Diffusion Models for Solving Inverse Problems via Posterior Sampling with Piecewise Guidance

Saeed Mohseni-Sehdeh†, Walid Saad†, Kei Sakaguchi§, and Tao Yu§
†Bradley Department of Electrical and Computer Engineering, Virginia Tech, Alexandria, VA, 22305, USA.
§Department of Electrical and Electronic Engineering, Institute of Science Tokyo, Tokyo, Japan.
Emails: {saeedmohseni, walids}@vt.edu, {sakaguchi, yutao}@mobile.ee.titech.ac.jp

arXiv:2507.18654v1 [cs.LG] 22 Jul 2025

***Abstract*—Diffusion models are powerful tools for sampling from high-dimensional distributions by progressively transforming pure noise into structured data through a denoising process. When equipped with a guidance mechanism, these models can also generate samples from conditional distributions. In this paper, a novel diffusion-based framework is introduced for solving inverse problems using a piecewise guidance scheme. The guidance term is defined as a piecewise function of the diffusion timestep, facilitating the use of different approximations during high-noise and low-noise phases. This design is shown to effectively balance computational efficiency with the accuracy of the guidance term. Unlike task-specific approaches that require retraining for each problem, the proposed method is problem-agnostic and readily adaptable to a variety of inverse problems. Additionally, it explicitly incorporates measurement noise into the reconstruction process. The effectiveness of the proposed framework is demonstrated through extensive experiments on image restoration tasks, specifically image inpainting and super-resolution. Using a class conditional diffusion model for recovery, compared to the ΠGDM baseline, the proposed framework achieves a reduction in inference time of 25% for inpainting with both random and center masks, and 23% and 24% for $4\times$ and $8\times$ super-resolution tasks, respectively, while incurring only negligible loss in PSNR and SSIM.**

***Index Terms*—Diffusion models, Inverse problems, Score function, Accelerated sampling.**

I. Introduction

Diffusion models are a class of deep generative models designed to sample from complex data distributions. Diffusion models have been shown to outperform alternatives like generative adversarial networks (GANs) in image synthesis tasks and currently represent the state of the art in this domain [1]. The core idea behind diffusion models is to gradually remove the structure from given input data through a forward diffusion process, transforming it into a tractable distribution, which is typically Gaussian white noise. The model is then trained to learn the reverse process, effectively denoising the data step by step to reconstruct a sample that closely approximates the original data distribution. Diffusion models have been applied across various fields, including computer vision [2], [3], natural language processing [4], [5], audio synthesis [6], and medical image reconstruction [7], [8].

Once trained on a dataset from a specific distribution, diffusion models can generate samples that follow that distribution [9]. These samples are inherently random, as the generation process begins with a noise vector sampled randomly. Consequently, the resulting samples may correspond to any region within the support of the learned distribution. Diffusion models can be employed for conditional sampling [10] when their denoising process is adapted to incorporate auxiliary information, enabling the generation of samples that are consistent with the provided conditions.

This conditional sampling capability makes diffusion models promising candidates for solving inverse problems, where the objective is to reconstruct a degraded signal. The core idea is to recover the original signal by sampling from the posterior distribution conditioned on the observed degraded input. In the context of image inverse problems, this approach has demonstrated the ability to produce perceptually high-quality outputs [11]. This motivates the use of diffusion models as effective tools for solving inverse problems.

Numerous challenges arise when employing diffusion models to solve inverse problems. These models are typically large and require extensive computational resources and substantial amounts of data for effective training. Given the wide variety of inverse problems, it is impractical to train a separate diffusion model for each specific task [10], [12], [13]. Therefore, a key challenge is developing a unified framework that can address multiple inverse problems using a single, pre-trained diffusion model without the need for task-specific retraining. Another key challenge is maintaining high-quality restoration, typically measured by standard metrics such as the peak signal-to-noise ratio (PSNR) and structural similarity index (SSIM). A general-purpose system must achieve results that are competitive with models trained specifically for individual inverse problems. In addition, computational efficiency is critical: The faster the conditional sampling process, the more practical the framework becomes.

A. Related Works

Various deep neural network-based techniques have been proposed for solving inverse problems [14]–[20] and the aforementioned challenges. These methods can be broadly categorized into supervised (problem-specific) and unsupervised (problem-agnostic) approaches. In supervised methods, the degradation model is known during both training and inference. In contrast, unsupervised methods assume that the degradation is only known at inference time. The unsupervised approaches are particularly appealing, as they better reflect real-world scenarios where access to degradation models

This research was supported by the U.S. National Science Foundation under Grant CNS-2210254.

during training is often limited or unavailable, and these approaches do not rely on training problem-specific models.

One class of unsupervised deep neural network techniques addresses inverse problems by iteratively applying a pretrained model [15]–[17], [21]. Methods such as plug-and-play (PnP) [15], regularization by denoising (RED) [16], and their successors in [17] and [21] incorporate a denoiser into an iterative recovery process. Another line of work leverages (GANs) [18]–[20], where the latent space of a pretrained GAN is searched to find latent codes that generate images best aligned with the observed measurements. These methods often require a large number of iterations to converge to a satisfactory solution making them time inefficient and not practical for real use cases in which low inference time is critical.

Diffusion models are another class of deep neural networks that can be used for solving inverse problems, with applications in both supervised [22]–[25] and unsupervised [1], [26]–[28] settings. Denoising diffusion reconstruction models (DDRM) [10] represent a diffusion-based approach for solving unsupervised inverse problems. In this method, denoising is performed in the spectral domain of the degradation matrix, and the results are subsequently transformed back into the original image space. While DDRM has demonstrated promising restoration quality, its effectiveness is limited in scenarios where the relationship between the measurement noise level and the diffusion noise level in the spectral domain is weak.

Another diffusion-based method for unsupervised inverse problems is the pseudoinverse-guided diffusion model (ΠGDM) [12]. This approach computes the guidance term using a one-step denoising approximation of the posterior distribution of the data conditioned on the noisy latent diffusion variable. Unlike DDRM, ΠGDM enables updates regardless of the levels of diffusion and measurement noise. Although effective, ΠGDM requires computing the derivative of the denoiser's output with respect to its input, a computationally intensive operation, particularly when the denoiser's complexity and the dimensionality of the data increase.

The computational and time complexity of GAN-based methods and ΠGDM, the sensitivity of DDRM to measurement noise, and the need for fine-tuning and retraining in supervised approaches highlight the need for a new, reliable, problem-agnostic framework for solving inverse problems. Such a framework should deliver high-quality restoration with low inference time and computational cost, while also explicitly accounting for measurement noise, which is almost always present in real-world scenarios.

B. Contributions

The main contribution of this paper is the introduction of a new diffusion-based framework for solving inverse problems, which uses a piecewise function to approximate the guidance term. This approach preserves accuracy while significantly reducing computational complexity. Moreover, the proposed method explicitly accounts for measurement noise. Specifically, our key contributions include:

- We propose a novel, problem-agnostic, diffusion-based framework for solving inverse problems via posterior sampling, which employs a *piecewise guidance function* that depends on both the measurement and the noisy latent variable at each diffusion time step.
- We show how the proposed method leverages the varying noise and information content of latent variables across time steps to compute *time-dependent guidance* values, enabling a more effective tradeoff between computational efficiency and reconstruction accuracy.
- We expose how the proposed approach explicitly accounts for measurement noise during guidance computation, which is often overlooked in existing methods. As a result, the proposed approach aligns the guidance term more consistently with the measurement distribution, enhancing its performance in real-world scenarios where the measurement is affected by noise.
- We derive mathematical expressions that quantify the quality of the approximation in terms of the Kullback–Leibler (KL) divergence between the true and approximated distributions used in the guidance computation, providing insights into how problem parameters affect the effectiveness of the approximation.
- Extensive experiments show that the proposed method reduces inference time while maintaining comparable performance in terms of PSNR and SSIM. Compared to ΠGDM, it achieves a reduction of 25% for both inpainting with random and center masks, and 23% and 24% for $4\times$ and $8\times$ super-resolution, respectively, using a class conditional diffusion model.

The rest of this paper is organized as follows. Section II introduces the problem and its challenges. Then, Section III details the proposed method. Section IV presents a theoretical analysis of how the problem parameters influence the proposed solution, and Section V presents and analyzes the simulation results. Finally, conclusions are drawn in Section VI.

II. Problem Statement

Inverse problems are a class of signal processing problems whose objective is to recover an original signal from observed measurements. Consider the following model that relates an original signal $\boldsymbol{x}_0 \in \mathbb{R}^n$ with a corresponding measurement $\boldsymbol{y} \in \mathbb{R}^m$ of that signal:

$$\boldsymbol{y} = \boldsymbol{C}\boldsymbol{x}_0 + \boldsymbol{z}, \tag{1}$$

where $\boldsymbol{C} \in \mathbb{R}^{m\times n}$ represents the measurement model and $\boldsymbol{z} \in \mathbb{R}^m$ is an i.i.d measurement noise distributed according to $\mathcal{N}(\boldsymbol{0}, \sigma_z^2\boldsymbol{I})$. In practical inverse problems $m < n$. The objective is to recover $\boldsymbol{x}_0$ from the measurement $\boldsymbol{y}$. Inverse problems are central to domains such as medical imaging [29], computer vision [30] (e.g., image deblurring, super-resolution), and astronomy [31].

Inverse problems are inherently ill-posed [32]; small changes in the observations can lead to large variations in the solution. Moreover, solutions to inverse problems may not be unique [32], requiring the incorporation of prior knowledge and regularization to obtain meaningful results. This "ill-posedness" and non-uniqueness arise from the indirect, incomplete, and noisy nature of the measurements. In (1),

the incompleteness of the measurement is reflected in the condition $m < n$.

Different approaches have been proposed for solving inverse problems, such as regularization-based techniques, which impose constraints on the solution, such as sparsity or low-rank structure [29], and statistical approaches, which assume a prior distribution on the original signal and solve the problem by finding suitable estimators [29]. One method that has been shown to produce high-quality solutions is *posterior sampling* [11]. The core idea of this approach is to solve (1) by generating samples from the conditional distribution $p(\boldsymbol{x}_0|\boldsymbol{y})$. Diffusion models can be used for sampling from such distributions, provided their denoising procedures are appropriately modified.

Diffusion models perform sampling from distributions by first structurally removing data information and increasing noise. This is achieved by forming a chain of random variables which progressively get noisier. This procedure is called *forward path* and it continues until the point that the structure in the data is reduced to pure noise, where sampling from that is practical. Diffusion models by learning the reverse process of removing noise gradually from the noisy data known as *backward path*, achieve sampling from the distribution.

When the chain of created noisy random variables modeled as a continuum, the forward path of diffusion models can be expressed by a stochastic differential equation (SDE) of the form

$$d\boldsymbol{x} = \boldsymbol{f}(\boldsymbol{x}, t)dt + g(t)d\boldsymbol{w}_f, \tag{2}$$

where $\boldsymbol{f} : \mathbb{R}^{n+1} \to \mathbb{R}^n$ is the drift term which represents the deterministic trend of the state variable $\boldsymbol{x}$ and slowly removes the presence of data, $g : \mathbb{R} \to \mathbb{R}$ represents the intensity of the randomness affecting the state variable and gradually increases the presence of noise in the data and $\boldsymbol{w}_f$ is standard Wiener processes. One such choice that has been adopted heavily are $\boldsymbol{f}(\boldsymbol{x}, t) = -\frac{1}{2}\beta(t)\boldsymbol{x}$ and $g(t) = \sqrt{\beta(t)}$, in which $\beta(t) : \mathbb{R} \to (0, 1)$ is a monotonically increasing function of t.

As shown in [33], (2) can be solved in reverse by solving

$$d\boldsymbol{x} = [\boldsymbol{f}(\boldsymbol{x}, t) - \frac{1}{2}g(t)^2\nabla_{\boldsymbol{x}} \log p_t(\boldsymbol{x})] + g(t)d\boldsymbol{w}_b, \tag{3}$$

where the time variable t moves in the opposite direction of (2), $\boldsymbol{w}_b$ is the standard Wiener process in reverse time, and p_t is the probability distribution function (PDF) of the random variable $\boldsymbol{x}(t)$. As shown in [33], (2) and (3) are equivalent in the sense that the PDF of $\boldsymbol{x}(t)$ is the same for the solutions of both for all t. Thus, solving (3) can result in samples from $p_t(\boldsymbol{x})$ for each time step t, and $t = 0$ corresponds to the PDF of data. In practice, (3) is solved using numerical methods such as Euler–Maruyama, provided that $\nabla_{\boldsymbol{x}} \log p_t(\boldsymbol{x})$ (the score function) is known for all the noise levels, which can be approximated with using score estimating models. In numerical solvers the continuous process $\boldsymbol{x}(t)$ is discretized as $\boldsymbol{x}(t) = \boldsymbol{x}_t$ where $t \in \{1, 2, \ldots, T\}$. Then, (3) is solved by assuming that $\boldsymbol{x}_T$ is almost pure noise, and a random sample from that serves as the starting point of the numerical solvers.

In order to use diffusion models for solving inverse problems using posterior sampling, one can solve (3) with replacing $\nabla_{\boldsymbol{x}} \log p_t(\boldsymbol{x})$ with its problem specific counterpart, $\nabla_{\boldsymbol{x}_t} \log p_t(\boldsymbol{x}_t|\boldsymbol{y})$, where $p_t(\boldsymbol{x}_t|\boldsymbol{y})$ represents the PDF of the noisy variable at time step t given the measurement. To calculate the problem-specific score function, Bayes' rule can be used

$$\nabla_{\boldsymbol{x}_t} \log p_t(\boldsymbol{x}_t|\boldsymbol{y}) = \nabla_{\boldsymbol{x}_t} \log p_t(\boldsymbol{x}_t) + \nabla_{\boldsymbol{x}_t} \log p_t(\boldsymbol{y}|\boldsymbol{x}_t), \tag{4}$$

where $\nabla_{\boldsymbol{x}_t} \log p_t(\boldsymbol{x}_t)$ can be approximated using score estimation networks trained on the original data and it is part of the diffusion model training process. The second term (guidance term) in (4) adapts the sampling procedure to produce samples that are consistent with the given measurement.

Unfortuantely, the calculation of a closed-form expression for $\nabla_{\boldsymbol{x}_t} \log p_t(\boldsymbol{y}|\boldsymbol{x}_t)$ is generally interactable. The reason for this is the presence of the diffusion noise on both sides of $p_t(\boldsymbol{y}|\boldsymbol{x}_t)$. To elaborate, consider the following equation derived from (2)

$$\boldsymbol{x}_t = \sqrt{\bar{\alpha}_t}\boldsymbol{x}_0 + \sqrt{1 - \bar{\alpha}_t}\boldsymbol{\epsilon}, \tag{5}$$

where $\bar{\alpha}_t = \prod_{i=1}^{t} \alpha_t$, $\alpha_t = 1 - \beta_t$ and $\beta_t = \beta(t)$ for $t \in \{1, 2, \ldots, T\}$ and $\boldsymbol{\epsilon} \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{I})$. Then, for $p_t(\boldsymbol{y}|\boldsymbol{x}_t)$ we have

$$\begin{aligned} p_t(\boldsymbol{y}|\boldsymbol{x}_t) &= p(\boldsymbol{C}\boldsymbol{x}_0 + \boldsymbol{z} = \boldsymbol{y}|\sqrt{\bar{\alpha}_t}\boldsymbol{x}_0 + \sqrt{1 - \bar{\alpha}_t}\boldsymbol{\epsilon} = \boldsymbol{x}_t) \\ &= p\left(\frac{\boldsymbol{C}\boldsymbol{x}_t}{\sqrt{\bar{\alpha}_t}} - \frac{\sqrt{1 - \bar{\alpha}_t}}{\sqrt{\bar{\alpha}_t}}\boldsymbol{C}\boldsymbol{\epsilon} + \boldsymbol{z} = \boldsymbol{y}|\boldsymbol{x}_t\right). \end{aligned} \tag{6}$$

The presence of diffusion noise $\boldsymbol{\epsilon}$ in both the conditioned and conditioning parts of (6) makes the calculation interactable.

The computation of the problem-specific score function is crucial to the performance of diffusion-based methods for solving inverse problems. For the approach to be practical, the score function must be computable without training a separate model, as model retraining for each new problem reduces generality and increases computational overhead. Moreover, the computation should be efficient to avoid compromising inference speed and must account for measurement noise inherent in the observations. To address these challenges, we propose a method that enables efficient problem-specific score computation that accounts for measurement noise without model retraining. The next section outlines the key components of our approach.

III. Proposed Conditional Score Estimation

We propose to approximate $p_t(\boldsymbol{y}|\boldsymbol{x}_t)$ by using a piecewise function that varies across different time steps along the diffusion path. The idea is that (6) can be simplified if $\frac{\sqrt{1-\bar{\alpha}_t}}{\sqrt{\bar{\alpha}_t}}\boldsymbol{C}\boldsymbol{\epsilon} = \boldsymbol{0}$. There is no control over the value of $\boldsymbol{C}\boldsymbol{\epsilon}$ as the diffusion noise is random and $\boldsymbol{C}$ is fixed, thus the only way to make the expression small is through the coefficient ($\frac{\sqrt{1-\bar{\alpha}_t}}{\sqrt{\bar{\alpha}_t}}$). By design of the diffusion models, this coefficient is always small at lower time steps t, near the end of the backward path as for those values we have $\bar{\alpha}_t \approx 1$. This condition is a property of diffusion models and does not depend on the inverse problem, making it a versatile property. Under this

observation, for low values of t in the diffusion path, $p_t(\boldsymbol{y}|\boldsymbol{x}_t)$ can be computed as follows

$$\begin{aligned} p_t(\boldsymbol{y}|\boldsymbol{x}_t) &= p\left(\frac{1}{\sqrt{\bar{\alpha}_t}}\boldsymbol{C}\boldsymbol{x}_t - \frac{\sqrt{1-\bar{\alpha}_t}}{\sqrt{\bar{\alpha}_t}}\boldsymbol{C}\boldsymbol{\epsilon} + \boldsymbol{z} = \boldsymbol{y}|\boldsymbol{x}_t\right), \\ &\approx p\left(\boldsymbol{z} = \boldsymbol{y} - \frac{1}{\sqrt{\bar{\alpha}_t}}\boldsymbol{C}\boldsymbol{x}_t|\boldsymbol{x}_t\right) \qquad t < T_0, \\ &= p\left(\boldsymbol{z} = \boldsymbol{y} - \frac{1}{\sqrt{\bar{\alpha}_t}}\boldsymbol{C}\boldsymbol{x}_t\right) = \mathcal{N}\left(\boldsymbol{y}; \frac{1}{\sqrt{\bar{\alpha}_t}}\boldsymbol{C}\boldsymbol{x}_t, \sigma_z^2\boldsymbol{I}\right), \end{aligned}$$

where T_0 is the highest time step at which this approximation performs well. The final expression follows from the assumption that the measurement noise $\boldsymbol{z}$ and the diffusion noise $\boldsymbol{\epsilon}$ are independent, which is a reasonable assumption because measurement plays no role in the creation of the diffusion noise. With this assumption, the score function will be

$$\nabla_{\boldsymbol{x}_t} \log p_t(\boldsymbol{y}|\boldsymbol{x}_t) = \frac{1}{\sigma_z^2\sqrt{\bar{\alpha}_t}}\boldsymbol{C}^T\left(\boldsymbol{y} - \frac{1}{\sqrt{\bar{\alpha}_t}}\boldsymbol{C}\boldsymbol{x}_t\right) \qquad t < T_0. \tag{7}$$

In order to compute the guidance term for $T > t > T_0$, we use the approach in [12] in which the guidance term is approximated with one-step denoising. To elaborate, note that the following equation holds:

$$\begin{aligned} p_t(\boldsymbol{y}|\boldsymbol{x}_t) &= \int p(\boldsymbol{y}|\boldsymbol{x}_0, \boldsymbol{x}_t)p(\boldsymbol{x}_0|\boldsymbol{x}_t)d\boldsymbol{x}_0 \\ &= \int p(\boldsymbol{y}|\boldsymbol{x}_0)p(\boldsymbol{x}_0|\boldsymbol{x}_t)d\boldsymbol{x}_0. \end{aligned} \tag{8}$$

As suggested in [12], $p(\boldsymbol{x}_0|\boldsymbol{x}_t)$ can be approximated by the one step denoising that estimates $\boldsymbol{x}_0$ from $\boldsymbol{x}_t$ and to account for the randomness in denoising process, $p(\boldsymbol{x}_0|\boldsymbol{x}_t)$ is modeled as a Gaussian distribution around the estimated $\boldsymbol{x}_0$. In other words

$$p(\boldsymbol{x}_0|\boldsymbol{x}_t) \sim \mathcal{N}(\hat{\boldsymbol{x}}_0^t, r_t^2\boldsymbol{I}), \tag{9}$$

where $\hat{\boldsymbol{x}}_0^t$ is the estimate of the $\boldsymbol{x}_0$ from the given $\boldsymbol{x}_t$ obtained from the denoising mechanism of the diffusion models and r_t controls the variance of the approximate distribution. Using (8) and (9), it can be shown that

$$p_t(\boldsymbol{y}|\boldsymbol{x}_t) \sim \mathcal{N}(\boldsymbol{C}\hat{\boldsymbol{x}}_0^t, r_t^2\boldsymbol{C}\boldsymbol{C}^T + \sigma_z^2\boldsymbol{I}). \tag{10}$$

The score function then follows as

$$\nabla_{\boldsymbol{x}_t} \log p_t(\boldsymbol{y}|\boldsymbol{x}_t) \approx (\frac{\partial\hat{\boldsymbol{x}}_0^t}{\partial\boldsymbol{x}_t})^T\boldsymbol{C}^T(r_t^2\boldsymbol{C}\boldsymbol{C}^T + \sigma_z^2\boldsymbol{I})^{-1}(\boldsymbol{y} - \boldsymbol{C}\hat{\boldsymbol{x}}_0^t). \tag{11}$$

Putting this altogether, the proposed guidance term will be:

$$\nabla_{\boldsymbol{x}_t} \log p_t(\boldsymbol{x}_t|\boldsymbol{y}) = \nabla_{\boldsymbol{x}_t} \log p_t(\boldsymbol{x}_t) \tag{12}$$

$$+ \begin{cases} \frac{1}{\sigma_z^2\sqrt{\bar{\alpha}_t}}\boldsymbol{C}^T(\boldsymbol{y} - \frac{1}{\sqrt{\bar{\alpha}_t}}\boldsymbol{C}\boldsymbol{x}_t) & \text{if } t < T_0, \\ (\frac{\partial\hat{\boldsymbol{x}}_0^t}{\partial\boldsymbol{x}_t})^T\boldsymbol{C}^T(r_t^2\boldsymbol{C}\boldsymbol{C}^T + \sigma_z^2\boldsymbol{I})^{-1}(\boldsymbol{y} - \boldsymbol{C}\hat{\boldsymbol{x}}_0^t), & \text{o.w.} \end{cases} \tag{13}$$

Algorithm 1 outlines the complete step-by-step implementation of the proposed method. It reconstructs the original data from the given measurement matrix $\boldsymbol{C}$ and the observed measurement $\boldsymbol{y}$. The key distinction between this approach and ΠGDM lies in the use of a conditional statement that enables piecewise guidance based on the diffusion time step. Unlike ΠGDM, which uses the same formula to compute the problem-specific score function across all diffusion time steps, the proposed method employs a computationally simpler function at lower time steps, leading to accelerated inference at those time steps.

Algorithm 1 Posterior sampling for inverse problems via piecewise guidance

Inputs: $\boldsymbol{y} \in \mathbb{R}^m$, $\boldsymbol{C} \in \mathbb{R}^{m\times n}$, σ_z, $\eta \in [0,1]$, diffusion-based noise predictor model $\boldsymbol{D}$, r_t (estimation noise sequence) T_0, k_1 and k_2 (for tuning).
Initialize: $\boldsymbol{x} \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{I})$
for $t = N$ to 1 **do**
  $\hat{\boldsymbol{\epsilon}} \leftarrow \boldsymbol{D}(\boldsymbol{x}, t)$ ▷ Predicts the added noise at time step t
  $\hat{\boldsymbol{x}} \leftarrow \frac{\boldsymbol{x} - \sqrt{1-\bar{\alpha}_t}}{\sqrt{\bar{\alpha}_t}}$
  $c_1 \leftarrow \eta\sqrt{(1 - \frac{\bar{\alpha}_t}{\bar{\alpha}_{t-1}})\frac{1-\bar{\alpha}_{t-1}}{1-\bar{\alpha}_t}}$ ▷ c_1, c_2 : DDIM coeficients
  $c_2 \leftarrow \sqrt{1 - \bar{\alpha}_{t-1} - c_1^2}$
  if $i < T_0$ **then**
    $\boldsymbol{g} \leftarrow k_1\frac{1}{\sigma_z^2\sqrt{\bar{\alpha}_t}}\boldsymbol{C}^T(\boldsymbol{y} - \boldsymbol{C}\frac{\boldsymbol{x}}{\sqrt{\bar{\alpha}_t}})$
  else
    $\boldsymbol{g} \leftarrow k_2(\frac{\partial\hat{\boldsymbol{x}}}{\partial\boldsymbol{x}})^T\boldsymbol{C}^T(r_t^2\boldsymbol{C}\boldsymbol{C}^T + \sigma_z^2\boldsymbol{I})^{-1}(\boldsymbol{y} - \boldsymbol{C}\hat{\boldsymbol{x}})$
  end if
  $\boldsymbol{\epsilon} \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{I})$
  $\boldsymbol{x} \leftarrow \sqrt{\bar{\alpha}_{t-1}} + c_1\boldsymbol{\epsilon} + c_2\hat{\boldsymbol{\epsilon}} + \sqrt{\bar{\alpha}_t}\boldsymbol{g}$
end for

The computational advantage of using a piecewise guidance function can be understood by observing (11), which involves the computation of a vector-Jacobian product. This requires taking derivatives of the denoiser's output with respect to its input, a computationally expensive operation. Additionally, this expression includes the inversion of a matrix, which further increases the computational cost. In contrast to [12], which uses these computationally expensive operations across all the diffusion steps, our piecewise guidance approach substitutes such operations with the simpler form given in (7) for certain time steps, reducing the computational cost. Using a piecewise function to approximate the score function takes advantage of the different levels of diffusion noise at each time step to accelerate sampling in the final diffusion steps. It offers a reduction in inference time while maintaining performance. The proposed method requires specifying the parameter T_0. Next, we provide preliminary insights into how T_0 influences performance, offering an intuitive understanding of its selection.

IV. Theoretical Analysis of Piecewise Score Estimation

In this section, we provide a theoretical analysis of the proposed method by examining the approximation of the guidance term. We assess the quality of this approximation by deriving an expression for the divergence between the true and estimated conditional distributions, from which problem-specific scores are obtained. This analysis offers insight into the effectiveness of the approximation and highlights the factors that influence its accuracy.

To formalize this, we begin by considering the Kullback–Leibler (KL) divergence between the ground-truth conditional distribution $p_0(\boldsymbol{y}|\boldsymbol{x}_t)$ and its proposed approximation.

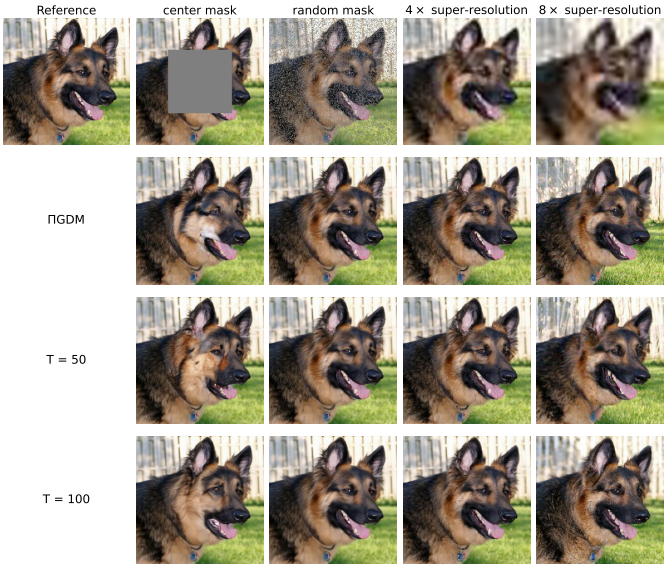


Fig. 1. Restoration results on four inverse problems, inpainting with center mask (second column from left), inpainting with random mask, covering 30% of the pixels (third column form left), 4× super-resolution (fourth column from left) and 8× super-resolution (the fifth column from left) for different values of the T_0 using an image of a dog using **class conditional** diffusion model. T_0 controls when each component of the piece-wise guidance term is active. ΠGDM corresponds to $T_0 = 0$.

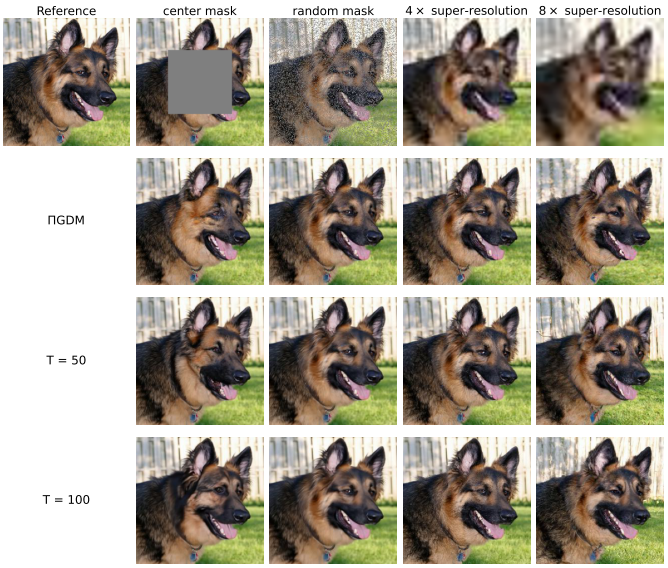


Fig. 2. Restoration results on four inverse problems, inpainting with center mask (second column from left), inpainting with random mask, covering 30% of the pixels (third column form left), 4× super-resolution (fourth column from left) and 8× super-resolution (the fifth column from left) for different values of the T_0 using an image of a dog using **class unconditional** diffusion model. T_0 controls when each component of the piece-wise guidance term is active. ΠGDM corresponds to $T_0 = 0$.

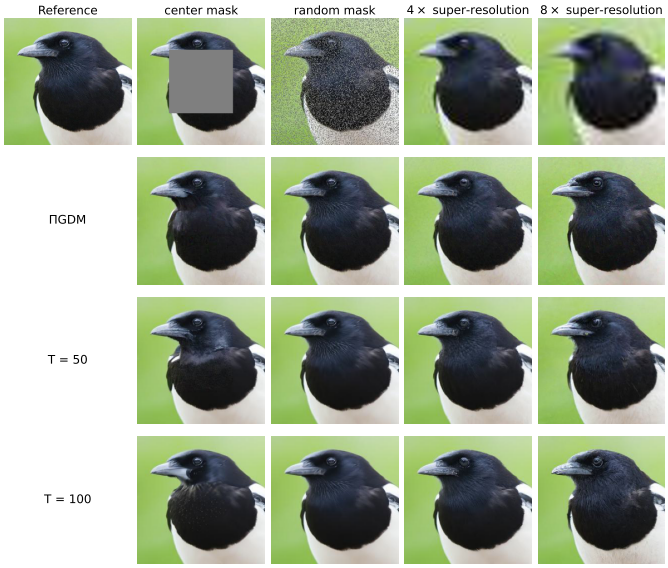


Fig. 3. Restoration results on four inverse problems, inpainting with center mask (second column from left), inpainting with random mask, covering 30% of the pixels (third column form left), 4× super-resolution (fourth column from left) and 8× super-resolution (the fifth column from left) for different values of the T_0 using an image of a magpie using **class conditional** diffusion model. T_0 controls when each component of the piece-wise guidance term is active. ΠGDM corresponds to $T_0 = 0$.

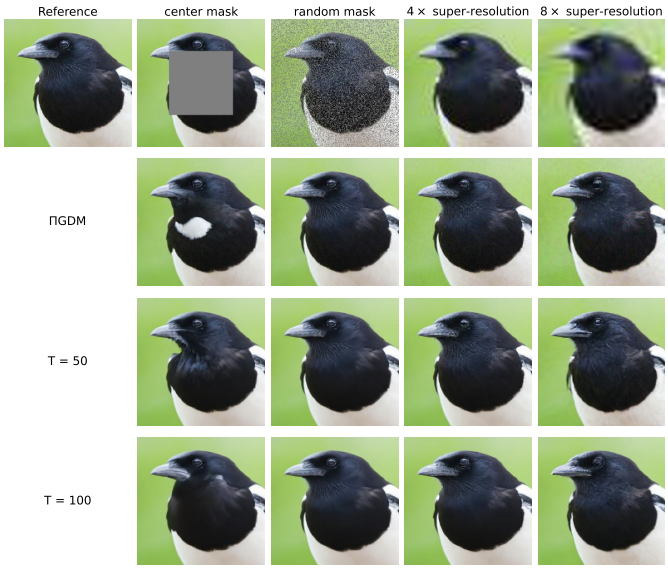


Fig. 4. Restoration results on four inverse problems, inpainting with center mask (second column from left), inpainting with random mask, covering 30% of the pixels (third column form left), 4× super-resolution (fourth column from left) and 8× super-resolution (the fifth column from left) for different values of the T_0 using an image of a magpie using **class unconditional** diffusion model. T_0 controls when each component of the piece-wise guidance term is active. ΠGDM corresponds to $T_0 = 0$.

The following theorem provides a closed-form expression for this divergence under the assumption that the noise term becomes negligible, i.e., $\frac{\sqrt{1-\bar{\alpha}_t}}{\sqrt{\bar{\alpha}_t}}\boldsymbol{C}\boldsymbol{\epsilon} \approx \mathbf{0}\ \forall t < T_0$, which is a reasonable assumption and inherent to the diffusion models architecture [9].

Theorem 1. Suppose the value of the added noise for the latent diffusion variable at time step t is $\boldsymbol{v}_t \in \mathbb{R}^n$, i.e., $\boldsymbol{x}_t = \sqrt{1-\bar{\alpha}_t}\boldsymbol{x}_0 + \sqrt{\bar{\alpha}_t}\boldsymbol{v}_t$, then the value of the KL divergence between the true distribution $p_0(\boldsymbol{y}|\boldsymbol{x}_t)$ and the approximated version $p_a(\boldsymbol{y}|\boldsymbol{x}_t)$ is $\frac{1}{2\sigma_z^2}\frac{1-\bar{\alpha}_t}{\bar{\alpha}_t}\|\boldsymbol{C}\boldsymbol{v}_t\|_2^2$.

Proof. To prove Theorem 1, we need the following lemma from [34]:

Lemma 1. If $\boldsymbol{X}_1 \sim \mathcal{N}(\boldsymbol{\mu}_1, \boldsymbol{\Sigma}_1) \in \mathbb{R}^n$ and $\boldsymbol{X}_2 \sim \mathcal{N}(\boldsymbol{\mu}_2, \boldsymbol{\Sigma}_2) \in \mathbb{R}^n$, then the KL divergence between the distribution of theses two variables is given by

$$\frac{1}{2}\left[\log\frac{|\boldsymbol{\Sigma}_2|}{|\boldsymbol{\Sigma}_1|} - n + \text{tr}(\boldsymbol{\Sigma}_2^{-1}\boldsymbol{\Sigma}_1) + (\boldsymbol{\mu}_2 - \boldsymbol{\mu}_1)^T\boldsymbol{\Sigma}_2^{-1}(\boldsymbol{\mu}_2 - \boldsymbol{\mu}_1)\right].$$

The ground truth conditional distribution $p_0(\boldsymbol{y}|\boldsymbol{x}_t)$ can be written as

$$\begin{aligned} p_t(\boldsymbol{y}|\boldsymbol{x}_t) &= p(\boldsymbol{C}\boldsymbol{x}_0 + \boldsymbol{z} = \boldsymbol{y}|\sqrt{\bar{\alpha}_t}\boldsymbol{x}_0 + \sqrt{1-\bar{\alpha}_t}\boldsymbol{v}_t = \boldsymbol{x}_t) \\ &= p\left(\boldsymbol{z} = \boldsymbol{y} - \frac{1}{\sqrt{\bar{\alpha}_t}}\boldsymbol{C}\boldsymbol{x}_t + \frac{\sqrt{1-\bar{\alpha}_t}}{\sqrt{\bar{\alpha}_t}}\boldsymbol{C}\boldsymbol{v}_t\right) \\ &= \mathcal{N}\left(-\frac{\sqrt{1-\bar{\alpha}_t}}{\sqrt{\bar{\alpha}_t}}\boldsymbol{C}\boldsymbol{v}_t + \frac{1}{\sqrt{\bar{\alpha}_t}}\boldsymbol{C}\boldsymbol{x}_t, \sigma_z^2\boldsymbol{I}\right), \end{aligned}$$

where the first line follows from (2) and (1), the second line follows from the assumption that at each time step t, the corresponding true noise $\boldsymbol{v}_t$ is known and the third line follows from the model assumption that the measurement noise follows $\mathcal{N}(\boldsymbol{0}, \boldsymbol{I})$. For the proposed method, by the assumption of $\frac{\sqrt{1-\bar{\alpha}_t}}{\sqrt{\bar{\alpha}_t}}\boldsymbol{C}\boldsymbol{v}_t = \boldsymbol{0}$ for low values of t, the approximate conditional distribution $p_a(\boldsymbol{y}|\boldsymbol{x}_t)$ can be written as

$$\begin{aligned} p_a(\boldsymbol{y}|\boldsymbol{x}_t) &= p(\boldsymbol{C}\boldsymbol{x}_0 + \boldsymbol{z} = \boldsymbol{y}|\sqrt{\bar{\alpha}_t}\boldsymbol{x}_0 + \sqrt{1-\bar{\alpha}_t}\boldsymbol{v}_t = \boldsymbol{x}_t) \\ &= p\left(\boldsymbol{z} = \boldsymbol{y} - \frac{1}{\sqrt{\bar{\alpha}_t}}\boldsymbol{C}\boldsymbol{x}_t\right) = \mathcal{N}\left(\frac{1}{\sqrt{\bar{\alpha}_t}}\boldsymbol{C}\boldsymbol{x}_t, \sigma_z^2\boldsymbol{I}\right). \end{aligned}$$

Using Lemma 1, we have

$$\text{KL}(p_t(\boldsymbol{y}|\boldsymbol{x}_t) \,\|\, p_a(\boldsymbol{y}|\boldsymbol{x}_t)) = \frac{1}{2\sigma_z^2}\frac{1-\bar{\alpha}_t}{\bar{\alpha}_t}\|\boldsymbol{C}\boldsymbol{v}_t\|_2^2. \tag{14}$$

□

This theorem provides insight into the performance of the proposed approximation. According to Theorem 1, the KL divergence between the approximate and true conditional distributions at each time step t is influenced by the parameter $\bar{\alpha}_t$ and the degradation matrix $\boldsymbol{C}$. When $\bar{\alpha}_t$ is close to one, the KL divergence becomes negligible, indicating that the approximation closely matches the true distribution. Furthermore, (14) highlights the role of the degradation matrix $\boldsymbol{C}$ in determining the quality of the approximation, as it directly affects the divergence.

Next, we find an expression for the KL divergence between the true $p_t(\boldsymbol{y}|\boldsymbol{x}_t)$ and an approximate $p_{a\prime}(\boldsymbol{y}|\boldsymbol{x}_t)$ for the conditional distribution, where the approximate distribution is calculated by estimating the added noise using a denoiser.

Theorem 2. Suppose that the value of the added noise for the latent diffusion variable at time step t is $\boldsymbol{v}_t \in \mathbb{R}^n$, i.e., $\boldsymbol{x}_t = \sqrt{1-\bar{\alpha}_t}\boldsymbol{x}_0 + \sqrt{\bar{\alpha}_t}\boldsymbol{v}_t$, and let $\hat{\boldsymbol{v}}_t = \boldsymbol{D}(\boldsymbol{x}_t, t)$ be the estimate of the true noise $\boldsymbol{v}_t$ calculated by the denoiser $\boldsymbol{D}$ from the value of $\boldsymbol{x}_t$ and time step t, then the KL divergence between the true conditional distribution $p_t(\boldsymbol{y}|\boldsymbol{x}_t)$ and the denoising approximation $p_{a\prime}(\boldsymbol{y}|\boldsymbol{x}_t)$ is $\frac{1}{2\sigma_z^2}\frac{1-\bar{\alpha}_t}{\bar{\alpha}_t}\|\boldsymbol{C}(\boldsymbol{v}_t - \hat{\boldsymbol{v}}_t)\|_2^2$.

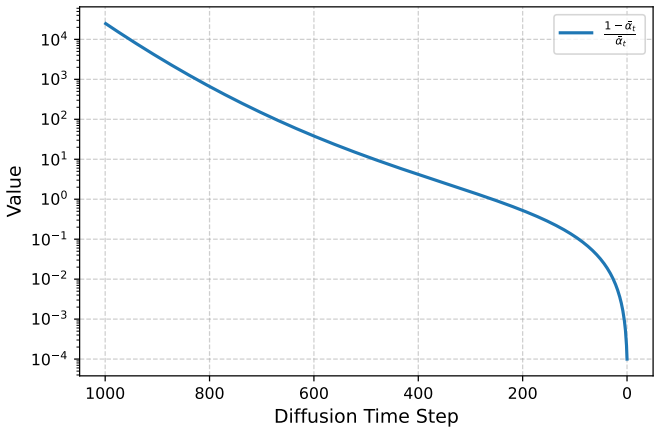


Fig. 5. Behavior of the coefficient $\frac{1-\bar{\alpha}_t}{\bar{\alpha}_t}$ across different diffusion time-step values, shown on a logarithmic scale. Consistent with Theorems 1 and 2, the coefficient decreases rapidly at lower diffusion time steps, supporting the validity of the approximation in this regime.

Proof. Similar to the proof of Theorem 1, for the true conditional distribution we have

$$p_t(\boldsymbol{y}|\boldsymbol{x}_t) = \mathcal{N}\left(-\frac{\sqrt{1-\bar{\alpha}_t}}{\sqrt{\bar{\alpha}_t}}\boldsymbol{C}\boldsymbol{v}_t + \frac{1}{\sqrt{\bar{\alpha}_t}}\boldsymbol{C}\boldsymbol{x}_t, \sigma_z^2\boldsymbol{I}\right). \tag{15}$$

For the approximated version $p_{a\prime}(\boldsymbol{y}|\boldsymbol{x}_t)$ we have

$$\begin{aligned} p_{a\prime}(\boldsymbol{y}|\boldsymbol{x}_t) &= p\left(\boldsymbol{z} = \boldsymbol{y} - \frac{1}{\sqrt{\bar{\alpha}_t}}\boldsymbol{C}\boldsymbol{x}_t + \frac{\sqrt{1-\bar{\alpha}_t}}{\sqrt{\bar{\alpha}_t}}\boldsymbol{C}\hat{\boldsymbol{v}}_t\right) \\ &= \mathcal{N}\left(-\frac{\sqrt{1-\bar{\alpha}_t}}{\sqrt{\bar{\alpha}_t}}\boldsymbol{C}\hat{\boldsymbol{v}}_t + \frac{1}{\sqrt{\bar{\alpha}_t}}\boldsymbol{C}\boldsymbol{x}_t, \sigma_z^2\boldsymbol{I}\right). \end{aligned} \tag{16}$$

Then, using Lemma 1, we have:

$$\text{KL}(p_t(\boldsymbol{y}|\boldsymbol{x}_t) \,\|\, p_{a\prime}(\boldsymbol{y}|\boldsymbol{x}_t)) = \frac{1}{2\sigma_z^2}\frac{1-\bar{\alpha}_t}{\bar{\alpha}_t}\|\boldsymbol{C}(\boldsymbol{v}_t - \hat{\boldsymbol{v}}_t)\|_2^2 \tag{17}$$

□

Theorem 2 shows that the KL divergence can be again small in low values of t, due to the $\bar{\alpha}_t \approx 1$ for those time steps. However, depending on the performance of the denoiser, it can offer benefits in the expense of more computations. More specifically, according to Theorem 2, $\|\boldsymbol{C}(\boldsymbol{v}_t - \hat{\boldsymbol{v}}_t)\|_2^2$ can help reduce the KL divergence between ground truth and approximate distributions, provided that the denoiser performs well in estimating $\boldsymbol{v}_t$, although at the cost of increased computational complexity associated with the denoising process.

V. Simulation Results and Analysis

We evaluate the effectiveness of the proposed method on two inverse problems: Image super-resolution and image inpainting. All simulations are conducted on 256×256 images in alignment with standard settings adopted in prior work e.g., [10] and [12]. For these simulations, a subset of 50 images is selected from the ImageNet dataset [35], a widely used dataset in image generation and restoration tasks. All the following simulations are done using an NVIDIA H200 GPU (140 GB) on a server with AMD EPYC 7742 CPU.

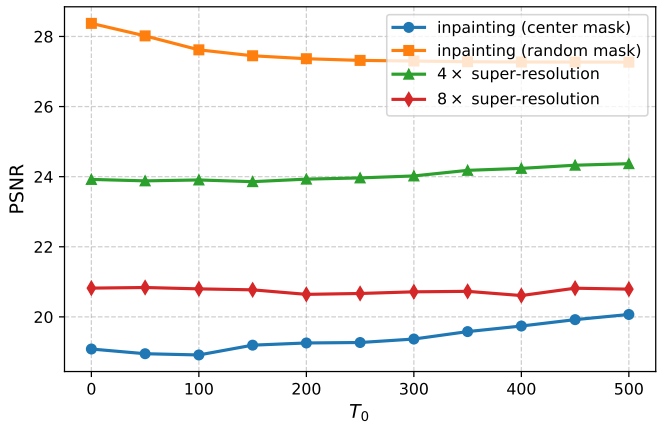


Fig. 6. Average PSNR scores across various inverse problems as a function of the guidance threshold T_0, using a class-conditional diffusion model. T_0 controls when each component of the piece-wise guidance term is active. ΠGDM corresponds to $T_0 = 0$.

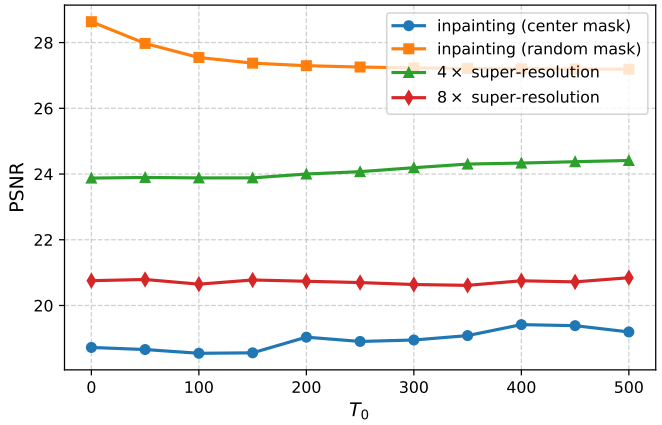


Fig. 7. Average PSNR scores across various inverse problems as a function of the guidance threshold T_0, using a class-unconditional diffusion model. T_0 controls when each component of the piece-wise guidance term is active. ΠGDM corresponds to $T_0 = 0$.

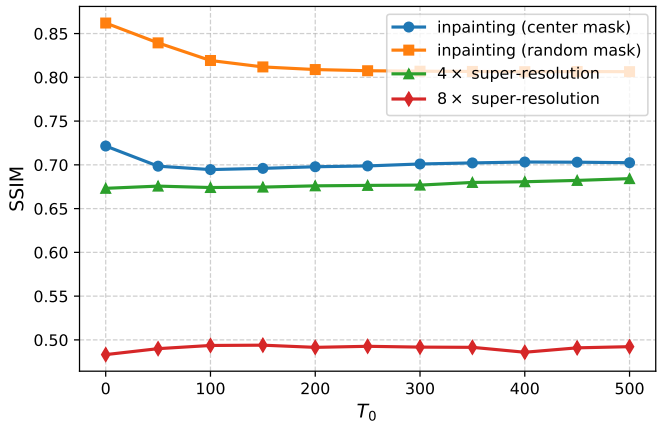


Fig. 8. Average SSIM scores across various inverse problems as a function of the guidance threshold T_0, using a class-conditional diffusion model. T_0 controls when each component of the piece-wise guidance term is active. ΠGDM corresponds to $T_0 = 0$.

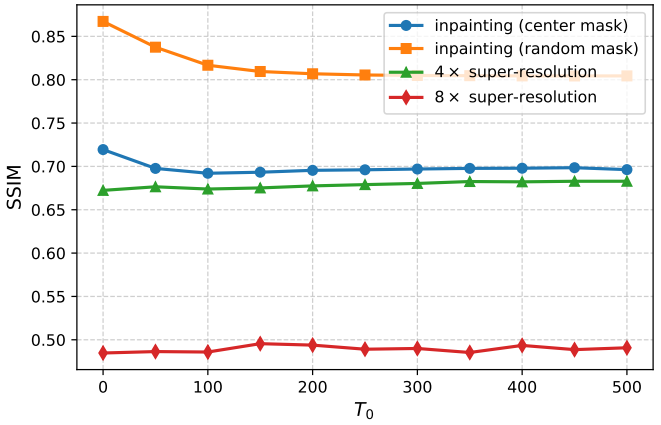


Fig. 9. Average SSIM scores across various inverse problems as a function of the guidance threshold T_0, using a class-unconditional diffusion model. T_0 controls when each component of the piece-wise guidance term is active. ΠGDM corresponds to $T_0 = 0$.

For the diffusion model, we rely on a publicly available pre-trained diffusion model[1] [1], trained specifically on 256×256 images. The model follows a standard denoising schedule with 1000 diffusion steps. Our method integrates this model into an inverse problem framework, and we compare its performance with ΠGDM [12], a recently proposed problem-agnostic approach that has demonstrated strong empirical performance across various inverse problems, often rivaling task-specific methods.

The simulations can be broadly categorized into two settings: class-conditional and class-unconditional. In the class-conditional case, it is assumed that the class label of the measurement image is known. A class-conditional diffusion model is employed, where the label is provided to the denoising process and incorporated during inference. In contrast, the class-unconditional case assumes that the class label is unknown at inference time; thus, the denoising process must operate solely based on the measurement without access to any class-specific information.

For the inpainting task, two masking schemes are considered to assess robustness under structured and unstructured missing data. The first is a *central block mask*, where a contiguous 128×128 square region at the image center is missing. The second is a *random mask*, in which 30% of the pixels are randomly removed throughout the image. In the super-resolution setting, the images are downsampled using average pooling with scaling factors 4 and 8, aiming to reconstruct the original high resolution images from their low resolution counterparts.

We evaluate performance using both pixel-level and perceptual metrics. Specifically, we report PSNR, SSIM, and learned perceptual image patch similarity (LPIPS). PSNR is a standard measure of reconstruction fidelity based on the pixel-wise error. SSIM, measures perceptual similarity by considering aspects such as luminance and contrast to follow how humans perceive an image, with values closer to one indicating better structural fidelity. LPIPS is also a perceptual metric and it

[1]https://github.com/openai/guided-diffusion/blob/main/README.md

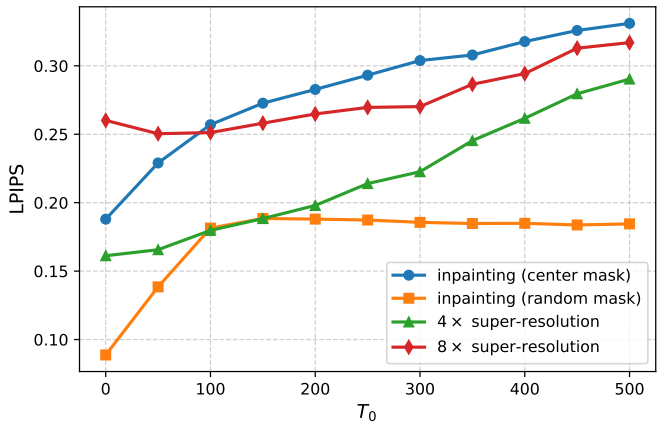


Fig. 10. Average LPIPS scores across various inverse problems as a function of the guidance threshold T_0, using a class-conditional diffusion model. T_0 controls when each component of the piece-wise guidance term is active. ΠGDM corresponds to $T_0 = 0$.

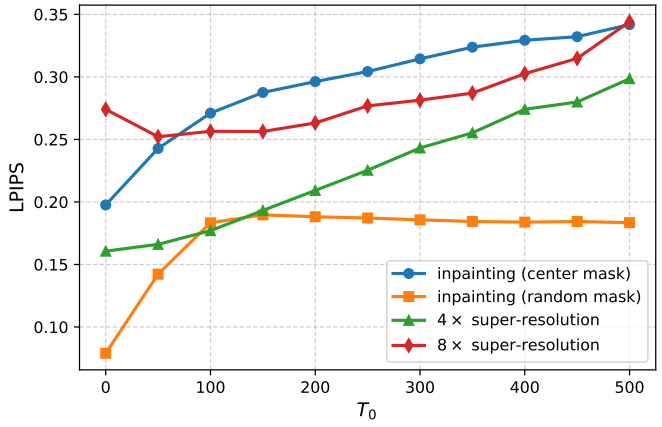


Fig. 11. Average LPIPS scores across various inverse problems as a function of the guidance threshold T_0, using a class-unconditional diffusion model. T_0 controls when each component of the piece-wise guidance term is active. ΠGDM corresponds to $T_0 = 0$.

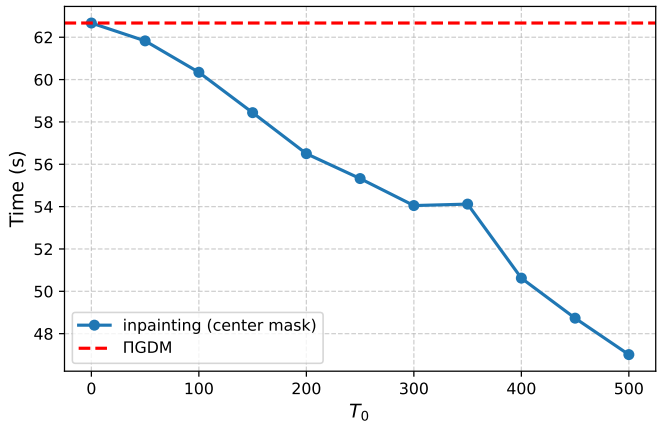


Fig. 12. Average inference time per image as a function of the guidance threshold T_0 for the inpainting task with a center mask, using a class-conditional diffusion model. T_0 controls when each component of the piece-wise guidance term is active. ΠGDM corresponds to $T_0 = 0$.

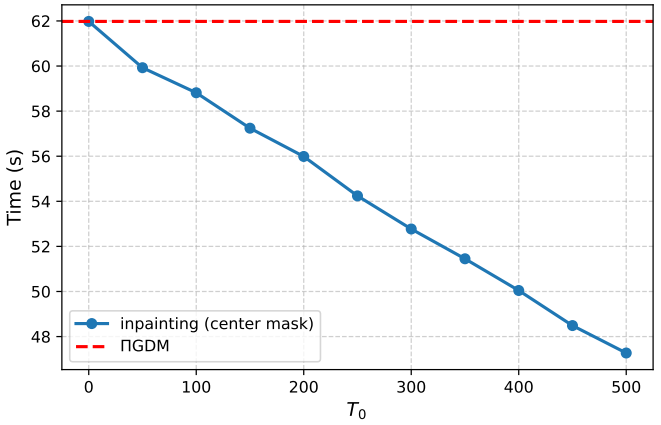


Fig. 13. Average inference time per image as a function of the guidance threshold T_0 for the inpainting task with a center mask, using a class-unconditional diffusion model. T_0 controls when each component of the piece-wise guidance term is active. ΠGDM corresponds to $T_0 = 0$.

leverages deep neural network features to evaluate perceptual similarity, where lower scores imply higher perceptual quality.

Figure 5 shows the behavior of the coefficient $\frac{1-\bar{\alpha}_t}{\bar{\alpha}_t}$. According to Theorems 1 and 2, the rapid decrease of this coefficient in lower diffusion time-steps reduces the KL divergence between true and the approximated problem-specific score functions.

Figures 1–4 present restored images across different values of T_0, illustrating the visual quality achieved by the proposed method. Compared to ΠGDM, the results show that high-quality reconstructions are maintained even as T_0 increases, which enables faster restoration by reducing the number of computationally expensive steps.

Figures 1 and 3 correspond to class-conditional models, while Figures 2 and 4 show class-unconditional results. As expected, class conditioning improves visual quality due to the added label information, but the proposed method performs robustly in both settings. These findings highlight the efficiency and adaptability of the proposed method across a range of different inverse problems and diffusion model configuration.

Figures 6 and 7 illustrate the PSNR metric across various inverse problems for different values of T_0. The reported values represent the average PSNR computed over all images in the dataset. The baseline comparison with ΠGDM corresponds to the data points at $T_0 = 0$. From these two figures, we observe that, increasing T_0 generally leads to a decline in pixel-level reconstruction quality, due to the reduced validity of the piecewise approximation at higher diffusion time steps. Notably, for super-resolution (both $4\times$ and $8\times$) and inpainting tasks (both center and random masks), the drop in PSNR remains negligible until $T_0 = 500$, indicating the robustness of the proposed method. A comparison between Figures 6 and 7 further reveals that slightly higher PSNR values are achieved when using class-conditional diffusion models, compared to their class-unconditional counterparts.

Figures 8 and 9 present the SSIM metric for the same set of inverse problems evaluated at varying values of T_0. Higher SSIM values indicate better perceptual quality. Similar to the

PSNR trends, an increase in T_0 generally leads to a decline in image quality. However, for inpainting and super-resolution tasks, this decline remains negligible for $T_0 \leq 500$, showing that computational savings can be realized without significantly compromising perceptual fidelity. Among the evaluated tasks, $8\times$ super-resolution exhibits the weakest performance, reflecting its increased difficulty due to the higher number of missing pixels. A comparison between Figures 8 and 9 again confirms that class-conditional diffusion models slightly outperform class-unconditional ones in terms of perceptual similarity.

Figures 10 and 11 evaluate the performance of the proposed approach in terms of the LPIPS metric for restoration results obtained using both class-conditional and class-unconditional diffusion models across various inverse problems and values of T_0. Unlike pixel-based metrics, LPIPS compares feature representations extracted by a neural network. In this simulation, AlexNet is used to assess perceptual similarity. Lower LPIPS values indicate higher perceptual quality. Consistent with previous observations, increasing T_0 leads to an increase in LPIPS, reflecting a degradation in image quality due to the diminished validity of the guidance term at higher time steps. However, the loss in performance remains minimal when T_0 is increased to 200, enabling substantial computational savings corresponding to a reduction of 200 diffusion steps. Among the tasks, $8\times$ super-resolution and inpainting with center mask exhibits the greatest performance decline, whereas inpainting with a random mask shows the least. Similar to the cases of PSNR and SSIM, the class-conditional diffusion model slightly outperforms its class-unconditional counterpart according to the LPIPS metric.

Figures 12–19 present the average inference time per image across various inverse problems and different values of T_0. Here, inference time refers to the duration required by the algorithm to restore a given image. In all these figures, the baseline inference time corresponds to the ΠGDM algorithm at $T_0 = 0$, represented by a dashed red line.

From Figures 12–19, we can observe that increasing the value of T_0 generally accelerates the algorithm. This is because a higher T_0 implies fewer diffusion steps requiring the derivative computation of the diffusion model, a computationally expensive operation involved in calculating the guidance term.

Figures 12 and 13 present the inference time per image for the inpainting problem with a center mask, comparing class-conditional and class-unconditional diffusion models, respectively. At $T_0 = 500$ where losses in PSNR and SSIM are negligible, the algorithm achieves speed improvements of 24% and 25% for the unconditional and conditional models, respectively. The difference in speedup arises because the class-conditional denoising model is more complex, making backpropagation for guidance computation more demanding. Consequently, reducing the number of these back-propagations yields greater computational savings.

Figures 14 and 15 illustrate inference times for the inpainting problem with random masks under various T_0 values for both model types. Consistent with previous results, increasing T_0 accelerates the algorithm. Notably, at $T_0 = 500$, where metric losses remain negligible, inference time is again reduced

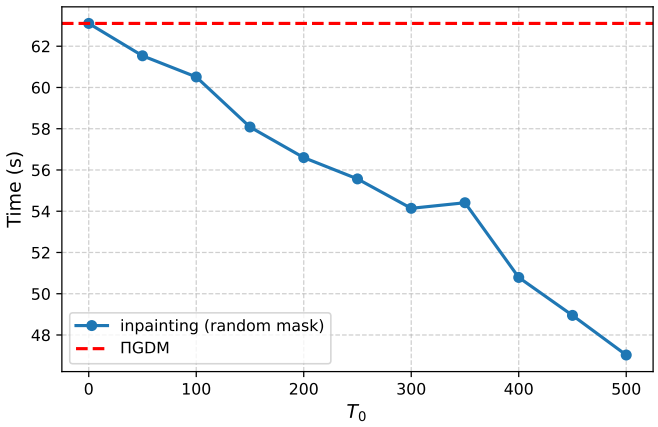


Fig. 14. Average inference time per image as a function of the guidance threshold T_0 for the inpainting task with a random mask, using a class-conditional diffusion model. T_0 controls when each component of the piecewise guidance term is active. ΠGDM corresponds to $T_0 = 0$.

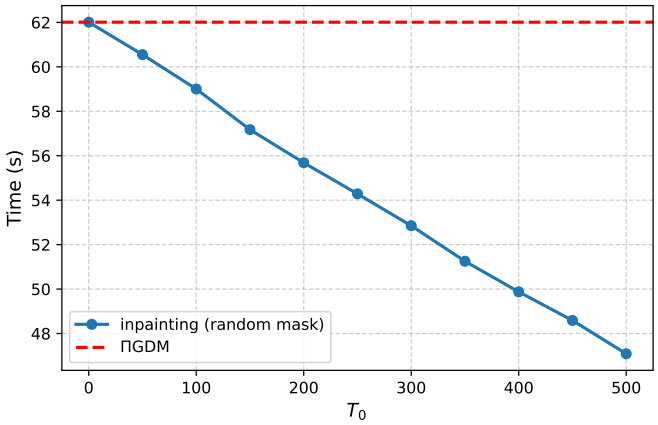


Fig. 15. Average inference time per image as a function of the guidance threshold T_0 for the inpainting task with a random mask, using a class-unconditional diffusion model. T_0 controls when each component of the piecewise guidance term is active. ΠGDM corresponds to $T_0 = 0$.

by 24% and 25% for unconditional and conditional models, respectively.

Figures 16 and 17 report inference times for the $4\times$super-resolution problem. Similar trends are observed, with higher T_0 values resulting in faster inference. At $T_0 = 500$, where metric degradation is minimal, speed improvements over ΠGDM amount to 21% for the unconditional model and 23% for the conditional model.

Finally, Figures 18 and 19 present inference times for the $8\times$ super-resolution task. From these two figures, we again observe that increasing T_0 decreases inference time. At $T_0 = 500$, the method is faster than ΠGDM by 23% and 24% for the unconditional and conditional diffusion models, respectively.

VI. Conclusion

In this paper, a novel framework for solving inverse problems by posterior sampling using diffusion models has been developed. We have proposed a piecewise function approximation for the guidance term based on the different levels

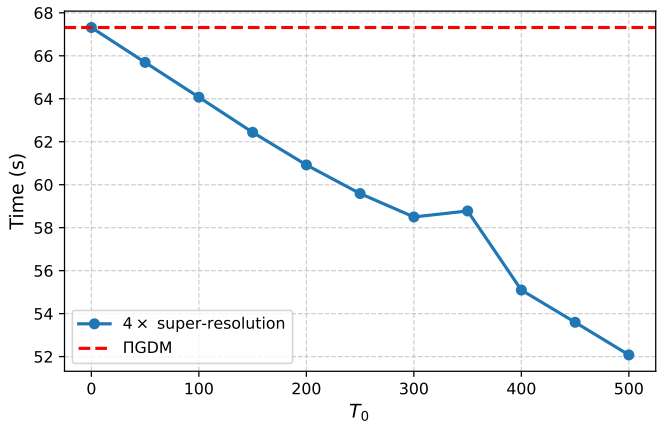


Fig. 16. Average inference time per image as a function of the guidance threshold T_0 for the $4\times$ super-resolution task, using a class-conditional diffusion model. T_0 controls when each component of the piece-wise guidance term is active. ΠGDM corresponds to $T_0 = 0$.

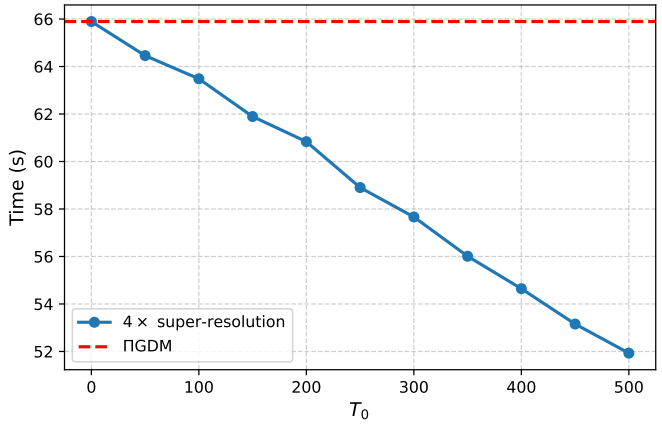


Fig. 17. Average inference time per image as a function of the guidance threshold T_0 for the $4\times$ super-resolution task, using a class-unconditional diffusion model. T_0 controls when each component of the piece-wise guidance term is active. ΠGDM corresponds to $T_0 = 0$.

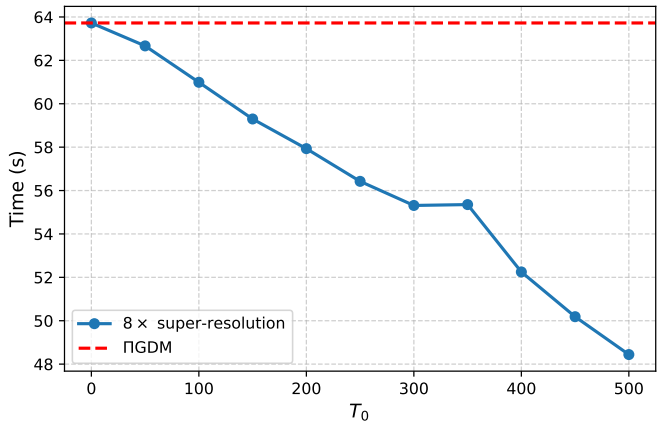


Fig. 18. Average inference time per image as a function of the guidance threshold T_0 for the $8\times$ super-resolution task, using a class-conditional diffusion model. T_0 controls when each component of the piece-wise guidance term is active. ΠGDM corresponds to $T_0 = 0$.

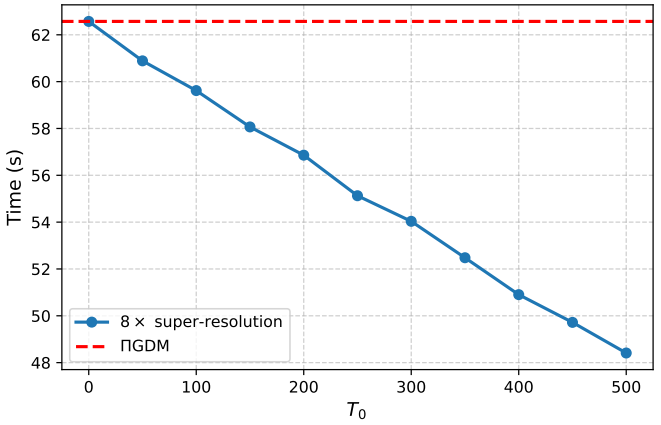


Fig. 19. Average inference time per image as a function of the guidance threshold T_0 for the $8\times$ super-resolution task, using a class-unconditional diffusion model. T_0 controls when each component of the piece-wise guidance term is active. ΠGDM corresponds to $T_0 = 0$.

of diffusion noise determined by the time step. We have shown that in low time steps, the computation of the guidance term can be simplified to the gradient of a Gaussian function because the diffusion noise is insignificant in that regime. Our simulation results showed that an inference time reduction of at least 23% is achievable without loss of PSNR and SSIM in the restored images compared to the baseline ΠGDM.

References

[1] P. Dhariwal and A. Nichol, "Diffusion Models Beat GANs on Image Synthesis," in *Advances in Neural Information Processing Systems*, vol. 34. Curran Associates, Inc., 2021, pp. 8780–8794. [Online]. Available: https://proceedings.neurips.cc/paper/2021/hash/49ad23d1ec9fa4bd8d77d02681df5cfa-Abstract.html

[2] D. Baranchuk, A. Voynov, I. Rubachev, V. Khrulkov, and A. Babenko, "Label-Efficient Semantic Segmentation with Diffusion Models," in *Proc. of International Conference on Learning Representations*, Oct. 2021. [Online]. Available: https://openreview.net/forum?id=SlxSY2UZQT

[3] T. Amit, T. Shaharbany, E. Nachmani, and L. Wolf, "SegDiff: Image Segmentation with Diffusion Probabilistic Models," Sep. 2022, arXiv:2112.00390 [cs]. [Online]. Available: http://arxiv.org/abs/2112.00390

[4] J. Austin, D. D. Johnson, J. Ho, D. Tarlow, and R. van den Berg, "Structured Denoising Diffusion Models in Discrete State-Spaces," in *Advances in Neural Information Processing Systems*, vol. 34. Curran Associates, Inc., 2021, pp. 17 981–17 993. [Online]. Available: https://proceedings.neurips.cc/paper/2021/hash/958c530554f78bcd8e97125b70e6973d-Abstract.html

[5] E. Hoogeboom, D. Nielsen, P. Jaini, P. Forré, and M. Welling, "Argmax Flows and Multinomial Diffusion: Learning Categorical Distributions," in *Advances in Neural Information Processing Systems*, vol. 34, 2021, pp. 12 454–12 465. [Online]. Available: https://proceedings.neurips.cc/paper/2021/hash/67d96d458abdef21792e6d8e590244e7-Abstract.html

[6] Z. Kong, W. Ping, J. Huang, K. Zhao, and B. Catanzaro, "DiffWave: A Versatile Diffusion Model for Audio Synthesis," Mar. 2021. [Online]. Available: http://arxiv.org/abs/2009.09761

[7] H. Chung, E. S. Lee, and J. C. Ye, "MR Image Denoising and Super-Resolution Using Regularized Reverse Diffusion," *IEEE Transactions on Medical Imaging*, vol. 42, no. 4, pp. 922–934, Apr. 2023. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/9941138

[8] C. Cao, Z.-X. Cui, Y. Wang, S. Liu, T. Chen, H. Zheng, D. Liang, and Y. Zhu, "High-Frequency Space Diffusion Model for Accelerated MRI," *IEEE Transactions on Medical Imaging*, vol. 43, no. 5, pp. 1853–1865, May 2024. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/10385176

[9] L. Yang, Z. Zhang, Y. Song, S. Hong, R. Xu, Y. Zhao, W. Zhang, B. Cui, and M.-H. Yang, "Diffusion Models: A Comprehensive Survey of Methods and Applications," *ACM Computing Surveys*, vol. 56, no. 4, pp. 1–39, Nov. 2023.

[10] B. Kawar, M. Elad, S. Ermon, and J. Song, "Denoising Diffusion Restoration Models," *Advances in Neural Information Processing Systems*, vol. 35, pp. 23 593–23 606, Dec. 2022. [Online]. Available: https://proceedings.neurips.cc/paper_files/paper/2022/hash/95504595b6169131b6ed6cd72eb05616-Abstract-Conference.html

[11] Y. Blau and T. Michaeli, "The Perception-Distortion Tradeoff," in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Jun. 2018, pp. 6228–6237. [Online]. Available: https://ieeexplore.ieee.org/document/8578750/authors

[12] J. Song, A. Vahdat, M. Mardani, and J. Kautz, "Pseudoinverse-Guided Diffusion Models for Inverse Problems," in *Proc. of International Conference on Learning Representations*, Sep. 2022. [Online]. Available: https://openreview.net/forum?id=9_gsMA8MRKQ

[13] Y. Song, L. Shen, L. Xing, and S. Ermon, "Solving Inverse Problems in Medical Imaging with Score-Based Generative Models," Jun. 2022, arXiv:2111.08005 [eess]. [Online]. Available: http://arxiv.org/abs/2111.08005

[14] G. Ongie, A. Jalal, C. A. Metzler, R. G. Baraniuk, A. G. Dimakis, and R. Willett, "Deep Learning Techniques for Inverse Problems in Imaging," *IEEE Journal on Selected Areas in Information Theory*, vol. 1, no. 1, pp. 39–56, May 2020. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/9084378

[15] S. V. Venkatakrishnan, C. A. Bouman, and B. Wohlberg, "Plug-and-Play priors for model based reconstruction," in *IEEE Global Conference on Signal and Information Processing*, pp. 945–948. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/6737048

[16] Y. Romano, M. Elad, and P. Milanfar, "The Little Engine That Could: Regularization by Denoising (RED)," *SIAM Journal on Imaging Sciences*, vol. 10, no. 4, pp. 1804–1844, Jan. 2017. [Online]. Available: https://epubs.siam.org/doi/abs/10.1137/16M1102884

[17] G. Mataev, M. Elad, and P. Milanfar, "DeepRED: Deep Image Prior Powered by RED," Oct. 2019, arXiv:1903.10176 [cs]. [Online]. Available: http://arxiv.org/abs/1903.10176

[18] A. Bora, A. Jalal, E. Price, and A. G. Dimakis, "Compressed Sensing using Generative Models," in *Proceedings of the 34th International Conference on Machine Learning*. PMLR, Jul. 2017, pp. 537–546. [Online]. Available: https://proceedings.mlr.press/v70/bora17a.html

[19] G. Daras, J. Dean, A. Jalal, and A. G. Dimakis, "Intermediate Layer Optimization for Inverse Problems using Deep Generative Models," Feb. 2021, arXiv:2102.07364 [cs]. [Online]. Available: http://arxiv.org/abs/2102.07364

[20] S. Menon, A. Damian, S. Hu, N. Ravi, and C. Rudin, "PULSE: Self-Supervised Photo Upsampling via Latent Space Exploration of Generative Models," 2020, pp. 2437–2445. [Online]. Available: https://openaccess.thecvf.com/content_CVPR_2020/html/Menon_PULSE_Self-Supervised_Photo_Upsampling_via_Latent_Space_Exploration_of_Generative_CVPR_2020_paper.html

[21] Y. Sun, B. Wohlberg, and U. S. Kamilov, "An Online Plug-and-Play Algorithm for Regularized Image Reconstruction," *IEEE Transactions on Computational Imaging*, vol. 5, no. 3, pp. 395–408, Sep. 2019. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/8616843

[22] Z. Kadkhodaie and E. Simoncelli, "Stochastic Solutions for Linear Inverse Problems using the Prior Implicit in a Denoiser," in *Advances in Neural Information Processing Systems*, vol. 34, 2021, pp. 13 242–13 254. [Online]. Available: https://proceedings.neurips.cc/paper/2021/hash/6e28943943dbed3c7f82fc05f269947a-Abstract.html

[23] A. Jalal, S. Karmalkar, A. G. Dimakis, and E. Price, "Instance-Optimal Compressed Sensing via Posterior Sampling," Jun. 2021, arXiv:2106.11438 [cs]. [Online]. Available: http://arxiv.org/abs/2106.11438

[24] B. Kawar, G. Vaksman, and M. Elad, "SNIPS: Solving Noisy Inverse Problems Stochastically," in *Advances in Neural Information Processing Systems*, vol. 34, 2021, pp. 21 757–21 769. [Online]. Available: https://proceedings.neurips.cc/paper_files/paper/2021/hash/b5c01503041b70d41d80e3dbe31bbd8c-Abstract.html

[25] ——, "Stochastic Image Denoising by Sampling From the Posterior Distribution," 2021, pp. 1866–1875. [Online]. Available: https://openaccess.thecvf.com/content/ICCV2021W/AIM/html/Kawar_Stochastic_Image_Denoising_by_Sampling_From_the_Posterior_Distribution_ICCVW_2021_paper.html

[26] H. Chung, B. Sim, and J. C. Ye, "Come-Closer-Diffuse-Faster: Accelerating Conditional Diffusion Models for Inverse Problems through Stochastic Contraction," in *Proc. of IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2022, pp. 12 403–12 412. [Online]. Available: https://ieeexplore.ieee.org/document/9879691/

[27] C. Saharia, W. Chan, H. Chang, C. Lee, J. Ho, T. Salimans, D. Fleet, and M. Norouzi, "Palette: Image-to-Image Diffusion Models," in *ACM SIGGRAPH Conference Proceedings*. Association for Computing Machinery, Jul. 2022, pp. 1–10. [Online]. Available: https://dl.acm.org/doi/10.1145/3528233.3530757

[28] C. Saharia, J. Ho, W. Chan, T. Salimans, D. J. Fleet, and M. Norouzi, "Image Super-Resolution via Iterative Refinement," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 45, no. 4, pp. 4713–4726, Apr. 2023. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/9887996

[29] M. Bertero, P. Boccacci, and C. D. Mol, *Introduction to Inverse Problems in Imaging*, 2nd ed. Boca Raton: CRC Press, Dec. 2021.

[30] A. Mohammad-Djafari, N. Chu, L. Wang, and L. Yu, "Deep Learning and Inverse Problems," Sep. 2023, arXiv:2309.00802 [cs]. [Online]. Available: http://arxiv.org/abs/2309.00802

[31] I. Craig and J. Brown, "Inverse problems in astronomy," Jan. 1986, publisher: Adam Hilger Ltd.,Accord, MA. [Online]. Available: https://www.osti.gov/biblio/5734250

[32] D. Calvetti and E. Somersalo, "Inverse problems: From regularization to Bayesian inference," *WIREs Computational Statistics*, vol. 10, no. 3, p. e1427, 2018.

[33] B. D. O. Anderson, "Reverse-time diffusion equation models," *Stochastic Processes and their Applications*, vol. 12, no. 3, pp. 313–326, May 1982. [Online]. Available: https://www.sciencedirect.com/science/article/pii/0304414982900515

[34] J. R. Hershey and P. A. Olsen, "Approximating the Kullback Leibler Divergence Between Gaussian Mixture Models," in *Proc. of IEEE International Conference on Acoustics, Speech and Signal Processing*, vol. 4, Apr. 2007, pp. IV–317–IV–320. [Online]. Available: https://ieeexplore.ieee.org/document/4218101

[35] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei, "ImageNet Large Scale Visual Recognition Challenge," *International Journal of Computer Vision*, vol. 115, no. 3, pp. 211–252, Dec. 2015. [Online]. Available: https://doi.org/10.1007/s11263-015-0816-y