Denoising Diffusion Models for Plug-and-Play Image Restoration

Yuanzhi Zhu[1] Kai Zhang[1,*] Jingyun Liang[1] Jiezhang Cao[1]
Bihan Wen[2] Radu Timofte[1,3] Luc Van Gool[1,4]
[1]ETH Zurich [2]Nanyang Technological University [3]University of Wurzburg [4]KU Leuven
yuazhu@student.ethz.ch kai.zhang@vision.ee.ethz.ch

Abstract

Plug-and-play Image Restoration (IR) has been widely recognized as a flexible and interpretable method for solving various inverse problems by utilizing any off-the-shelf denoiser as the implicit image prior. However, most existing methods focus on discriminative Gaussian denoisers. Although diffusion models have shown impressive performance for high-quality image synthesis, their potential to serve as a generative denoiser prior to the plug-and-play IR methods remains to be further explored. While several other attempts have been made to adopt diffusion models for image restoration, they either fail to achieve satisfactory results or typically require an unacceptable number of Neural Function Evaluations (NFEs) during inference. This paper proposes DiffPIR, which integrates the traditional plug-and-play method into the diffusion sampling framework. Compared to plug-and-play IR methods that rely on discriminative Gaussian denoisers, DiffPIR is expected to inherit the generative ability of diffusion models. Experimental results on three representative IR tasks, including super-resolution, image deblurring, and inpainting, demonstrate that DiffPIR achieves state-of-the-art performance on both the FFHQ and ImageNet datasets in terms of reconstruction faithfulness and perceptual quality with no more than 100 NFEs. The source code is available at https://github.com/yuanzhi-zhu/DiffPIR

1. Introduction

Recent studies have demonstrated that plug-and-play Image Restoration (IR) methods can effectively handle a variety of low-level vision tasks, such as image denoising [5,58], image Super-Resolution (SR) [16,17,37], image deblurring [12] and image inpainting [27], with excellent results [6, 11, 54, 57, 59]. With the help of variable splitting algorithms, such as the Alternating Direction Method of Multipliers (ADMM) [4] and Half-Quadratic-Splitting (HQS) [22], plug-and-play IR methods integrate Gaussian denoisers into the iterative process, leading to improved performance and convergence.

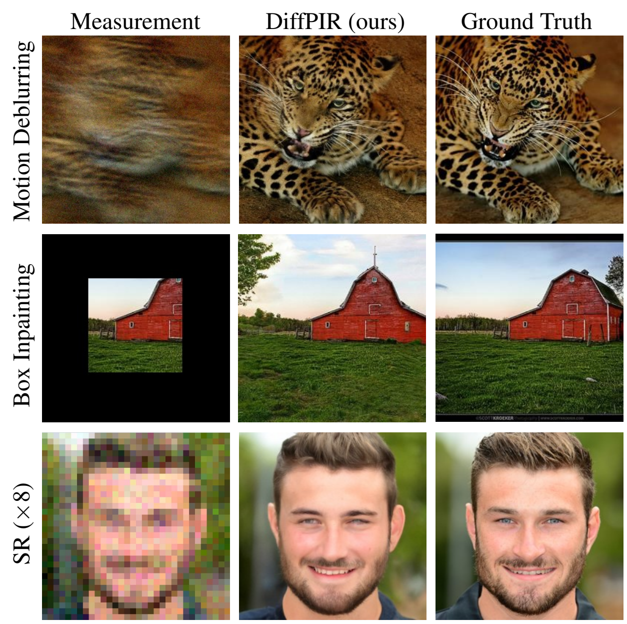

Figure 1. **DiffPIR examples of reconstruction.** We present the reconstructed images and corresponding measurements and ground truth labels for several common image restoration tasks.

The main idea of plug-and-play IR methods is to separate the data term and prior term of the following optimization problem

$$\hat{\mathbf{x}} = \arg\min_{\mathbf{x}} \frac{1}{2\sigma_n^2}\|\mathbf{y} - \mathcal{H}(\mathbf{x})\|^2 + \lambda\mathcal{P}(\mathbf{x}), \quad (1)$$

where $\mathbf{y}$ is the measurement of ground truth $\mathbf{x}_0$ given the degradation model $\mathbf{y} = \mathcal{H}(\mathbf{x}_0) + \mathbf{n}$, $\mathcal{H}$ is a known degradation operator, σ_n denotes the known standard deviation of i.i.d. Gaussian noise $\mathbf{n}$, and $\lambda\mathcal{P}(\cdot)$ is the prior term with regularization parameter λ. To be specific, the data term ensures that the solution adheres to the degradation process, while the prior term enforces the solution to be close

*Corresponding author.

proximity to the desired data distribution. In particular, the prior term can be implicitly addressed by Gaussian denoisers [39, 47, 54, 60]. Venkatakrishnan *et al*. [54] proposed to solve (1) by forming the augmented Lagrangian function and using the ADMM technique with various image-denoising methods. Kamilov *et al*. [29] used the BM3D denoising operator [10] to solve the prior subproblem for non-linear inverse problems. While the above methods used traditional denoisers, Zhang *et al*. [59] made the first attempt to incorporate deep denoiser priors to solve various IR tasks. In subsequent research, Zhang *et al*. [57] further proposed a more powerful denoiser for plug-and-play IR, which has since been adopted in numerous recent studies [1, 3, 21, 34].

Unlike those traditional or convolutional neural network based discriminative Gaussian denoisers, denoisers parameterized by deep generative models are expected to better handle those ill-defined inverse problems due to their ability to model complex distributions. Deep generative models such as Generative Adversarial Networks (GANs) [23, 31], Normalizing Flows (NFs) [15] and Variational Autoencoders (VAEs) [33, 44] have been used as denoisers of plug-and-play IR framework [18, 35, 55]. However, these generative models are not designed for denoising tasks and their generative capabilities are hindered when employed as plug-and-play prior.

Recently, diffusion models have demonstrated the ability to generate images with higher quality [14, 41, 43] than previous generative models such as GANs, VAEs and NFs even when starting from pure Gaussian noise. Diffusion models define a forward diffusion process that maps data to noise by gradually perturbing the input data with Gaussian noise. While in the reverse process, they generate images by gradually removing Gaussian noise, with the intuition from non-equilibrium thermodynamics [50]. The representative works in this area include Denoising Diffusion Probabilistic Models (DDPM) [24] and score-based Stochastic Differential Equation (SDE) [53]. In addition to their unconditional generative power, diffusion models have also achieved remarkable success in the field of general inverse problems. Saharia *et al*. [49] employed a conditional network, using low-resolution images as conditional inputs, for the purpose of single image SR. Lugmayr *et al*. [38] proposed an improved sampling strategy that resamples iterations for image inpainting. Chung *et al*. [8] introduced a Diffusion Posterior Sampling (DPS) method with Laplacian approximation for posterior sampling, which can be applied to noisy non-linear inverse problems. Choi *et al*. [7] proposed to adopt low-frequency information from measurement $\mathbf{y}$ to guide the generation process towards a narrow data manifold. Kawar *et al*. [32] endorsed a time-efficient approach named Denoising Diffusion Restoration Models (DDRM) which performs diffusion sampling to reconstruct the missing information in $\mathbf{y}$ in the spectral space of $\mathcal{H}$ with Singular Value Decomposition (SVD). While the above methods achieve promising results, these methods either are hand-designed (*e.g*., [38]) or suffer from sampling speed to get favorable performance (*e.g*., [8, 32]).

There exists another line of work, known as plug-and-play posterior sampling methods, which leverage the gradient of log posteriors to drive the samples to high-density regions. In this approach, the posteriors are decomposed into explicit likelihood functions and plug-and-play priors. Durmus *et al*. [19] proposed a Moreau-Yosida regularised unadjusted Langevin algorithm for Bayesian computation such as imaging inverse problems. Laumont *et al*. [36] extended this idea with Tweedie's identity [20] and introduced PnP-unadjusted Langevin algorithm for image inverse problems. Both Romano *et al*. [45] and Kadkhodaie *et al*. [28] explicitly established the connection between a prior and a denoiser and used denoisers for stochastic posterior sampling.

Inspired by the ability of plug-and-play IR to utilize any off-the-shelf denoisers as an implicit image prior, and considering that diffusion models are essentially generative denoisers, we propose denoising diffusion models for plug-and-play IR, referred to as DiffPIR. Following the plug-and-play IR method [57], we decouple the data term and the prior term and solve them iteratively within the diffusion sampling framework. The data term can be solved independently, allowing DiffPIR to handle a wide range of degradation models with various degradation operators $\mathcal{H}$. As for the prior term, it can be solved using an off-the-shelf unconditional diffusion model as a plug-and-play denoiser prior [32].

We conduct experiments including SR, image deblurring and image inpainting on FFHQ [31] and ImageNet [46]. By comparing our method with the other competitive approaches, we demonstrate that DiffPIR can efficiently restore images with superior quality (See Figure 1 for qualitative examples).

2. Background

2.1. Score-based Diffusion Models

Diffusion is the process of destructing a signal (image) by adding Gaussian noise until the signal-to-noise ratio is negligible. This forward process can be described by an Itô SDE [53]:

$$\mathrm{d}\mathbf{x} = \mathbf{f}(\mathbf{x}, t)\mathrm{d}t + g(t)\mathrm{d}\mathbf{w}, \tag{2}$$

where $\mathbf{w}$ is the standard Wiener process, $\mathbf{f}(\cdot, t)$ is a vector-valued function called the drift coefficient, and $g(\cdot, t)$ is a scalar function known as the diffusion coefficient.

The diffusion process described in (2) can be reversed in time and has the form of [2]:

$$\mathrm{d}\mathbf{x} = [\mathbf{f}(\mathbf{x}, t) - g^2(t)\nabla_{\mathbf{x}} \log p_t(\mathbf{x})]\mathrm{d}t + g(t)\mathrm{d}\mathbf{w}, \tag{3}$$

where $p_t(\mathbf{x})$ is the marginal probability density at timestep t, and the only unknown part $\nabla_{\mathbf{x}} \log p_t(\mathbf{x})$ can be modelled as so-called score function $\mathbf{s}_\theta(\mathbf{x}, t)$ with score matching methods [26, 52]. $\mathbf{x}$ at t is denoted as $\mathbf{x}_t$. We can generate data samples according to (3) by evaluating the score function $\mathbf{s}_\theta(\mathbf{x}_t, t)$ at each intermediate timestep during sampling, even if the initial state is Gaussian noise. The training objective of time-dependent score function $\mathbf{s}_\theta(\mathbf{x}_t, t)$ with denoising score matching can be formulated as:

$$\mathbb{E}_t\Big\{\gamma(t)\mathbb{E}_{\mathbf{x}_0}\mathbb{E}_{\mathbf{x}_t|\mathbf{x}_0}\left[\|\mathbf{s}_\theta(\mathbf{x}_t, t) - \nabla_{\mathbf{x}_t} \log p_{0t}(\mathbf{x}_t|\mathbf{x}_0)\|_2^2\right]\Big\}, \tag{4}$$

where $\gamma(t)$ is a positive weight coefficient, t is uniformly sampled over $[0, T]$, $(\mathbf{x}_0, \mathbf{x}_t) \sim p_0(\mathbf{x})p_{0t}(\mathbf{x}_t|\mathbf{x}_0)$. We can observe from (4) that a well-trained denoising score function $\mathbf{s}_\theta(\mathbf{x}_t, t)$ is also an ideal Gaussian denoiser when the transition probability $p_{0t}(\mathbf{x}_t|\mathbf{x}_0)$ is Gaussian.

2.2. Denoising Diffusion Probabilistic Models

For the specific choice of $\mathbf{f}(\mathbf{x}, t) = -\frac{1}{2}\beta(t)\mathbf{x}$ and $g(\mathbf{x}, t) = \sqrt{\beta(t)}$, we have the forward and reverse SDEs as the continuous version of the diffusion process in DDPM [24]. One forward step of (discrete) DDPM is

$$\mathbf{x}_t = \sqrt{1-\beta_t}\mathbf{x}_{t-1} + \sqrt{\beta_t}\epsilon_{t-1}, \tag{5}$$

with $\epsilon_{t-1} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$. The sample $\mathbf{x}_t$ is obtained by adding i.i.d. Gaussian noise with variance β_t and scaling $\mathbf{x}_{t-1}$ with $\sqrt{1-\beta_t}$. In this way, the total variance is preserved and DDPM is also called "Variance Preserving" (VP) SDE [53]. We can also sample $\mathbf{x}_t$ at an arbitrary timestep t from $\mathbf{x}_0$ in closed form thanks to the good properties of Gaussian:

$$\mathbf{x}_t = \sqrt{\bar{\alpha}_t}\mathbf{x}_0 + \sqrt{1-\bar{\alpha}_t}\epsilon, \tag{6}$$

with new variance $1-\bar{\alpha}_t$ and scaling factor $\sqrt{\bar{\alpha}_t}$. In this work, $\{\beta_t\}$ is the noise schedule and we use the same notation $\alpha_t = 1-\beta_t$ and $\bar{\alpha}_t = \prod_{s=1}^t \alpha_s$ following Ho *et al.* [24]. One reverse step of DDPM is

$$\mathbf{x}_{t-1} = \frac{1}{\sqrt{\alpha_t}}\Big(\mathbf{x}_t - \frac{\beta_t}{\sqrt{1-\bar{\alpha}_t}}\epsilon_\theta(\mathbf{x}_t, t)\Big) + \sqrt{\beta_t}\epsilon_t, \tag{7}$$

where $\epsilon_\theta(\mathbf{x}, t)$ is the function approximator intended to predict the total noise ϵ between $\mathbf{x}_t$ and $\mathbf{x}_0$ in (6).

In DDPM, the goal is to learn the noise added to $\mathbf{x}_0$; in score-based SDE, the goal is to learn the score function, the gradient of log-density of perturbed data; both with a U-Net. The connection between score function and noise prediction in DDPM can be formulated approximately as: $\mathbf{s}_\theta(\mathbf{x}_t, t) \approx -\frac{\epsilon_\theta(\mathbf{x}_t, t)}{\sqrt{1-\bar{\alpha}_t}}$. From now on, we use both $\epsilon_\theta(\mathbf{x}, t)$ and $\mathbf{s}_\theta(\mathbf{x}, t)$ to represent diffusion models.

Song *et al.* proposed Denoising Diffusion Implicit Models (DDIM) [51], where the diffusion process can be extended from Markovian to non-Markovian and (7) can be rewritten as:

$$\begin{aligned}\mathbf{x}_{t-1} =& \sqrt{\bar{\alpha}_{t-1}}\left(\frac{\mathbf{x}_t - \sqrt{1-\bar{\alpha}_t}\epsilon_\theta(\mathbf{x}_t, t)}{\sqrt{\bar{\alpha}_t}}\right) \\ &+ \sqrt{1-\bar{\alpha}_{t-1}-{\sigma_{\eta_t}}^2}\cdot\epsilon_\theta(\mathbf{x}_t, t) + \sigma_{\eta_t}\epsilon_t,\end{aligned} \tag{8}$$

where ϵ_t is standard Gaussian noise, inside the parentheses of the first term is the predicted $\mathbf{x}_0$ at timestep t, and the magnitude of σ_{η_t} controls how stochastic the forward process is. One advantage of this non-Markovian diffusion process is that we can sample with diffusion models more efficiently [51].

2.3. Conditional Diffusion Models

For conditional generation tasks given the condition $\mathbf{y}$, we aim to sample from the posterior $p(\mathbf{x}|\mathbf{y})$ rather than the prior. In the work of Song *et al.* [53], (3) can be rewritten as follows for conditional generation with the help of Bayes' theorem

$$\mathrm{d}\mathbf{x} = [\mathbf{f}(\mathbf{x}, t) - g^2(t)\nabla_{\mathbf{x}}(\log p_t(\mathbf{x}) + \log p_t(\mathbf{y}|\mathbf{x}))]\mathrm{d}t + g(t)\mathrm{d}\mathbf{w}, \tag{9}$$

where the posterior is divided into $p_t(\mathbf{x})$ and $p_t(\mathbf{y}|\mathbf{x})$. In this way, the unconditional pre-trained diffusion models can be used for conditional generation with an additional classifier.

Ho *et al.* [25] introduced the classifier-free diffusion guidance with $\mathbf{s}_\theta(\mathbf{x}, t, \mathbf{y}) = \nabla_{\mathbf{x}} \log p_t(\mathbf{x}|\mathbf{y})$ the image-conditional diffusion models. With the same idea, Saharia *et al.* [48, 49] trained image-conditional diffusion models for SR and image-to-image translation in concurrent work. Nichol *et al.* [40] propose to use text-guided diffusion models to generate photo-realistic images with classifier-free guidance. The hyperparameter λ in (1) can be interpreted as the guidance scale in classifier-free diffusion models.

While the above methods require additional neural network modules, conditional generation can be done with merely unconditional pre-trained diffusion models [7, 8, 32, 38]. Given (9), we can first update with one unconditional reverse diffusion step and then handle the conditional input.

3. Proposed Method

We adapt the HQS algorithm to decouple the data term and prior term of (1), because we can solve the decoupled subproblems iteratively and hence leverage the diffusion sampling framework [57]. By introducing an auxiliary variable $\mathbf{z}$, (1) can be split into the following subproblems and be solved iteratively,

$$\begin{cases}\mathbf{z}_k = \arg\min_{\mathbf{z}} \dfrac{1}{2(\sqrt{\lambda/\mu})^2}\|\mathbf{z}-\mathbf{x}_k\|^2 + \mathcal{P}(\mathbf{z}) & \text{(10a)}\\ \mathbf{x}_{k-1} = \arg\min_{\mathbf{x}} \|\mathbf{y}-\mathcal{H}(\mathbf{x})\|^2 + \mu\sigma_n^2\|\mathbf{x}-\mathbf{z}_k\|^2, & \text{(10b)}\end{cases}$$

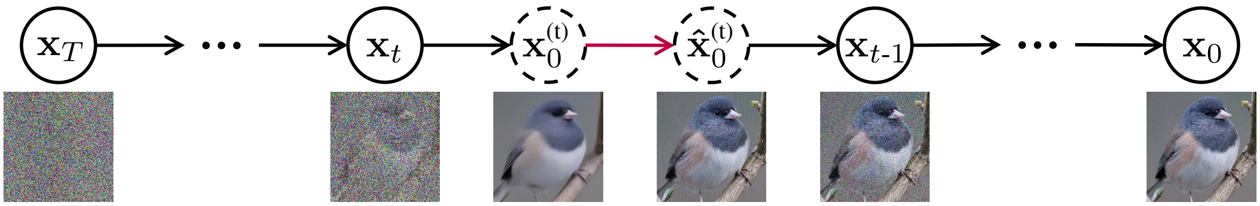

Figure 2. **Illustration of our sampling method.** For every state $\mathbf{x}_t$, following the prediction of the estimated $\mathbf{x}_0^{(t)}$ by the diffusion model, the measurement $\mathbf{y}$ is incorporated by solving the data proximal subproblem (indicated by the red arrow). Subsequently, the next state $\mathbf{x}_{t-1}$ is derived by adding noise back and thus completing one step of reverse diffusion sampling.

where the parameter μ is introduced as the coefficient for the data-consistent constraint term. Here the subproblem (10a) with prior term is a Gaussian denoising problem , and the subproblem (10b) with the data term is indeed a proximal operator [42] which usually has a closed-form solution that depends on $\mathcal{H}$.

Our goal is to solve inverse problems via posterior sampling with generative diffusion models. Just like most plug-and-play methods, we can decouple the data term and prior term [57]. The prior term ensures the generated sample is from the prior data distribution, and the data term narrows down the image manifold with the given measurement $\mathbf{y}$ [7]. We introduce them first in Section 3.1 and Section 3.2, then our proposed sampling method in Section 3.3. In Section 3.4, we highlight the differences between DiffPIR and several closely related diffusion-based methods. In Section 3.5, we show that our sampling can be accelerated like DDIM.

3.1. Diffusion Models as Generative Denoiser Prior

One important property of diffusion models is that the models can be understood as a combination of a generator (for the first few steps) and denoiser (for the rest of the steps) [13]. Intuitively, we can simply apply diffusion models as deep prior denoiser in HQS algorithm with a suitable initialization for plug-and-play IR [57]. However, one significant difference between diffusion models and other deep denoisers is the generative power of diffusion models. With this generative ability, we will show that our method is capable of solving especially highly challenging inverse problems, e.g. image inpainting with large masks.

It would be beneficial to build the connection between (10a) and the diffusion process. Assume we want to solve noiseless $\mathbf{z}_k$ from $\mathbf{x}_t$ with noise level $\bar{\sigma}_t = \sqrt{\frac{1-\bar{\alpha}_t}{\bar{\alpha}_t}}$, we can let $\sqrt{\lambda/\mu} = \bar{\sigma}_t$ for convenience. Given the noise schedule $\{\beta_t\}$ and hyperparameter λ, $\bar{\sigma}_t$ is known. Indeed, (10a) can be solved as a proximal operator. Note that we have $\nabla_{\mathbf{x}}\mathcal{P}(\mathbf{x}) = -\nabla_{\mathbf{x}} \log p(\mathbf{x}) = -\mathbf{s}_\theta(\mathbf{x})$, we can write immediately:

$$\mathbf{z}_k \approx \mathbf{x}_k + \frac{1-\bar{\alpha}_t}{\bar{\alpha}_t}\mathbf{s}_\theta(\mathbf{x}_k), \tag{11}$$

which means $\mathbf{z}_k$ is the estimated clear image $\mathbf{x}_0^{(t)}$ in "Variance Exploding" (VE) SDE form. Since the VP and VE Diffusion Models are actually equivalent to each other [32], from now on we limit our discussion within DDPM without loss of generality, which is a special case of VP SDE diffusion models. To make the discussion more clear, we rewrite (10) as

$$\begin{cases} \mathbf{x}_0^{(t)} = \arg\min_{\mathbf{z}} \frac{1}{2\bar{\sigma}_t^2}\|\mathbf{z}-\mathbf{x}_t\|^2 + \mathcal{P}(\mathbf{z}) & \text{(12a)} \\ \hat{\mathbf{x}}_0^{(t)} = \arg\min_{\mathbf{x}} \|\mathbf{y}-\mathcal{H}(\mathbf{x})\|^2 + \rho_t\|\mathbf{x}-\mathbf{x}_0^{(t)}\|^2 & \text{(12b)} \\ \mathbf{x}_{t-1} \longleftarrow \hat{\mathbf{x}}_0^{(t)}. & \text{(12c)} \end{cases}$$

where $\rho_t = \lambda(\sigma_n/\bar{\sigma}_t)^2$. Here (12b) is the data subproblem to solve and (12c) is a necessary step to finish our sampling method which we will introduce in 3.3. Additionally, we show in Appendix A.1 that we can also derive one-step reverse diffusion from HQS.

Algorithm 1 DiffPIR

Require: $\mathbf{s}_\theta, T, \mathbf{y}, \sigma_n, \{\bar{\sigma}_t\}_{t=1}^T, \zeta, \lambda$

Initialize $\mathbf{x}_T \sim \mathcal{N}(\mathbf{0},\mathbf{I})$, pre-calculate $\rho_t \triangleq \lambda\sigma_n^2/\bar{\sigma}_t^2$.
for $t = T$ **to** 1 **do**
  $\mathbf{x}_0^{(t)} = \frac{1}{\sqrt{\bar{\alpha}_t}}(\mathbf{x}_t + (1-\bar{\alpha}_t)\mathbf{s}_\theta(\mathbf{x}_t, t))$ // *Predict $\hat{\mathbf{z}}_0$ with score model as denoisor*
  $\hat{\mathbf{x}}_0^{(t)} = \arg\min_{\mathbf{x}} \|\mathbf{y}-\mathcal{H}(\mathbf{x})\|^2 + \rho_t\|\mathbf{x}-\mathbf{x}_0^{(t)}\|^2$ // *Solving data proximal subproblem*
  $\hat{\epsilon} = \frac{1}{\sqrt{1-\bar{\alpha}_t}}(\mathbf{x}_t - \sqrt{\bar{\alpha}_t}\hat{\mathbf{x}}_0^{(t)})$ // *Calculate effective $\hat{\epsilon}(\mathbf{x}_t, \mathbf{y})$*
  $\epsilon_t \sim \mathcal{N}(\mathbf{0},\mathbf{I})$
  $\mathbf{x}_{t-1} = \sqrt{\bar{\alpha}_{t-1}}\hat{\mathbf{x}}_0^{(t)} + \sqrt{1-\bar{\alpha}_{t-1}}(\sqrt{1-\zeta}\hat{\epsilon} + \sqrt{\zeta}\epsilon_t)$ // *Finish one step reverse diffusion sampling*
end for
return $\mathbf{x}_0$

3.2. Analytic Solution to Data Subproblem

For IR tasks like image deblurring, image inpainting and SR, we have a fast solution of (12b) based on the estimated $\mathbf{z}_0$ on the image manifold [57]. Since the data term and prior term are decoupled, the degradation model where we observe the measurement $\mathbf{y}$ is only related to (12b).

FFHQ		Deblur (Gaussian)			Deblur (motion)			SR (×4)		
Method	NFEs ↓	PSNR ↑	FID ↓	LPIPS ↓	PSNR ↑	FID ↓	LPIPS ↓	PSNR ↑	FID ↓	LPIPS ↓
DiffPIR	100	27.36	**59.65**	**0.236**	**26.57**	**65.78**	**0.255**	26.64	**65.77**	0.260
DPS [8]	1000	25.46	65.57	0.247	23.31	73.31	0.289	25.77	67.01	**0.256**
DDRM [32]	20	25.93	101.89	0.298	-	-	-	27.92	89.43	0.265
DPIR [57]	>20	**27.79**	123.99	0.450	26.41	146.44	0.467	**28.03**	133.39	0.456

ImageNet		Deblur (Gaussian)			Deblur (motion)			SR (×4)		
Method	NFEs ↓	PSNR ↑	FID ↓	LPIPS ↓	PSNR ↑	FID ↓	LPIPS ↓	PSNR ↑	FID ↓	LPIPS ↓
DiffPIR	100	22.80	**93.36**	**0.355**	**24.01**	**124.63**	**0.366**	23.18	**106.32**	0.371
DPS [8]	1000	19.58	138.80	0.434	17.75	184.45	0.491	22.16	114.93	0.383
DDRM [32]	20	22.33	160.73	0.427	-	-	-	23.89	118.55	**0.358**
DPIR [57]	>20	**23.86**	189.92	0.476	23.60	210.31	0.489	**23.99**	204.83	0.475

Table 1. **Noisy quantitative results on FFHQ (top) and ImageNet (bottom).** We compute the average PSNR (dB), FID and LPIPS of different methods on Gaussian deblurring, motion deblurring and 4× SR.

When there is no analytical solution to (12b), we can still solve it approximately as a first-order proximal operator. The approximate solution is:

$$\hat{\mathbf{x}}_0^{(t)} \approx \mathbf{x}_0^{(t)} - \frac{\bar{\sigma}_t^2}{2\lambda\sigma_n^2}\nabla_{\mathbf{x}_0^{(t)}}\|\mathbf{y} - \mathcal{H}(\mathbf{x}_0^{(t)})\|^2. \tag{13}$$

This is also one gradient descent step and we can calculate it numerically.

3.3. DiffPIR Sampling

With the discussion in the above two subsections, we can get an estimation of $\hat{\mathbf{x}}_0^{(t)}(\mathbf{y})$ given its noised version $\mathbf{x}_t$ after calculating the analytical solution. However, this estimation is not accurate, and we can add noise back to noise level $t-1$ as in (12c). This estimation-correction idea was proposed in both [51] and [30]. In the DDIM fashion, we can first consider the estimation $\hat{\mathbf{x}}_0^{(t)}(\mathbf{y})$ as a sample from the conditional distribution $p(\mathbf{x}|\mathbf{y})$. Then we can calculate the effective predicted noise $\hat{\epsilon}(\mathbf{x}_t, \mathbf{y}) = \frac{1}{\sqrt{1-\bar{\alpha}_t}}(\mathbf{x}_t - \sqrt{\bar{\alpha}_t}\hat{\mathbf{x}}_0^{(t)})$ to get the final one-step sampling expression similar to (8)

$$\mathbf{x}_{t-1} = \sqrt{\bar{\alpha}_{t-1}}\hat{\mathbf{x}}_0^{(t)}(\mathbf{y}) + \sqrt{1-\bar{\alpha}_{t-1}-\sigma_{\eta_t}^2}\hat{\epsilon}(\mathbf{x}_t, \mathbf{y}) + \sigma_{\eta_t}\epsilon_t, \tag{14}$$

where $\hat{\epsilon}$ is the corrected version of predicted noise and $\epsilon_t \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$. In our case, we found that the noise term $\sigma_{\eta_t}\epsilon_t$ may not be strong enough and we can set $\sigma_{\eta_t} = 0$ in our case. Instead, we use a hyperparameter ζ to introduce noise to balance ϵ_t and $\hat{\epsilon}$ and the explicit form of (12c) becomes

$$\mathbf{x}_{t-1} = \sqrt{\bar{\alpha}_{t-1}}\hat{\mathbf{x}}_0^{(t)} + \sqrt{1-\bar{\alpha}_{t-1}}(\sqrt{1-\zeta}\hat{\epsilon} + \sqrt{\zeta}\epsilon_t), \tag{15}$$

Here the hyperparameter ζ controls the variance of the noise injected in each step and our sampling strategy becomes deterministic when $\zeta = 0$.

Based on the above discussion, we summarized the detailed algorithm of our method, namely DiffPIR, in Algorithm 1. Our sampling method is demonstrated in Figure 2. It is worth to mention that by estimating $\hat{x}_0^{(t)}(x_t, y)$and correcting $\hat{\epsilon}(x_t, y)$ we are implicitly calculating the conditional score $s_\theta(x_t, y)$.

3.4. Comparison to Related Works

In this section, we distinguish the proposed DiffPIR from several closely related diffusion-based methods.

DDRM [51]. In DDRM, Kawar *et al*. introduced variational distribution of variables in the spectral space of general linear operator $\mathcal{H}$. It is worth noting that DDRM is structurally similar to our method, as both first predict $\mathbf{x}_0$ and then add noise to forward sample $\mathbf{x}_{t-1}$. However, DDRM is severely limited because it only applies to linear $\mathcal{H}$ and its efficiency is not guaranteed when fast SVD is not feasible. On the contrary, DiffPIR can handle arbitrary degradation operator $\mathcal{H}$ with (13).

DPS [8]. In DPS, Chung *et al*. use Laplacian approximation to circumvent the intractability of posterior sampling by diffusion models, and their method can solve general noisy inverse problems. However, DPS suffers from its sampling speed and its reconstruction is not faithful with fewer sampling steps. Moreover, while DPS and DiffPIR have a similar solution for general inverse problems, just like the other posterior sampling methods with diffusion models (i.e. [7, 8, 38] and sampling methods in Appendix A.2), it handles the measurement after each reverse diffusion step. As for DiffPIR, it adds measurement *within* reverse diffusion steps based on DDIM, which supports fast sampling.

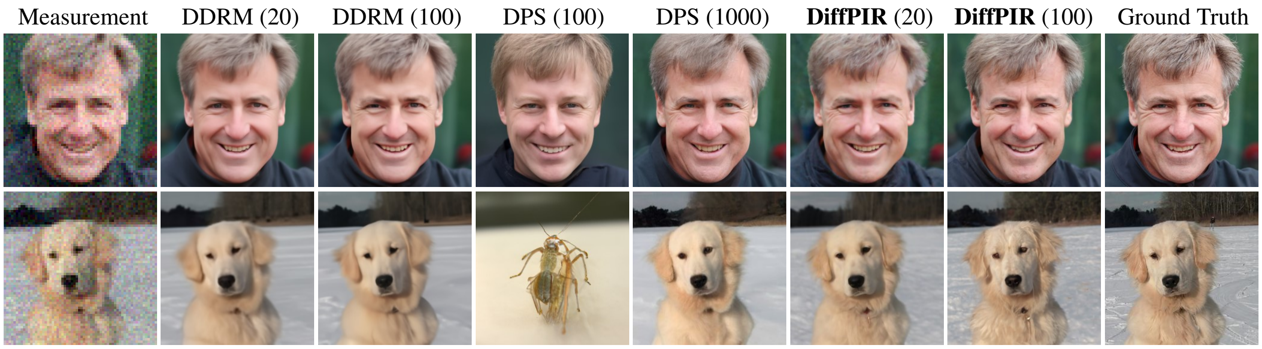

Figure 3. **Qualitative results of 4× SR.** We compare DiffPIR, DPS and DDRM with $\sigma_n = 0.05$

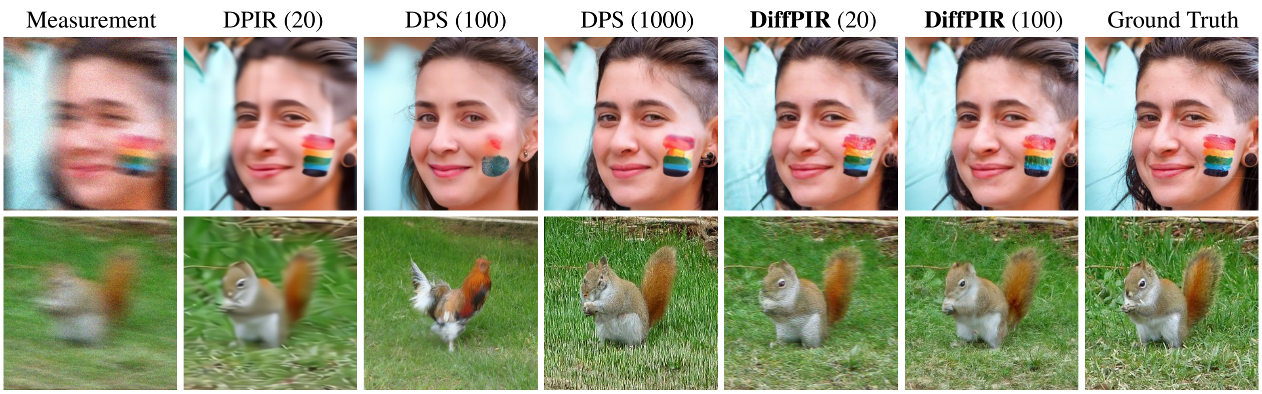

Figure 4. **Qualitative results of motion deblurring.** We compare DiffPIR, DPS and DPIR with $\sigma_n = 0.05$

3.5. Accelerated Generation Process

While the generative ability of diffusion models has been proven to be better than other generative models like GAN and VAE, their slow inference speed ($\sim$ 1000 Neural Function Evaluations (NFEs)) impedes them from being applied in many real-world applications [56]. As suggested in [51], DDPMs can be generalized to DDIMs with non-Markovian diffusion processes but the same training objective, because the denoising objective (4) does not depend on any specific forward procedure as long as $p_{0t}(\mathbf{x}_t \mid \mathbf{x}_0)$ is fixed. As a result, our sampling sequence (length T) can be a subset of $[1, ..., N]$ used in training. To be specific, we adapt the quadratic sequence in DDIM for sampling, which has more sampling steps at low-noise regions [51].

4. Experiments

4.1. Implementation Details

We performed extensive experiments on the FFHQ 256×256 [31] and ImageNet 256×256 [46] datasets, to evaluate different methods. For each dataset, we evaluate 100 hold-out validation images. As our method is training-free, we use pre-trained models from [14] and [7] for samples from ImageNet and FFHQ, respectively. In all experiments, we use the same linear noise schedule $\{\beta_t\}$ with different sampling sequences for each method and all other settings of the diffusion models keep the same. We also report NFEs in each experiment for comparison.

The degradation models are specified as follows: (*i*) For inpainting with box-type mask, the mask is 128×128 box region following [8]; for inpainting with random-type mask, we mask out half of the total pixels at random; and for inpainting with prepared mask images, we load the masks from [38]. (*ii*) Gaussian blur kernel has a size of 61 × 61 with a standard deviation of 3.0, and motion blur is randomly generated with a size of 61 × 61 and intensity value of 0.5 followings [8]. To make a fair comparison, we use the same motion blur kernel for all experiments. (*iii*) For SR, bicubic downsampling is performed. For image inpainting, we only consider the noiseless case. For image deblurring and SR, we do experiments with both noisy and noiseless settings. All images are normalized to the range $[0, 1]$. For more experimental details like the parameter setting, see Appendix B.

4.2. Quantitative Experiments

For quantitative experiments, the metrics we used for comparison are Peak Signal-to-Noise Ratio (PSNR),

FFHQ		Inpaint (box)		Inpaint (random)			Deblur (Gaussian)			Deblur (motion)			SR (×4)		
Method	NFEs ↓	FID ↓	LPIPS ↓	PSNR ↑	FID ↓	LPIPS ↓	PSNR ↑	FID ↓	LPIPS ↓	PSNR ↑	FID ↓	LPIPS ↓	PSNR ↑	FID ↓	LPIPS ↓
DiffPIR	20	35.72	0.117	34.03	30.81	0.116	30.74	46.64	0.170	37.03	20.11	0.084	29.17	58.02	0.187
DiffPIR	100	**25.64**	**0.107**	**36.17**	**13.68**	**0.066**	**31.00**	**39.27**	**0.152**	37.53	**11.54**	**0.064**	29.52	**47.80**	**0.174**
DPS [8]	1000	43.49	0.145	34.65	33.14	0.105	27.31	51.23	0.192	26.73	58.63	0.222	27.64	59.06	0.209
DDRM [32]	20	37.05	0.119	31.83	56.60	0.164	28.40	67.99	0.238	-	-	-	30.09	68.59	0.188
DPIR [57]	>20	-	-	-	-	-	30.52	96.16	0.350	**38.39**	27.55	0.233	**30.41**	96.16	0.362

Table 2. **Noiseless quantitative results on FFHQ.** We compute the average PSNR (dB), FID and LPIPS of different methods on inpainting, deblurring, and SR.

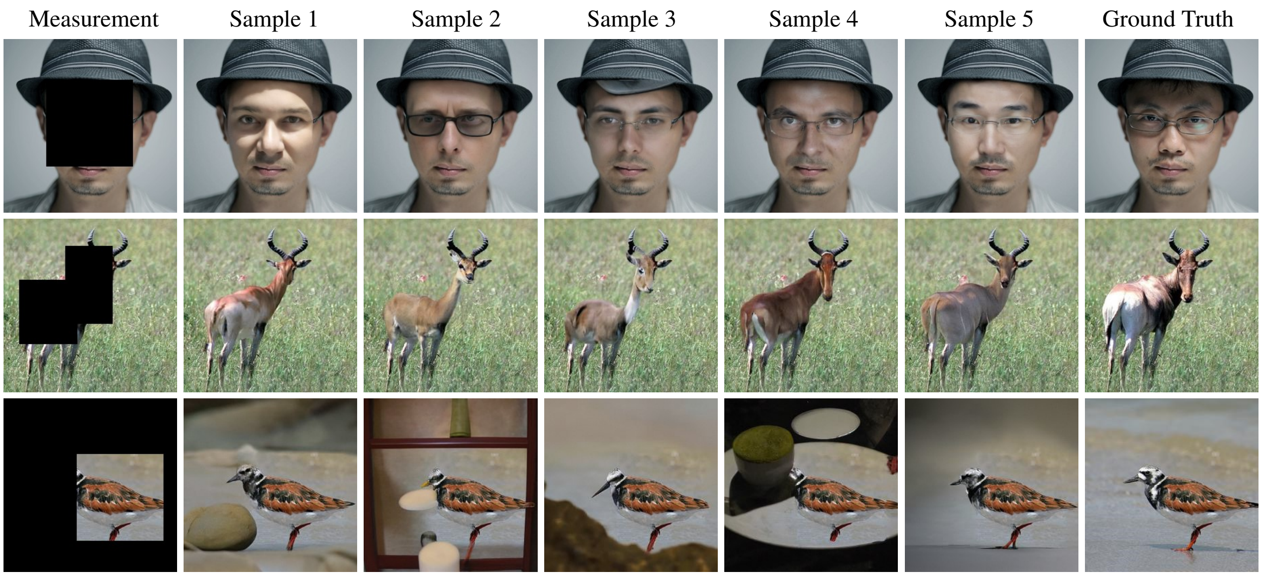

Figure 5. **Qualitative results of inpainting.** We demonstrate the ability of DiffPIR to generate diverse reconstructions for different masks.

Fréchet Inception Distance (FID), and Learned Perceptual Image Patch Similarity (LPIPS) distance. The FID evaluates the visual quality and distance between two image distributions. LPIPS measures the perceptual similarity between two images. PSNR measures the faithfulness of restoration between two images, which is not important but necessary for IR tasks. We report the results for both FFHQ 256×256 and ImageNet 256×256 datasets.

We compare DiffPIR (with 20 and 100 steps) with diffusion-based methods including DDRM [32] and DPS [8], and plug-and-play method DPIR [57]. The sampling steps for DDRM and DPS are 20 and 1000 according to the original paper. Since DPIR has different iteration numbers for different tasks, we indicated the lowest needed number in the results. We used the same pre-trained score-based models and blur kernels for all diffusion-based methods and DPIR to ensure fairness.

For noisy measurement with $\sigma_n = 0.05$, we evaluate all methods on both datasets for 4× SR, Gaussian deblurring, and motion deblurring, except DDRM on motion deblurring as DDRM only supports separable kernel for image deblurring. Table 1 shows that DiffPIR outperforms all the other comparison methods in FID and LPIPS on both datasets and scores competitive PSNR. The only exception is the LPIPS score of SR, and the reason could be that the approximated bicubic kernels $\mathbf{k}$ are not accurate and may cause accumulated error during sampling.

For noiseless measurement with $\sigma_n = 0.0$, we evaluate all methods on FFHQ 256×256 for image inpainting, deblurring, and SR. We skip DPIR for inpainting because DPIR does not have initialization for arbitrary masks. The quantitative results are summarized in Table 2. For noiseless cases, our method with 100 NFEs outperforms the other comparison methods significantly in FID and LPIPS. While DPIR has a larger advantage in PSNR for noiseless tasks, the generated images fail to present high perceptual quality. Given 20 NFEs, DiffPIR already has competitive FID and LPIPS scores, but the visual quality for tasks like inpainting is not as good as methods like DPS.

4.3. Qualitative Experiments

DiffPIR is able to produce high-quality reconstructions, as shown in Figure 1 and Appendix D. In Figure 3, we compare DiffPIR with DPS and DDRM on 4× SR. In Figure 4, we compare DiffPIR with DPS and DPIR on motion deblurring. We found that unlike DDRM (no matter with 20 or

100 NFEs) and DPIR which tend to produce blurry images, DiffPIR is able to reconstruct images with details. Moreover, compared to DPS, DiffPIR needs much fewer NFEs (100 NFEs is not enough for DPS to reconstruct faithfully).

Moreover, our sampling method can generate diverse reconstructions like DDPM because all the reverse diffusion steps are inherently stochastic when hyperparameter $\zeta > 0$. With examples from image inpainting (see Figure 5) we show that DiffPIR can generate high-quality reconstruction with diversity and good semantic alignment even when the degradation is strong (75% masked).

4.4. Ablation Study

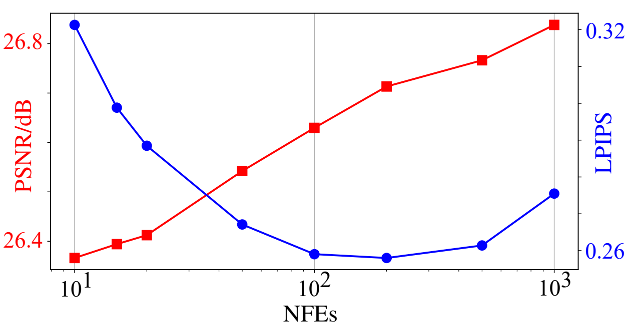

Figure 6. Effect of sampling steps/NFEs

Effect of sampling steps. To investigate the effect of sampling steps or equivalently the number of NFEs, we perform 4× noisy SR (σ_n=0.05) experiment on 100 images from ImageNet validation set for sampling steps $T \in [10, 15, 20, 50, 100, 200, 500, 1000]$. Hyperparameters are fixed as λ=8.0 and ζ=0.3, respectively. We plot the quantitative results in Figure 6. From this plot, we observe that while the PSNR is log-linear to the number of NFEs, the LPIPS score is lowest for $T \in [100, 500]$. As a result, DiffPIR can produce detailed images with less than 100 NFEs and the default number of NFEs is 100 in this paper.

Effect of t_{start}. Similar to [9], our methods can also start the reverse diffusion process from a partial noised image rather than pure Gaussian noise ($t_{start} = 1000$ in our case) to reduce the NFEs for sampling, especially for tasks like deblurring and SR. To analyze if skipping the first few diffusion steps will cause a decrease of IR ability, we show in Figure 7 how PSNR and LPIPS change with the t_{start} for noisy Gaussian deblurring task. For each t_{start}, we also indicate the actual number of NFEs in the Figure. Hyperparameters are fixed as $\lambda = 8.0$ and $\zeta = 0.5$. We find that our method performs well for even $t_{start} = 400$, which leads to a great reduction of NFEs without loss of quality (see Appendix C for further explanation). We further put two images of $t_{start} = 400$ and $t_{start} = 200$ for comparison.

Effects of λ and ζ. DiffPIR has two hyperparameters λ and ζ, which control the strength of the condition guidance and the level of noise injected at each timestep. To illustrate their effects, we show the reconstructed images of a motion-blurred sample in Figure 8. From these results, we observe that (*i*) when the guidance is too strong (*e.g.* $\lambda < 1.0$) the noise is amplified and when the guidance is too weak (*e.g.* $\lambda > 1000$), the generated images are more *unconditional*; (*ii*) the generated images tend to be blurry when ζ approaches 1.

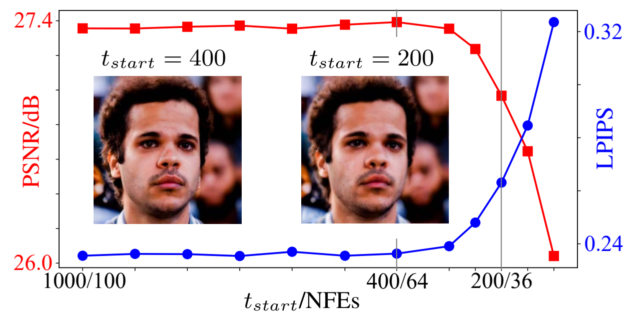

Figure 7. Effect of t_{start}

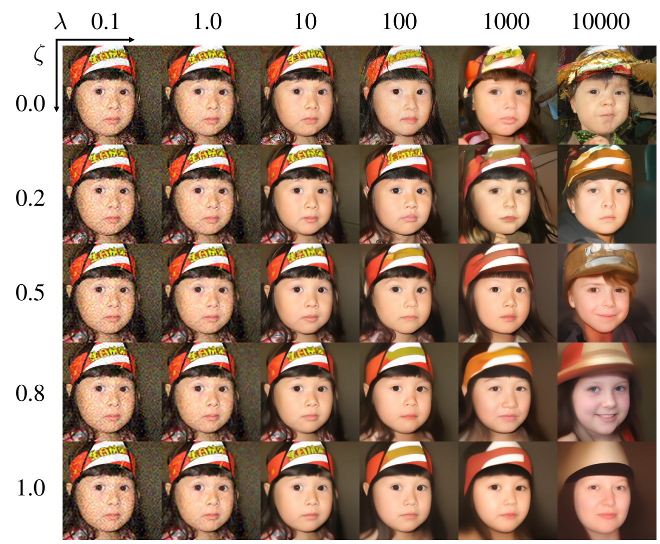

Figure 8. Effect of hyperparameters ζ and λ

5. Conclusions

In this paper, we introduce a new diffusion model-based sampling technique for plug-and-play image restoration, referred to as DiffPIR. Specifically, DiffPIR employs an HQS-based diffusion sampling approach that utilizes off-the-shelf diffusion models as plug-and-play denoising prior and solves the data subproblem in the clean image manifold. Extensive experimental results highlight the superior flexibility, efficiency, and generalizability of DiffPIR in comparison to other competitive methods.

Acknowledgements: This work was partly supported by the ETH Zürich General Fund (OK), the Alexander von Humboldt Foundation and the Huawei Fund.

References

[1] Saeed Ranjbar Alvar, Mateen Ulhaq, Hyomin Choi, and Ivan V Bajić. Joint image compression and denoising via latent-space scalability. *arXiv preprint arXiv:2205.01874*, 2022. 2

[2] Brian DO Anderson. Reverse-time diffusion equation models. *Stochastic Processes and their Applications*, 12(3):313–326, 1982. 2

[3] Sheyda Ghanbaralizadeh Bahnemiri, Mykola Ponomarenko, and Karen Egiazarian. Learning-based noise component map estimation for image denoising. *IEEE Signal Processing Letters*, 29:1407–1411, 2022. 2

[4] Stephen Boyd, Neal Parikh, Eric Chu, Borja Peleato, Jonathan Eckstein, et al. Distributed optimization and statistical learning via the alternating direction method of multipliers. *Foundations and Trends® in Machine learning*, 3(1):1–122, 2011. 1

[5] Antoni Buades, Bartomeu Coll, and J-M Morel. A non-local algorithm for image denoising. In *2005 IEEE computer society conference on computer vision and pattern recognition (CVPR'05)*, volume 2, pages 60–65. Ieee, 2005. 1

[6] Stanley H Chan, Xiran Wang, and Omar A Elgendy. Plug-and-play admm for image restoration: Fixed-point convergence and applications. *IEEE Transactions on Computational Imaging*, 3(1):84–98, 2016. 1

[7] Jooyoung Choi, Sungwon Kim, Yonghyun Jeong, Youngjune Gwon, and Sungroh Yoon. Ilvr: Conditioning method for denoising diffusion probabilistic models. *arXiv preprint arXiv:2108.02938*, 2021. 2, 3, 4, 5, 6

[8] Hyungjin Chung, Jeongsol Kim, Michael T Mccann, Marc L Klasky, and Jong Chul Ye. Diffusion posterior sampling for general noisy inverse problems. *arXiv preprint arXiv:2209.14687*, 2022. 2, 3, 5, 6, 7, 12

[9] Hyungjin Chung, Byeongsu Sim, and Jong Chul Ye. Come-closer-diffuse-faster: Accelerating conditional diffusion models for inverse problems through stochastic contraction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12413–12422, 2022. 8

[10] Kostadin Dabov, Alessandro Foi, Vladimir Katkovnik, and Karen Egiazarian. Image denoising by sparse 3-d transform-domain collaborative filtering. *IEEE Transactions on image processing*, 16(8):2080–2095, 2007. 2

[11] Aram Danielyan, Vladimir Katkovnik, and Karen Egiazarian. Image deblurring by augmented lagrangian with bm3d frame prior. In *Workshop on Information Theoretic Methods in Science and Engineering*, volume 1, 2010. 1

[12] Aram Danielyan, Vladimir Katkovnik, and Karen Egiazarian. Bm3d frames and variational image deblurring. *IEEE Transactions on image processing*, 21(4):1715–1728, 2011. 1

[13] Kamil Deja, Anna Kuzina, Tomasz Trzciński, and Jakub M Tomczak. On analyzing generative and denoising capabilities of diffusion-based deep generative models. *arXiv preprint arXiv:2206.00070*, 2022. 4

[14] Prafulla Dhariwal and Alexander Nichol. Diffusion models beat gans on image synthesis. *Advances in Neural Information Processing Systems*, 34:8780–8794, 2021. 2, 6

[15] Laurent Dinh, Jascha Sohl-Dickstein, and Samy Bengio. Density estimation using real nvp. *arXiv preprint arXiv:1605.08803*, 2016. 2

[16] Chao Dong, Chen Change Loy, Kaiming He, and Xiaoou Tang. Learning a deep convolutional network for image super-resolution. In *European conference on computer vision*, pages 184–199. Springer, 2014. 1

[17] Chao Dong, Chen Change Loy, Kaiming He, and Xiaoou Tang. Image super-resolution using deep convolutional networks. *IEEE transactions on pattern analysis and machine intelligence*, 38(2):295–307, 2015. 1

[18] Qi Dou, Cheng Ouyang, Cheng Chen, Hao Chen, Ben Glocker, Xiahai Zhuang, and Pheng-Ann Heng. Pnp-adanet: Plug-and-play adversarial domain adaptation network at unpaired cross-modality cardiac segmentation. *IEEE Access*, 7:99065–99076, 2019. 2

[19] Alain Durmus, Eric Moulines, and Marcelo Pereyra. Efficient bayesian computation by proximal markov chain monte carlo: when langevin meets moreau. *SIAM Journal on Imaging Sciences*, 11(1):473–506, 2018. 2

[20] Bradley Efron. Tweedie's formula and selection bias. *Journal of the American Statistical Association*, 106(496):1602–1614, 2011. 2

[21] Rita Fermanian, Mikael Le Pendu, and Christine Guillemot. Learned gradient of a regularizer for plug-and-play gradient descent. *arXiv preprint arXiv:2204.13940*, 2022. 2

[22] Donald Geman and Chengda Yang. Nonlinear image recovery with half-quadratic regularization. *IEEE transactions on Image Processing*, 4(7):932–946, 1995. 1

[23] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial networks. *Communications of the ACM*, 63(11):139–144, 2020. 2

[24] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in Neural Information Processing Systems*, 33:6840–6851, 2020. 2, 3

[25] Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. *arXiv preprint arXiv:2207.12598*, 2022. 3

[26] Aapo Hyvärinen and Peter Dayan. Estimation of non-normalized statistical models by score matching. *Journal of Machine Learning Research*, 6(4), 2005. 3

[27] Satoshi Iizuka, Edgar Simo-Serra, and Hiroshi Ishikawa. Globally and locally consistent image completion. *ACM Transactions on Graphics (ToG)*, 36(4):1–14, 2017. 1

[28] Zahra Kadkhodaie and Eero Simoncelli. Stochastic solutions for linear inverse problems using the prior implicit in a denoiser. *Advances in Neural Information Processing Systems*, 34:13242–13254, 2021. 2

[29] Ulugbek S Kamilov, Hassan Mansour, and Brendt Wohlberg. A plug-and-play priors approach for solving nonlinear imaging inverse problems. *IEEE Signal Processing Letters*, 24(12):1872–1876, 2017. 2

[30] Tero Karras, Miika Aittala, Timo Aila, and Samuli Laine. Elucidating the design space of diffusion-based generative models. *arXiv preprint arXiv:2206.00364*, 2022. 5

[31] Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4401–4410, 2019. 2, 6

[32] Bahjat Kawar, Michael Elad, Stefano Ermon, and Jiaming Song. Denoising diffusion restoration models. *arXiv preprint arXiv:2201.11793*, 2022. 2, 3, 4, 5, 7

[33] Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013. 2

[34] Pin-Hung Kuo, Jinshan Pan, Shao-Yi Chien, and Ming-Hsuan Yang. Learning discriminative shrinkage deep networks for image deconvolution. In *European Conference on Computer Vision*, pages 217–234. Springer, 2022. 2

[35] Maxim Kuznetsov, Daniil Polykovskiy, Dmitry P Vetrov, and Alex Zhebrak. A prior of a googol gaussians: a tensor ring induced prior for generative models. *Advances in Neural Information Processing Systems*, 32, 2019. 2

[36] Rémi Laumont, Valentin De Bortoli, Andrés Almansa, Julie Delon, Alain Durmus, and Marcelo Pereyra. Bayesian imaging using plug & play priors: when langevin meets tweedie. *SIAM Journal on Imaging Sciences*, 15(2):701–737, 2022. 2

[37] Christian Ledig, Lucas Theis, Ferenc Huszár, Jose Caballero, Andrew Cunningham, Alejandro Acosta, Andrew Aitken, Alykhan Tejani, Johannes Totz, Zehan Wang, et al. Photo-realistic single image super-resolution using a generative adversarial network. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4681–4690, 2017. 1

[38] Andreas Lugmayr, Martin Danelljan, Andres Romero, Fisher Yu, Radu Timofte, and Luc Van Gool. Repaint: Inpainting using denoising diffusion probabilistic models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11461–11471, 2022. 2, 3, 5, 6

[39] Tim Meinhardt, Michael Moller, Caner Hazirbas, and Daniel Cremers. Learning proximal operators: Using denoising networks for regularizing inverse imaging problems. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1781–1790, 2017. 2

[40] Alex Nichol, Prafulla Dhariwal, Aditya Ramesh, Pranav Shyam, Pamela Mishkin, Bob McGrew, Ilya Sutskever, and Mark Chen. Glide: Towards photorealistic image generation and editing with text-guided diffusion models. *arXiv preprint arXiv:2112.10741*, 2021. 3

[41] Alexander Quinn Nichol and Prafulla Dhariwal. Improved denoising diffusion probabilistic models. In *International Conference on Machine Learning*, pages 8162–8171. PMLR, 2021. 2

[42] Neal Parikh, Stephen Boyd, et al. Proximal algorithms. *Foundations and trends® in Optimization*, 1(3):127–239, 2014. 4

[43] Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. Hierarchical text-conditional image generation with clip latents. *arXiv preprint arXiv:2204.06125*, 2022. 2

[44] Ali Razavi, Aaron Van den Oord, and Oriol Vinyals. Generating diverse high-fidelity images with vq-vae-2. *Advances in neural information processing systems*, 32, 2019. 2

[45] Yaniv Romano, Michael Elad, and Peyman Milanfar. The little engine that could: Regularization by denoising (red). *SIAM Journal on Imaging Sciences*, 10(4):1804–1844, 2017. 2

[46] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. Imagenet large scale visual recognition challenge. *International journal of computer vision*, 115(3):211–252, 2015. 2, 6

[47] Ernest Ryu, Jialin Liu, Sicheng Wang, Xiaohan Chen, Zhangyang Wang, and Wotao Yin. Plug-and-play methods provably converge with properly trained denoisers. In *International Conference on Machine Learning*, pages 5546–5557. PMLR, 2019. 2

[48] Chitwan Saharia, William Chan, Huiwen Chang, Chris Lee, Jonathan Ho, Tim Salimans, David Fleet, and Mohammad Norouzi. Palette: Image-to-image diffusion models. In *ACM SIGGRAPH 2022 Conference Proceedings*, pages 1–10, 2022. 3

[49] Chitwan Saharia, Jonathan Ho, William Chan, Tim Salimans, David J Fleet, and Mohammad Norouzi. Image super-resolution via iterative refinement. *arXiv preprint arXiv:2104.07636*, 2021. 2, 3

[50] Jascha Sohl-Dickstein, Eric Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. In *International Conference on Machine Learning*, pages 2256–2265. PMLR, 2015. 2

[51] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. *arXiv preprint arXiv:2010.02502*, 2020. 3, 5, 6

[52] Yang Song and Stefano Ermon. Generative modeling by estimating gradients of the data distribution. *Advances in Neural Information Processing Systems*, 32, 2019. 3

[53] Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. *arXiv preprint arXiv:2011.13456*, 2020. 2, 3, 12

[54] Singanallur V Venkatakrishnan, Charles A Bouman, and Brendt Wohlberg. Plug-and-play priors for model based reconstruction. In *2013 IEEE Global Conference on Signal and Information Processing*, pages 945–948. IEEE, 2013. 1, 2

[55] Xinyi Wei, Hans van Gorp, Lizeth Gonzalez-Carabarin, Daniel Freedman, Yonina C Eldar, and Ruud JG van Sloun. Deep unfolding with normalizing flow priors for inverse problems. *IEEE Transactions on Signal Processing*, 70:2962–2971, 2022. 2

[56] Zhisheng Xiao, Karsten Kreis, and Arash Vahdat. Tackling the generative learning trilemma with denoising diffusion gans. *arXiv preprint arXiv:2112.07804*, 2021. 6

[57] Kai Zhang, Yawei Li, Wangmeng Zuo, Lei Zhang, Luc Van Gool, and Radu Timofte. Plug-and-play image restoration with deep denoiser prior. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2021. 1, 2, 3, 4, 5, 7, 13

[58] Kai Zhang, Wangmeng Zuo, Yunjin Chen, Deyu Meng, and Lei Zhang. Beyond a gaussian denoiser: Residual learning of

deep cnn for image denoising. *IEEE transactions on image processing*, 26(7):3142–3155, 2017. 1

[59] Kai Zhang, Wangmeng Zuo, Shuhang Gu, and Lei Zhang. Learning deep cnn denoiser prior for image restoration. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3929–3938, 2017. 1, 2

[60] Kai Zhang, Wangmeng Zuo, and Lei Zhang. Deep plug-and-play super-resolution for arbitrary blur kernels. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1671–1681, 2019. 2