \frac{\alpha^2}{\sigma^4}\left(\mathbb{E}\left(\|D_\sigma(\mathbf{x}_k + \zeta) - \zeta\|^2|\mathbf{x}_k\right) - \|\mathbb{E}_\zeta\left(D_\sigma(\mathbf{x}_k + \zeta) - \zeta|\mathbf{x}_k\right)\|^2\right)$$

$$\leq \frac{\alpha^2}{\sigma^4}\mathbb{E}\left(\|D_\sigma(\mathbf{x}_k + \zeta) - \zeta\|^2|\mathbf{x}_k\right)$$

$$\leq \frac{2\alpha^2}{\sigma^4}\left(\mathbb{E}\left(\|D_\sigma(\mathbf{x}_k + \zeta)\|^2|\mathbf{x}_k\right) + \mathbb{E}\left(\|\zeta\|^2|\mathbf{x}_k\right)\right)$$

$$\leq \frac{2\alpha^2}{\sigma^4}\left(\mathbb{E}\left((\|\mathbf{x}_k + \zeta\| + C\sigma)^2|\mathbf{x}_k\right) + d\sigma^2\right)$$

$$\leq \frac{2\alpha^2}{\sigma^4}\left(2\mathbb{E}\left((\|\mathbf{x}_k\| + \|\zeta\|)^2|\mathbf{x}_k\right) + 2C^2\sigma^2 + d\sigma^2\right)$$

$$\leq \frac{2\alpha^2}{\sigma^4}\left(4R_{\mathbf{K}}^2 + 4d\sigma^2 + 2C^2\sigma^2 + d\sigma^2\right) < +\infty.$$

By using the Doob inequality as in Appendix D.1.3, we have demonstrated the first part of Assumption 2.2. of (Doucet & Tadic, 2017).

Now we study the asymptotic behavior of η_k, for $R > 0$

$$\|\eta_k\| = \|\mathbb{E}(\xi_k)\| = \|\mathbb{E}(\frac{\alpha}{\sigma^2}\left(D_\sigma(\mathbf{x}_k + \zeta_k) - \mathbf{x}_k - \zeta_k\right) - \frac{\alpha}{\sigma^2}\mathbb{E}_\zeta\left(D_\sigma^\star(\mathbf{x}_k + \zeta) - \mathbf{x}_k - \zeta\right)\|$$

$$= \|\frac{\alpha}{\sigma^2}\mathbb{E}\left(D_\sigma(\mathbf{x}_k + \zeta) - D_\sigma^\star(\mathbf{x}_k + \zeta)\right)\|$$

$$= \frac{\alpha}{\sigma^2}\|\int_{\zeta \in \mathbb{R}^d}(D_\sigma(\mathbf{x}_k + \zeta) - D_\sigma^\star(\mathbf{x}_k + \zeta))\mathcal{N}(\zeta; 0, \sigma^2\mathbf{I}_d)d\zeta\|$$

$$\leq \frac{\alpha}{\sigma^2}\int_{\|\mathbf{x}_k + \zeta\| \leq R}\|D_\sigma(\mathbf{x}_k + \zeta) - D_\sigma^\star(\mathbf{x}_k + \zeta)\|\mathcal{N}(\zeta; 0, \sigma^2\mathbf{I}_d)d\zeta$$

$$+ \frac{\alpha}{\sigma^2}\int_{\|\mathbf{x}_k + \zeta\| > R}\|D_\sigma(\mathbf{x}_k + \zeta) - D_\sigma^\star(\mathbf{x}_k + \zeta)\|\mathcal{N}(\zeta; 0, \sigma^2\mathbf{I}_d)d\zeta$$

$$\leq \frac{\alpha}{\sigma^2}\left(M(R) + \int_{\|\mathbf{x}_k + \zeta\| > R}\left(\|D_\sigma(\mathbf{x}_k + \zeta)\| + \|D_\sigma^\star(\mathbf{x}_k + \zeta)\|\right)\mathcal{N}(\zeta; 0, \sigma^2\mathbf{I}_d)d\zeta\right)$$

$$\leq \frac{\alpha}{\sigma^2}\left(M(R) + \int_{\|\mathbf{x}_k + \zeta\| > R}\left(2\|\mathbf{x}_k + \zeta\| + 2C\sigma\right)\mathcal{N}(\zeta; 0, \sigma^2\mathbf{I}_d)d\zeta\right)$$

$$\leq \frac{\alpha}{\sigma^2}\left(M(R) + \int_{\|\mathbf{x}_k + \zeta\| > R}\left(2\|\mathbf{x}_k\| + 2\|\zeta\| + 2C\sigma\right)\mathcal{N}(\zeta; 0, \sigma^2\mathbf{I}_d)d\zeta\right).$$

For $R = R_{\mathbf{K}} + 1$, if $\|\mathbf{x}_k + \zeta\| > R$, then $\|\zeta\| > 1$ because $\|\mathbf{x}_k\| \leq R_{\mathbf{K}}$. This implies that

$$\|\eta_k\| \leq \frac{\alpha}{\sigma^2}\left(M(R) + \int_{\|\zeta\| > 1}\left(2\|\mathbf{x}_k\| + 2\|\zeta\| + 2C\sigma\right)\mathcal{N}(\zeta; 0, \sigma^2\mathbf{I}_d)d\zeta\right).$$

Then if $\|\zeta\| > 1$, $\|\zeta\|^2 \le \frac{1}{2} + \frac{\|\zeta\|^2}{2}$. So $\mathcal{N}(\zeta; 0, \sigma^2 \mathbf{I}_d) \le 2^{-\frac{d}{2}} \mathcal{N}(\zeta; 0, 2\sigma^2 \mathbf{I}_d) \exp\left(-\frac{1}{4\sigma^2}\right)$. So

$$
\|\eta_k\| \le \frac{\alpha}{\sigma^2} \left(M(R) + 2^{-\frac{d}{2}} \exp\left(-\frac{1}{4\sigma^2}\right) \int_{\|\zeta\|>1} (2\|\mathbf{x}_k\| + 2\|\zeta\| + 2C\sigma)\,\mathcal{N}(\zeta; 0, 2\sigma^2 \mathbf{I}_d)d\zeta \right)
$$

$$
\le \frac{\alpha}{\sigma^2} \left(M(R) + 2^{-\frac{d}{2}} \exp\left(-\frac{1}{4\sigma^2}\right) \left(2R_{\mathbf{K}} + 2\sqrt{d}\sigma + 2C\sigma \right) \right).
$$

This shows that

$$
\eta = \limsup_{k \in \mathbb{N}} \|\eta_k\| \underset{\sigma \to 0}{\le} \frac{\alpha}{\sigma^2} M(R) + o(\sigma).
$$

Then Assumption 2.2. of (Doucet & Tadic, 2017) is verified and (Doucet & Tadic, 2017, Theorem 2.1. (ii)) applies.

E. Discussion on Assumptions

In this section, we explicitly list and comment all the assumptions used for results presented in Section 3. We detail in which case each assumption is verified, especially in the PnP context. A special discussion on the boundedness assumption is postponed in Appendix H.

- Assumption 3.1 **(a)** implies that the score is non-localized with mass $p > 0$ everywhere, and the density is bounded and Lipschitz. Moreover the density is C^1, which is necessary to define the score $\nabla \log p$. Note that image distributions are typically supported on a compact set, so their score $\nabla \log p$ is not defined everywhere. Thus, this hypothesis is not realistic in imaging context but it is necessary to realize an analysis of the score $\nabla \log p$. However, if $\log p$ is taken among classical image models (e.g. related to Total Variation or Tychonov regularization), it is likely that Assumption 3.1 **(a)** is verified. In particular, we conjecture that this hypothesis is true for the PnP regularization with explicit potential proposed in (Hurault et al., 2022a).

- Assumption 3.1 **(b)** is verified for Gaussian, Gaussian Mixture or Cauchy distributions. This is a technical assumption to allow a critical point analysis. This hypothesis is non-restrictive because q can be arbitrarily large.

- Assumption 3.2 is implied by Assumption 3.5 with $r = 1$ and $\beta = -2$ (De Bortoli, 2022). We provided a simplified proof of this result in Section D.1.4. De Bortoli et al. (2021) have also proved Assumption 3.2 with $r = 1$ and $\beta = 0$ under the Assumption 3.1(**b**) with $q = 1$ and an extra assumption difficult to verify: there exist $m_0 > 0$ and $d_0 \ge 0$ such that $\forall \mathbf{x} \in \mathbb{R}^d$, $\langle \nabla \log p(\mathbf{x}), \mathbf{x} \rangle \le -m_0 \|\mathbf{x}\|^2 + d_0 \|\mathbf{x}\|$. However, to our knowledge, there is no general study of the score approximation $\nabla \log p_\sigma$ for a general smooth distribution or a non-bounded and non-smooth distribution. Assumption 3.2 seems non-restrictive because r can be arbitrarily large and β highly negative.

- Assumption 3.3 is standard in stochastic gradient descent (Metivier & Priouret, 1984; Benaïm, 1999; Doucet & Tadic, 2017). A stochastic process imposes a decreasing step-size to ensure convergence. If the step-size is constant, the noise (with a non-decreasing variance) that is added at each step makes the process explore even after a large number of steps, so there is no convergence to expect. In practice, we take a constant step-size to converge faster. In fact, we run the algorithm for only a small number of iterations (few hundreds).

- Assumption 3.4 makes the data-fidelity $\mathcal{F}$ smooth. It is verified for every linear inverse problem with a Gaussian noise. Assumption 3.4 can also be verified in the case of a non-Gaussian noise, for instance with a Fischer-Tippett noise (Deledalle et al., 2017). However, some specific data-fidelity terms do not verified this assumption, such as the ones related to salt and pepper noise (Chan et al., 2005; Nikolova, 2004) or Laplacian noise (Huang et al., 2017), or other data-fidelity used for image segmentation (Chan et al., 2006).

- Assumption 3.5 has been used in the same form as presented by De Bortoli (2022). This assumption is validated by real image distributions, typically encoded within a finite range, such as $[0, 1]$ or $[0, 255]$. Traditionally, the manifold hypothesis asserts that the distribution of images is confined to a low-dimensional manifold. Fefferman et al. (2016) conducted tests to evaluate this hypothesis, and more recently, Brown et al. (2022) have focused on analyzing image distributions. However, within the scope of our study, we refer to the *manifold hypothesis* as defined in (De Bortoli, 2022), suggesting that images are supported within a compact set. This diverges from the conventional manifold hypothesis, constituting a relaxed version of the original assumption.
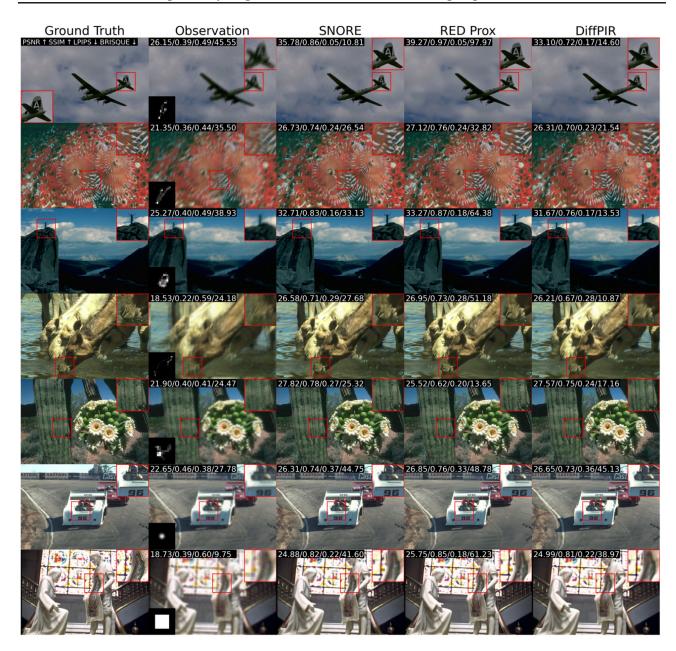
Figure 5. Results of SNORE, RED and DiffPIR algorithms on deblurring tasks with various blur kernels and various images from the dataset CBSD10. The input noise is $\sigma_{\mathbf{y}} = 10/255$. Kernels have various sizes and have been resized for clarity of the figure. The last kernel, uniform of size 9×9 is plotted on a black background for visibility.

- Assumption 3.6 quantifies the uniform distance between the exact and the inexact denoiser. If the activation function of the denoiser is $\mathcal{C}^\infty$, then the denoiser is $\mathcal{C}^\infty$. Laumont et al. (2023) make a similar assumption. In the literature, other types of assumptions have been made to control the error of the inexact MMSE (see for instance (Shoushtari et al., 2023, Assumption 5)). We choose to take this form of assumption because Laumont et al. (2022, Proposition 4) have proved that Assumption 3.6 can be ensured if the denoiser is learned with the Noise2Noise loss (Lehtinen et al., 2018).

- Assumption 3.7 is natural for a well trained denoiser. It means that the denoiser D_σ can only modify the image by a quantity bounded by the level of noise σ. As a practical example, a bounded denoiser (Chan et al., 2016, Definition 1) verifies Assumption 3.7. However, Assumption 3.7 is difficult to verify in practice (especially outside the training domain). To our knowledge there is no theoretical argument stronger than intuition to support this assumption. However,

Assumption 3.7 is necessary to analyze the stability of SNORE with an inexact denoiser, presented in Proposition 3.5.

F. Additional Experiments

In this section, we provide more details about our experiments. First we present general technical details. Then, we present more experiment for image deblurring including parameters influence and uncertainty of SNORE. Moreover, we give all parameter setting for image deblurring and inpainting. Finally, a preliminary experiment on image super-resolution is shown.

Metrics We use four metrics to evaluate our results. Structural SIMilarity (SSIM) and Peak Signal to Noise Ratio (PSNR) are two common distortions metrics. BRISQUE is a no-reference metric based on natural scene statistics that was proved to correlate well with human perception (Mittal et al., 2012). This metric gives a score between 0 (best) and 100 (worst). We use the Python library "brisque", with which we sometimes observe some incoherence with some outlier outputs (smaller than 0 or larger than 100). These outputs where rare so we kept this standard implementation for reproducibility purpose. LPIPS (Zhang et al., 2017) is another perceptual metric that compares the original image and the reconstructed one by measuring their differences in terms of deep features. As suggested in (Ren et al., 2023), looking at such perceptual metrics is relevant as they are correlated with human perception. We use the Python library "lpips" to compute this metric.

Denoiser We use the denoiser proposed by (Hurault et al., 2022a) based on the DRUNet (Zhang et al., 2021) trained on a dataset of natural images composed of Berkeley segmentation dataset (CBSD) (Martin et al., 2001), Waterloo Exploration dataset (Ma et al., 2017), DIV2K dataset (Agustsson & Timofte, 2017) and Flick2K (Lim et al., 2017). We take the training weights of (Hurault et al., 2022a). In order to better analyze the advantages and drawbacks of other PnP and RED approaches, this denoiser is used for all our experiments and comparisons.

RED Prox We name *RED Prox* (Algorithm 4) the splitting algorithm with a gradient descent step on the regularization and a proximal step on the data-fidelity. Just like RED (Algorithm 1), RED Prox minimizes Problem (1) with the regularization $\mathcal{P}_\sigma$, defined in Equation (2). That kind of splitting algorithm is commonly used in the PnP field (Ryu et al., 2019). As proposed in (Hurault et al., 2022a), thanks to the special form of gradient-step denoiser, we implement a backtracking procedure for RED and RED Prox.

Algorithm 4 RED Prox

1: **input:** $\mathbf{x}_0 \in \mathbb{R}^d, \sigma > 0, \alpha > 0, \delta > 0, N \in \mathbb{N}$
2: **for** $k = 0, 1, \ldots, N-1$ **do**
3:     $\mathbf{z}_k \leftarrow \mathbf{x}_k - \frac{\alpha\delta}{\sigma^2}(\mathbf{x}_k - D_\sigma(\mathbf{x}_k))$
4:     $\mathbf{x}_{k+1} \leftarrow \mathrm{Prox}_{\delta\mathcal{F}}(\mathbf{z}_k)$
5: **end for**

SNORE Prox We name SNORE Prox the algorithm detailed in Algorithm 5. A Proximal descent step is computed on the data-fidelity instead of a gradient-descent step. Note that our convergence analysis of Section 3 does not apply to this algorithm. We test this algorithm for a comparison with the original SNORE (Algorithm 3). Experimentally, we observe that both algorithms reach similar performance (see Table 1 or Table 2).

Algorithm 5 Annealed SNORE Prox

1: **input:** $\mathbf{x}_0 \in \mathbb{R}^d, m \in \mathbb{N}, \delta > 0, \sigma_0 > \sigma_1 > \cdots > \sigma_{m-1} \approx 0, \alpha_0, \alpha_1, \ldots, \alpha_{m-1} > 0, N_0, N_1, \ldots, N_{m-1} \in \mathbb{N}$
2: **for** $i = 0, 1, \ldots, m-1$ **do**
3:     **for** $k = 0, 1, \ldots, N_i - 1$ **do**
4:         $\boldsymbol{\epsilon} \leftarrow \mathcal{N}(0, \mathbf{I}_d)$
5:         $\tilde{\mathbf{x}}_k \leftarrow \mathbf{x}_k + \sigma_i \boldsymbol{\epsilon}$
6:         $\mathbf{z}_k \leftarrow \mathbf{x}_k - \frac{\alpha_i \delta}{\sigma_i^2}(\mathbf{x}_k - D_{\sigma_i}(\tilde{\mathbf{x}}_k))$
7:         $\mathbf{x}_{k+1} \leftarrow \mathrm{Prox}_{\delta\mathcal{F}}(\mathbf{z}_k)$
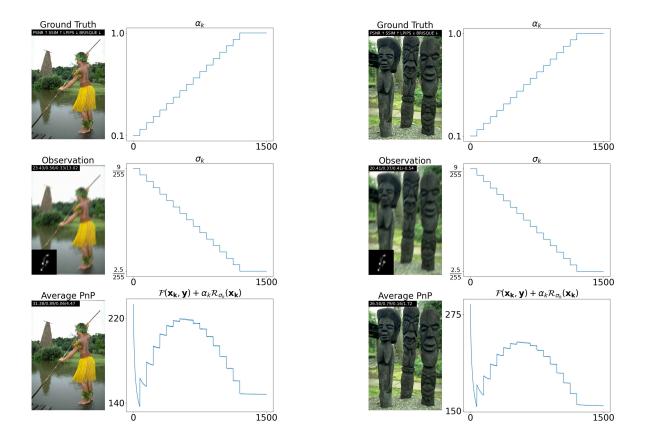8:   **end for**
9: **end for**

Figure 6. Image deblurring with an input noise level $\sigma_{\mathbf{y}} = 5/255$, annealing parameters α_k and σ_k and the potential $\mathcal{F} + \alpha_k \mathcal{R}_{\sigma_k}$ which is optimized by Annealed SNORE algorithm (Algorithm 3). There are 16 annealing levels in the first 1200 iterations and 300 iterations with the final parameters. Note that the potential is minimized at each level for fixed α_k and σ_k. The optimization can be understood in two steps. In the first one, σ_k is large and α_k is small to deblur the image. Then a refinement is realized with α_k large and σ_k small, to generate a realistic image.

DiffPIR We compare our algorithm to state-of-the-art methods, including a Diffusion Model. Among existing methods in the field of Diffusion Models, we choose to test DiffPIR (Zhu et al., 2023). DiffPIR is a recent algorithm which makes a connection between PnP and Diffusion Models. We use the implementation of the Python library DeepInverse modified to add a time $t_{\text{start}} < T$ such as proposed by the authors (Zhu et al., 2023, Section 4.4). We need to add this parameter t_{start} as the denoiser is not trained to generate relevant outputs for highly noisy images (compared to neural networks used for diffusion models), since the model has only been trained for noise with standard deviations in the set $[0, 50]/255$.

The parameter schemes $(\bar{\alpha}_t)_{0<t<T}$, $(\sigma_t)_{0<t<T}$ and $(\rho_t)_{0<t<T}$ are described in (Zhu et al., 2023). For DiffPIR, we choose the following parameters for all inverse problems: $\zeta = 0.8$, $T = 1000$, $t_{\text{start}} = 200$ and $\lambda = 0.13$.

PnP SGD We compare our algorithm to an other stochastic PnP method, PnP SGD (Laumont et al., 2023), which approximates the maximum of the Posterior Law of Problem 1. In the implementation of the method, we need to add an other parameter in PnP SGD, $\beta > 0$, to control the power of the additive noise at each iteration and optimize the performance of the algorithm. Adding the $\beta > 0$ parameter does not change the analysis of the method. As noticed by Laumont et al. (2023), keeping a fixed step-size allows us to obtain the best performance, so we decided to keep the step-size $\delta > 0$ fixed. We give the pseudo-code of PnP SGD in Algorithm 7 and the used parameters for deblurring in Table 4. We do not succeed to make

---

Algorithm 6 DiffPIR (Zhu et al., 2023)

1: **input:** D_σ, $T > 0$, $\mathbf{y} \in \mathbb{R}^q$, $0 < t_{\text{start}<T}$, $\zeta > 0$, $(\beta_t)_{0<t<T}$, $\lambda > 0$
2: Initialize $\boldsymbol{\epsilon}_{t_{\text{start}}} \sim \mathcal{N}(0, \mathbf{I}_d)$, pre-calculate $(\bar{\alpha}_t)_{0<t<T}$, $(\sigma_t)_{0<t<T}$ and $(\rho_t)_{0<t<T}$
3: $x_{t_{\text{start}}} = \sqrt{\bar{\alpha}_{\text{start}}}\mathbf{y} + \sqrt{1 - \bar{\alpha}_{\text{start}}}\boldsymbol{\epsilon}_{t_{\text{start}}}$
4: **for** $t = t_{\text{start}}, t_{\text{start}} - 1, \ldots, 1$ **do**
5:     $x_0^t \leftarrow D_{\sigma_t}(x_t)$
6:     $\hat{x_0}^t \leftarrow \text{Prox}_{2\mathcal{F}(\cdot,\mathbf{y})/\rho_t}(x_0^t)$
7:     $\hat{\boldsymbol{\epsilon}} \leftarrow \left(x_t - \sqrt{\bar{\alpha}_t}\hat{x_0}^t\right)/\sqrt{1 - \bar{\alpha}_t}$
8:     $\boldsymbol{\epsilon}_t \leftarrow \mathcal{N}(0, \mathbf{I}_d)$
9:     $x_{t_1} \leftarrow \sqrt{\bar{\alpha}_t}\hat{x_0}^t + \sqrt{1 - \bar{\alpha}_t}\left(\sqrt{1 - \zeta}\hat{\boldsymbol{\epsilon}} + \sqrt{\zeta}\boldsymbol{\epsilon}_t\right)$
10: **end for**

Inverse Problem	RED	RED Prox	SNORE	SNORE Prox	DiffPIR
Deblurring	5	2	67	41	1
Inpainting	39	40	40	40	1

Table 3. Computational time in second (averaged on 4 images) for various methods and inverse problems on a GPU NVIDIA Quadro RTX 8000.

the method competitive for image inpainting so we do not include in this paper the output of PnP SGD for inpainting.

Algorithm 7 PnP SGD

1: **input:** $\mathbf{x}_0 \in \mathbb{R}^d$, $\sigma > 0$, $\alpha > 0$, $\beta > 0$, $\delta > 0$, $N \in \mathbb{N}$
2: **for** $k = 0, 1, \ldots, N - 1$ **do**
3:     $\mathbf{z}_k \sim \mathcal{N}(0, \text{Id})$
4:     $\mathbf{x}_{k+1} \leftarrow \mathbf{x}_k - \delta\nabla\mathcal{F}(\mathbf{x}_k) - \frac{\alpha\delta}{\sigma^2}(\mathbf{x}_k - D_\sigma(\mathbf{x}_k)) + \beta\delta\mathbf{z}_k$
5: **end for**

Computational time On Table 3, we compare the computational time of various methods. For deblurring, SNORE and SNORE Prox are slow compare to other methods. This is due to the number of iterations require at each parameters level to converge, leading to a large number of iteration (1500). We observe on Figure 9 that with less annealing level, metrics performance are similar. However, we observe a qualitative impact of this parameter. For inpainting, out method outperform RED and RED Prox with a fixed number of iteration. Only DiffPIR remains faster. For this inverse problem, we have observed that a smaller number of iteration (500) is sufficient.

Computing all the necessary experiments to generate Table 1 requires 9 hours and 40 minutes on a GPU NVIDIA Quadro RTX 8000. Similarly, generating Table 2 requires 8 hours of computation on a GPU NVIDIA Quadro RTX 8000. The whole computational resources used for this paper are 17 hours and 40 minutes of computation on a GPU NVIDIA Quadro RTX 8000.

F.1. Deblurring

In this part, we give more details on our experiments for image deblurring. We also discuss the influence of the parameters m, α_{m-1} and σ_{m-1} on Annealing SNORE outputs (Algorithm 3).

Parameters setting In Table 4, we give the values of the different parameters used in our experiments.

More deblurring results For a better qualitative comparison between methods, we present in Figure 5 several image deblurring results obtained with various blur kernels. In Figure 6, we present the parameters evolution during the optimization process for the annealing SNORE algorithm (Algorithm 3). The annealing levels between α_0, α_{m-1} and σ_0, σ_{m-1} are chosen by a linear interpolation.

Next we focus on the choice of the final parameters $(\alpha_{m-1}, \sigma_{m-1})$, because these parameters define the optimization problem from which a critical point is finally computed. The preliminary resolution of optimization problems with parameters

Parameters	RED	RED Prox	SNORE	SNORE Prox	PnP SGD
α (motion blur, $\sigma_{\mathbf{y}} < 20/255$)	0.1	0.2			0.5
α (motion blur, $\sigma_{\mathbf{y}} \geq 20/255$)	0.1	0.3			0.5
α (fixed blur, $\sigma_{\mathbf{y}} < 20/255$)	0.075	0.2			0.5
α (fixed blur, $\sigma_{\mathbf{y}} \geq 20/255$)	0.075	0.3			0.5
α_0			0.1	0.1	
α_{m-1}			1	1	
$\sigma/\sigma_{\mathbf{y}}$ ($\sigma_{\mathbf{y}} < 20/255$)	1.8	1.4			1.
$\sigma/\sigma_{\mathbf{y}}$ ($\sigma_{\mathbf{y}} \geq 20/255$)	1.8	1.8			1.
$\sigma_0/\sigma_{\mathbf{y}}$			1.8	1.8	
$\sigma_{m-1}/\sigma_{\mathbf{y}}$			0.5	0.5	
$\max_{\text{itr}}$	100	100	1500	1500	1000
δ	$1/\alpha$	$1/\alpha$	0.1	0.1	0.1
β					0.01

Table 4. Parameters setting for image deblurring for the different implemented methods.

$(\alpha_i, \sigma_i)_{i<m-1}$ can be understood as a procedure to compute a relevant initialization for the last annealing level. We choose to optimize at different levels (α_i, σ_i) in order to compute a good approximation of a critical point at fixed (α_i, σ_i) and to fit with the setting of Proposition 3.2.

Influence of α_{m-1}   On Figure 7, we study the influence of the final weighting parameter α_{m-1} on the restoration provided by the SNORE algorithm. One can see that if α_{m-1} is too small, the problem is less regularized so there is a residual noise. On the other hand, if α_{m-1} is too large, the restored image is very flat.
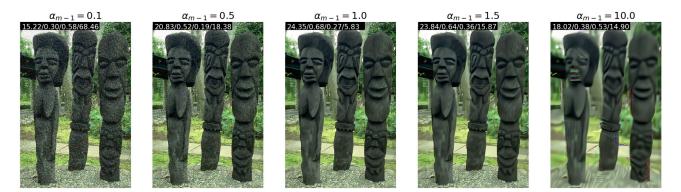

Figure 7. Influence of the parameter α_{m-1} in SNORE algorithm on deblurring with an input noise of level $\sigma_{\mathbf{y}} = 10/255$.

Influence of σ_{m-1}   On Figure 8, we illustrate the influence of the final denoiser parameter σ_{m-1} on the restoration obtained with the SNORE algorithm. One can see that if σ_{m-1} is too small, the problem is less regularized and a residual noise is present. On the other side, if σ_{m-1} is too large, the restored image is too flat.

The influence of parameters σ_{m-1} and α_{m-1} is therefore similar. However, we observed experimentally that having these two free parameters allows to obtain better restoration results.

Influence of m   On Figure 9, we observe the influence of the number of annealing levels m on the quality of the reconstruction. Metrics are not sensitive to this parameter but we observe on images that some artifacts are reduced with additional annealing levels. Our experiments with the SNORE algorithm suggest that with more annealing levels, the algorithm performs better to inverse the degradation and less artifacts are visible. Images of Figure 9 support this claim. Note that local artifacts do not seem to have a significant influence on the metric values.

On the step-size δ   Laumont et al. (2023) proposes to use a two-phase gradient-descent, a first one with $\delta_0 > 0$ fixed for finite number of iteration then a second phase with decreasing step-size $\delta_k = \delta_0/k^\gamma$ (in their experiments $\gamma = 0.8$). They ensure that Assumption 3.3 is verified. However, they observe that the second phase has no impact on the output of
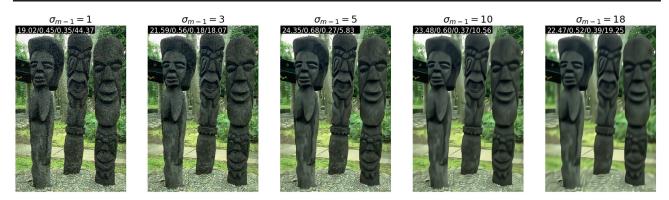
Figure 8. Influence of the parameter σ_{m-1} in SNORE algorithm on deblurring with an input noise of level $\sigma_{\mathbf{y}} = 10/255$.

Parameters	RED	RED Prox	SNORE	SNORE Prox
α	0.15	0.15		
α_0			0.15	0.15
α_{m-1}			0.4	0.15
σ	10/255	10/255		
σ_0			50/255	50/255
σ_{m-1}			5/255	5/255
n_{init}	10	100		
$\max_{\text{itr}}$	500	500	500	500
δ (initialization)	$1/\alpha$	0.5	0.5	1

Table 5. Parameters setting for image inpainting for the different implemented methods.

the algorithm. We try the same framework (with various $\gamma \in]\frac{1}{2}, 1]$) and also observe that the second phase is useless. For efficiency, we choose to only compute the first phase with fixed $\delta > 0$.

F.2. Uncertainty of SNORE

Seed sensitivity On Figure 10, we illustrate the robustness of of SNORE to stochasticity, by running the algorithm with different random seed and looking at the standard deviation of the corresponding reconstructions. We observe that our restoration has a low variability and thus a low uncertainty. This is a crucial behavior for the reliability of the algorithm. This experiment suggests that the restoration, and especially the reconstructed structures, is stable and thus reliable.

Initialization sensitivity On Figure 11, three different initialization are shown for a deblurring task. We notice that the SNORE algorithm does not diverge, even with a random initialization. This is a remarkable property since we face a non-convex optimization for which the initialization of the algorithm is crucial. With a pure noise initialization, artifacts are nevertheless present on the restored images. We also observe that the gap between the algorithm run with an oracle initialization or the observation is tight. This observation suggests that the observation is a good initialization for the SNORE algorithm.

F.3. Inpainting

The inpainting mask is created by sampling a Bernouilli law of success probability $p = 0.5$ for each pixel of the image.

Parameters setting On Table 5, we detail the practical choice of parameters we made. As suggested by (Hurault et al., 2022a), for image inpainting, we start by running the algorithm with a larger value $\sigma = 50/255$ for a number of iterations n_{init}. This allows the algorithm to tackle the ill-posedness of the inpainting task.
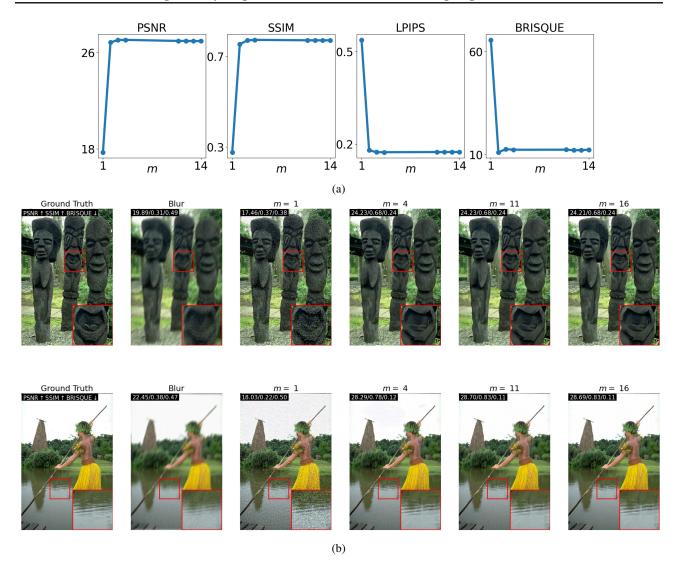
Figure 9. Influence of the number of annealing levels m on the reconstruction with SNORE algorithm for a motion blur with a noise of standard deviation $\sigma_{\mathbf{y}} = 10/255$ and a fixed number of 1500 iterations. *Figure 9a:* Metrics evolution with different m. One can note that the number of annealing levels m has a low influence on metrics values. *Figure 9b:* Reconstructed images with the SNORE algorithm for different numbers of annealing levels m. One can note that, the larger m is, the less artifacts are visible. However, compute a larger number of annealing levels impose to compute a larger number of iterations, to converge for each annealing parameters.

More results On Figure 12, we present various results of image inpainting for a better qualitative comparison between methods.

On Figure 13, we provide more results of SNORE algorithm on various inpainting problems (with of proportion $p \geq 0.5$ of masked pixels). As expected, we observe that the quality of the restoration decrease with the proportion of missing pixels.

F.4. Super-resolution

For image super-resolution, the observation $\mathbf{y} \in \mathbb{R}^q$ is the low-resolution version of $\mathbf{x} \in \mathbb{R}^d$, obtained by $\mathbf{y} = \mathbf{SHx} + \mathbf{n}$, where $\mathbf{H}$ is an anti-aliasing blur kernel and $\mathbf{n} \sim \mathcal{N}(0, \sigma_{\mathbf{y}})$. $\mathbf{S}$ is the standard down-sampling matrix with the super-resolution factor s_f. The data-fidelity is given by $\mathcal{F}(\mathbf{x}, \mathbf{y}) = \frac{1}{2\sigma_{\mathbf{y}}} \|\mathbf{SHx} - \mathbf{y}\|^2$ and its proximal operator (Zhao et al., 2015) by

$$\text{Prox}_{\delta\mathcal{F}}(\mathbf{z}) = \hat{\mathbf{z}} - \frac{1}{s_f^2} \mathbf{F}^{\star} \bar{\mathbf{\Lambda}}^{\star} \left(\mathbf{I}_q + \frac{\delta}{s_f^2} \bar{\mathbf{\Lambda}} \bar{\mathbf{\Lambda}}^{\star} \right)^{-1} \bar{\mathbf{\Lambda}} \mathbf{F} \hat{\mathbf{z}}, \text{ where } \hat{\mathbf{z}} = \delta \mathbf{H}^T \mathbf{S}^T \mathbf{y} + \mathbf{z} \text{ and } \bar{\mathbf{\Lambda}} = \left(\mathbf{\Lambda}_1, \dots \mathbf{\Lambda}_{s_f^2} \right) \in \mathbb{R}^{q \times d}, \text{ with}$$
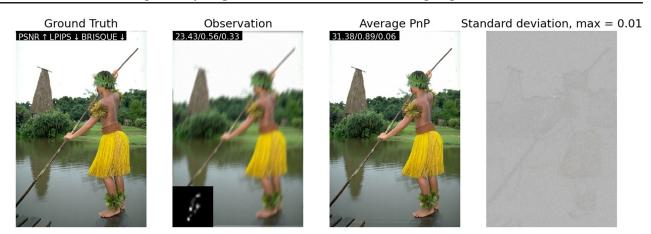
Figure 10. Uncertainty of SNORE to the algorithm randomness. *Rightmost:* The algorithm has been run with 10 different seeds and the standard deviation of restored images is shown. The blur kernel is shown on the observed images and the input noise level is $\sigma_{\mathbf{y}} = 5/255$. Note how the algorithm is stable with a small standard deviation of maximum 0.01 (pixel-values are in $[0, 1]$). Especially, structures are stable with a particularly low standard deviation.
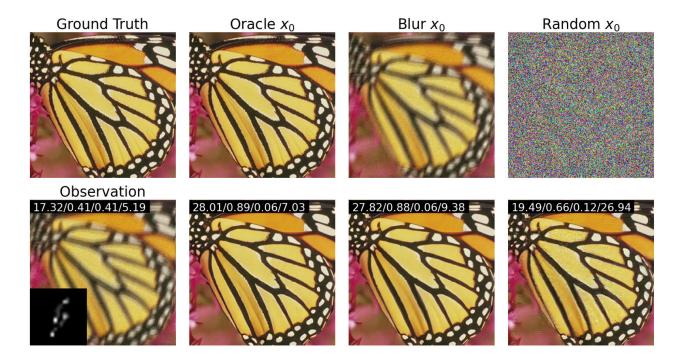

Figure 11. Sensitivity of SNORE to the algorithm initialization. A motion blur kernel and a noise of standard deviation $\sigma_{\mathbf{y}} = 10/255$ are used to degrade the image. *Top Three Leftmost:* Three initializations are used to start the SNORE algorithm: the ground-truth image (Oracle), the observation and a random image where each pixel is sampled uniformly in $[0, 1]$. *Bottom Three Rightmost:* Three corresponding reconstructions. Note how the algorithm succeeds to reconstruct a relevant image even with a random initialization.

$\mathbf{\Lambda} = \text{diag}\left(\mathbf{\Lambda}_1, \dots \mathbf{\Lambda}_{s_f^2}\right)$ a block-diagonal decomposition according to a $s_f \times s_f$ paving of the Fourier domain.

Thanks to the previous expression of the Proximal operator, SNORE Prox and RED Prox can be computed for super-resolution. On Figure 15, we give qualitative results for image super-resolutions on a kernel of blur and various images. Note that SNORE Prox produce better perceptual results (LPIPS, BRISQUE) by creating local texture. However, SNORE Prox is worst in distortion metrics (PSNR, SSIM) as for image deblurring. These are preliminaries experiments and we leave for futur work to adapt each methods for image super-resolution and make a quantitative evaluation.

Figure 12. Restorations obtained with SNORE, RED and DiffPIR algorithms for various images from the dataset CBSD68 on the inpainting task with a random mask with a proportion 0.5 of masked pixels.
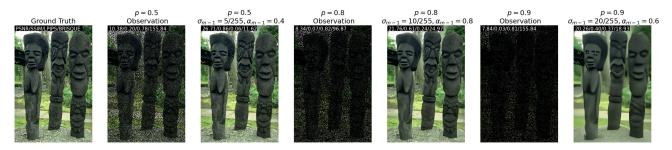
Figure 13. Restorations obtained with SNORE for inpainting, with a random mask of proportion p of masked pixels, on one image from the dataset CBSD68. The last annealing parameters $(\alpha_{m-1}, \sigma_{m-1})$ are given for each restored image.

Parameters	RED Prox	SNORE Prox
α	0.065	
α_0		0.02
α_{m-1}		0.3
$\sigma/\sigma_{\mathbf{y}}$	2	
$\sigma_0/\sigma_{\mathbf{y}}$		4
$\sigma_{m-1}/\sigma_{\mathbf{y}}$		2
$\max_{\text{itr}}$	400	400
δ (initialization)	$1/\alpha$	1

Table 6. Parameters setting for image super-resolution for the different implemented methods.

Parameters setting On Table 6, we details the practical choice of parameters we made. Note that the number of iterations of the algorithm are the same for image super-resolution.

F.5. SAR despeckling

For Synthetic Aperture Radar (SAR) despeckling, we consider the Goodman's model (Goodman, 1976) in which we aim at recovering the underlying reflexivity $\mathbf{R} \in \mathbb{R}_+^d$ from the observed intensity $\mathbf{I} \in \mathbb{R}_+^d$, that is a noisy version of $\mathbf{R}$ with a multiplicative gamma noise, $\mathbf{I} = \mathbf{N}\mathbf{R}$, where $\mathbf{N} \sim \Gamma(1, L)$ and $L > 0$ is called the number of looks (Goodman, 1976). By tacking the log of the previous model and denoting $\mathbf{y} = \log \mathbf{I}$ and $\mathbf{x} = \log \mathbf{R}$, we turn the multiplicative noise into an additive noise, $\mathbf{y} = \mathbf{x} + \mathbf{n}$, where $\mathbf{n}$ is following the Fisher-Tippett distribution.

To solve this problem, we can solve the following variation problem (Deledalle et al., 2017)

$$\arg\min_{\mathbf{x} \in \mathbb{R}^d} -\log p(\mathbf{y}|\mathbf{x}) + \alpha \mathcal{R}(\mathbf{x}),$$

where $-\log p(\mathbf{y}|\mathbf{x}) = L \sum_{k=1}^d x_k + e^{y_k - x_k} + \text{Cst.}$ (x_k is the value of the k-pixel of $\mathbf{x}$) and $\mathcal{R}$ the regularization.

Working in the log-domain allow us to solve an unconstrained optimization with a convex data-fidelity term. However the gradient of this data-fidelity term is not Lipschitz. By solving this problem with the SNORE regularization, our theoretical analysis still holds (the data-fidelity is C^∞). However, the boundedness hypothesis might be harder to verify because the objective function is not necessarily coercive (see Appendix G for more details).

To solve experimentally this problem, we have trained a GS-denoiser (Hurault et al., 2022a) on the SAR-speckle free dataset developed by Dalsasso et al. (2020). This denoiser is trained to remove additive gaussian noise to SAR images for $\sigma \in [0, 50]$ with the same traning parameters than proposed by Hurault et al. (2022a).

We use RED and SNORE algorithm to despeckle images with the parameters setting details in Table 7. On Figure 14, we show SNORE and RED algorithm qualitative performance on various SAR images. We only compute the PSNR and SSIM metrics because the LPIPS and the BRISQUE are designed for color images. We can notice that SNORE and RED succeed to restore good quality images for the non-standard data-fidelity term of image despeckling.
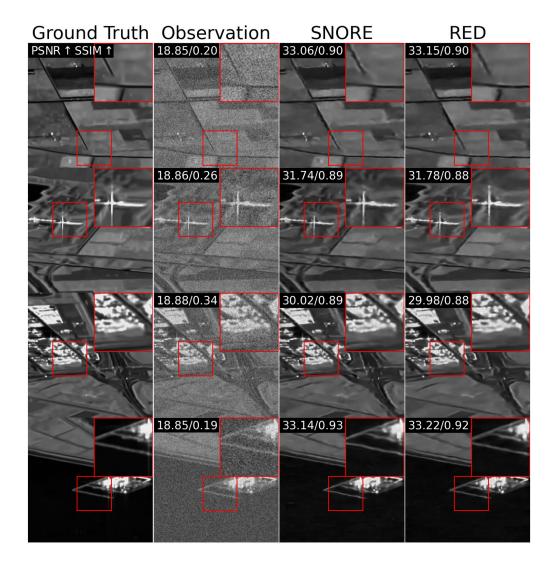
Figure 14. Image despeckling ($L = 1$) on various SAR images with RED and SNORE algorithm with a GS-denoiser trained on SAR images.

G. SNORE applies the denoiser on its training domain

One motivation to use SNORE is to force the denoiser to be applied on its training domain. To do so, at each iteration, the denoiser D_σ is not applied to the previous iteration $\mathbf{x}_k$ (as in RED) but to a noisy version of $\mathbf{x}_k$, $\tilde{\mathbf{x}}_k = \mathbf{x}_k + \sigma\epsilon$, where $\epsilon \sim \mathcal{N}(0, \mathrm{Id})$. In this noisy version $\tilde{\mathbf{x}}_k$, the input noise level is exactly the noise level of the denoiser.

However, in pratice, there might be residual noise in the iteration $\mathbf{x}_k$, so the noise level of $\tilde{\mathbf{x}}_k$ might be higher than σ. In order to verify this experimentally, we use a robust wavelet-based noise estimator (Donoho & Johnstone, 1994) implemented in the library library scikit-image (aka skimage) as the function estimate_sigma(). We define $\hat{\sigma}(\mathbf{x})$ the noise estimation of the image $\mathbf{x}$.
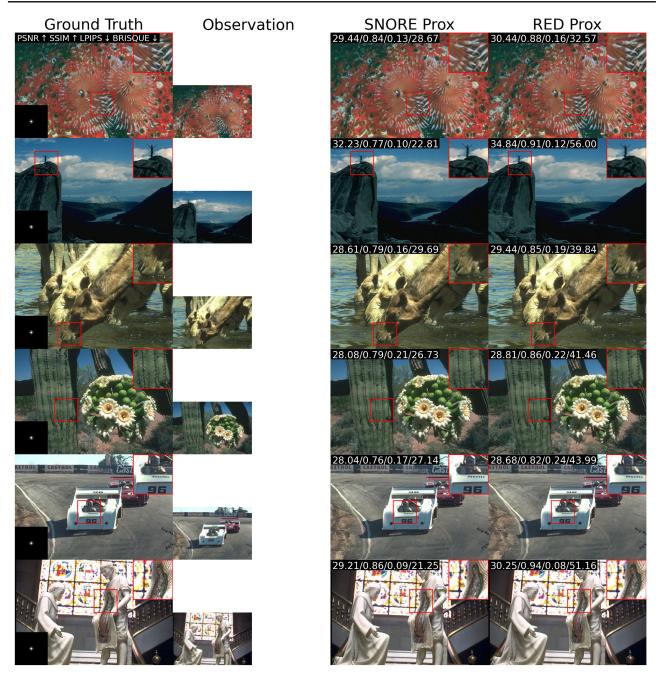
Figure 15. Image super-resolution obtained with SNORE Prox and RED Prox, with a super-resolution factor $s_f = 2$ and an input noise $\sigma_{\mathbf{y}} = 5/255$, on various images from the dataset CBSD68.

On Figure 16, we see that the annealing level are not visible for SNORE, which suggests that the algorithm is well adapt to the noise level. Moreover, the residual noise is decreasing and converging to the natural noise shift for SNORE. After a first phase of removing the noise of the initialization, SNORE algorithm effectively applies the denoiser to an image with the right noise level. For RED, the shift between the input noise level and the noise level of the denoiser is still large at convergence. This suggests that the denoiser is applied on an image out-of-distribution (without the right level of noise) at convergence.

Parameters	RED	SNORE
α	80	
$\alpha_0 = \alpha_{m-1}$		80
$\sigma \times 255$	10	
$\sigma_0 \times 255$		30
$\sigma_{m-1} \times 255$		10
$\max_{\text{itr}}$	100	100
δ	0.01	0.01

Table 7. Parameters setting for image despeckling for the different implemented methods.
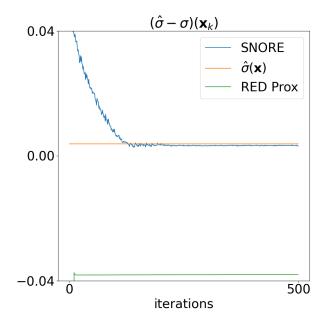

Figure 16. Difference between the estimated noise $\hat{\sigma}$ and the denoiser parameter σ for iterations of SNORE and RED algorithms for inpainting (with a proportion $p = 0.5$ of masked pixels) on one image of the dataset CBSD68. In orange, the estimated noise on the clean image.

H. Discussion on the boundedness of $(\mathbf{x}_k)_{k \in \mathbb{N}}$

In Proposition 3.3-3.5, the convergence of Algorithm 2 is studied almost surely on $\Lambda_{\mathbf{K}}$, the set of realizations where $(\mathbf{x}_k)_{k \in \mathbb{N}}$ is bounded in a compact $\mathbf{K}$. In what follows, we name that the *boundedness assumption*. That kind of assumption is standard in stochastic gradient descent analysis with non-convex objective functions (Benaïm, 1999; Doucet & Tadic, 2017). However, one can remark that in similar non-stochastic Plug-and-Play methods (Hurault et al., 2022a, Appendix D) or in posterior sampling algorithms (Laumont et al., 2022; Renaud et al., 2024), a projection or a penalty term can be added to guarantee a bounded sequence. Unfortunately, to our knowledge, a simple projected stochastic gradient descent step is not simple to analyze.

Davis & Drusvyatskiy (2018) prove convergence of a projected stochastic gradient descent algorithm, but the convergence analysis relies on a random choice of the ending step. Ghadimi & Lan (2013) develop a similar approach. The random choice of the ending step is not satisfying in our setting as we want to fix the number of iterations for a fair comparison with deterministic methods.

Another way to ensure convergence is to use the random projected stochastic gradient descent algorithm proposed by (Nur-

---

Algorithm 8 Randomly Projected SNORE

1: **input:** $\mathbf{x}_0 \in \mathbb{R}^d$, $m \in \mathbb{N}$, $\delta > 0$, $\sigma > 0$, $\alpha > 0$, $N \in \mathbb{N}$, $\beta_0 > 0$, $\lambda_0 = 0$.
2: **for** $k = 0, 1, \ldots, N - 1$ **do**
3: $\quad \delta_k \leftarrow \frac{\delta}{k+1}$
4: $\quad \boldsymbol{\epsilon} \leftarrow \mathcal{N}(0, \mathbf{I}_d)$
5: $\quad \tilde{\mathbf{x}}_k \leftarrow \mathbf{x}_k + \sigma \boldsymbol{\epsilon}$
6: $\quad \mathbf{z}_{k+1} \leftarrow \mathbf{x}_k - \delta_k \nabla \mathcal{F}(\mathbf{x}_k, \mathbf{y}) - \frac{\alpha \delta_k}{\sigma^2} \left(\mathbf{x}_k - D_\sigma(\tilde{\mathbf{x}}_k) \right)$
7: $\quad \mathbf{x}_{k+1} \leftarrow \mathbf{z}_{k+1} \mathbf{1}_{\|\mathbf{z}_{k+1}\| \leq \beta_{\lambda_k}} + \mathbf{x}_0 \mathbf{1}_{\|\mathbf{z}_{k+1}\| > \beta_{\lambda_k}}$
8: $\quad \lambda_{k+1} \leftarrow \lambda_k + \mathbf{1}_{\|\mathbf{z}_{k+1}\| > \beta_{\lambda_k}}$
9: **end for**

---

minski, 1973). As detailed in Algorithm 8, at each iteration, this algorithm realizes a projection onto a ball parameterized by an increasing sequence of positive real numbers $(\beta_n)_{n \in \mathbf{N}}$. This algorithm is proved to converge without the boundedness assumption. Doucet & Tadic (2017, Theorem A1.1.) explored this perspective and demonstrated that the iterates of this algorithm are bounded. However, in our context, (Doucet & Tadic, 2017, Assumption 1.2.) is difficult to verify. In fact, our demonstrations (Proof of Proposition 3.3 in Appendix D.1.3) rely on an upper bound $\mathbb{E}\left(\|\xi_k\|^2 \right)$ which is obtained thanks to the boundedness assumption. In our context, Algorithm 8 has not been proved to converge.

Finally, we leave for future work the exploration of a strategy to demonstrate the convergence of Algorithm 2 without any boundedness assumption.