Blind Inverse Problem Solving Made Easy by Text-to-Image Latent Diffusion

Anonymous Author(s)
Affiliation
Address
email

Abstract

This paper considers blind inverse image restoration, the task of predicting a target image from a degraded source when the degradation (i.e. the forward operator) is unknown. Existing solutions typically rely on restrictive assumptions such as operator linearity, curated training data or narrow image distributions limiting their practicality. We introduce LADiBI, a training-free method leveraging large-scale text-to-image diffusion to solve diverse blind inverse problems with minimal assumptions. Within a Bayesian framework, LADiBI uses text prompts to jointly encode priors for both target images and operators, unlocking unprecedented flexibility compared to existing methods. Additionally, we propose a novel diffusion posterior sampling algorithm that combines strategic operator initialization with iterative refinement of image and operator parameters, eliminating the need for highly constrained operator forms. Experiments show that LADiBI effectively handles both linear and challenging nonlinear image restoration problems across various image distributions, all without task-specific assumptions or retraining.

1 Introduction

Image restoration is a critical problem in many fields such as medical imaging and computational photography, as it addresses real-world challenges including image decompression, deblurring, and super-resolution [Yuan et al., 2007, Greenspan, 2009, Isaac and Kulkarni, 2015]. These restoration tasks can be formulated as *inverse problems*, where the goal is to recover unknown image data $\boldsymbol{x}$ from observed measurements $\boldsymbol{y}$. Formally, these problems can be expressed as $\boldsymbol{y} = \mathcal{A}_\phi(\boldsymbol{x}) + \boldsymbol{n}$, where $\mathcal{A}$ is an operator representing the forward degradation process parametrized by ϕ, and $\boldsymbol{n}$ is the measurement noise. Being widely applicable, this problem has attracted numerous solutions, ranging from methods with handcrafted inductive biases to deep learning, especially diffusion-based techniques [Kawar et al., 2022, Song et al., 2022, Bansal et al., 2023, Yu et al., 2023, He et al., 2024].

However, most existing research focuses on the settings where the operator $\mathcal{A}_\phi$ is *known*. In practice, the operator is often *unknown*, leading to what are termed *blind inverse problems* that present significant challenges due to their ill-posed nature. Current methods attempt to address this problem through several restrictive strategies: (1) introducing hand-crafted inductive biases via explicit formulas or specialized neural architectures [Pan et al., 2018, 2017, Ren et al., 2020], (2) simplifying operators with linear assumptions [Chung et al., 2023a, Murata et al., 2023], (3) constraining target distributions to narrow image classes [Chihaoui et al., 2024, Laroche et al., 2024], or (4) training task-specific models on curated measurement, operator or image datasets [Zamir et al., 2021, Kupyn et al., 2019]. While effective in specific scenarios, these approaches suffer from limited flexibility and significant deployment barriers due to costly training and laborious hyperparameter tuning. This raises a fundamental question: can we develop a more generalizable algorithm capable of handling diverse degradation operators and image distributions without additional training or data collection?

Submitted to 39th Conference on Neural Information Processing Systems (NeurIPS 2025). Do not distribute.

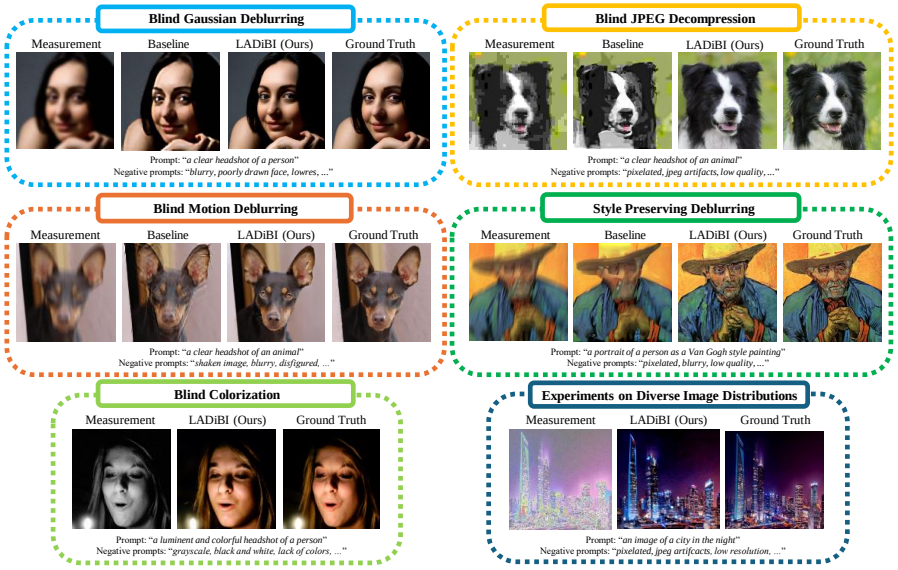


Figure 1: Our proposed **LADiBI** is a training-free blind inverse problem solving algorithm for image restoration using large pre-trained text-to-image diffusion models. LADiBI is applicable to a wide variety of image distribution as well as operators with minimal modeling assumptions imposed.

To tackle this challenge, we reframe the problem via Bayesian inference, where we sample from the posterior $p(\boldsymbol{x}, \mathcal{A}_\phi|\boldsymbol{y}) \propto p(\boldsymbol{y}|\boldsymbol{x}, \mathcal{A}_\phi)p(\boldsymbol{x}, \mathcal{A}_\phi)$. This formulation naturally decomposes the seemingly intractable problem into two simpler sub-problems: estimating the prior $p(\boldsymbol{x}, \mathcal{A}_\phi)$ and sampling according to the measurement likelihood $p(\boldsymbol{y}|\boldsymbol{x}, \mathcal{A}_\phi)$. This decomposition is particularly well-suited for diffusion-based frameworks, where we can leverage pre-trained models as the prior and guide the sampling process via measurement constraints. However, while pre-trained models typically exist for image, adequate priors or training data for the operators are generally unavailable. Thus, existing methods further factorize the joint prior into independent distributions and impose simplifying assumptions such as linearity on the operator, which severely restrict their generality and flexibility.

Instead, our approach is motivated by the key observation that, common restoration tasks can be intuitively described through natural language (e.g., "high-definition, clear image" for targets, "blurry, low-quality" for measurements). Moreover, large pre-trained text-to-image diffusion models already encapsulate rich distributions of both targets and measurements. Although this can, in some way, seem to render the approach less of a true "blind" solution, in practice we note that *all* methods for blind inverse problems require *some* assumptions over the space of transformations, and using English text to encode the joint is an extremely flexible and easy-to-use mechanism. Based on these insights, we propose a simple yet powerful method: using classifier-free guidance [Ho and Salimans, 2021], we can approximate the joint prior's score across diverse images and operators using a single pre-trained text-to-image diffusion model, dramatically broadening the flexibility of existing frameworks. Our approach is particularly advantageous in blind settings, since the appropriate prompts can often be easily inferred directly from the measurements, and textual descriptions enable us to co-encode unknown degradations and desired outputs without training or handcrafted priors.

In addition to estimating the prior score, effectively sampling from the posterior distribution is also crucial for ensuring that restored images satisfy measurement constraints. In blind inverse problems, achieving this requires accurately estimating both data and operator parameters, ideally with generalized parameter classes such as neural networks to improve flexibility. However, reliably initializing these highly unconstrained operator parameters can be challenging. To address this, we propose a novel co-optimization diffusion posterior sampling algorithm specifically tailored for blind inverse problems. Our method begins with a new initialization scheme that leverages pseudo-supervision signals derived from multiple lower-quality target image approximations generated by fast posterior diffusion sampling. We then iteratively refine both operator parameters and data estimates through an alternating optimization procedure integrated within the diffusion sampling process. This strategy eliminates restrictive assumptions about operator forms, thereby enabling nonlinear blind inverse problem solving with highly flexible operator parametrizations.

Table 1: A conceptual comparison between our proposed LADiBI and the existing literature.

Method Family	Method	Prior Type	Diverse Image Prior	Training-free	Flexible Operator
Optimization-based	Pan-ℓ_0 [Pan et al., 2017]	Inductive bias	×	✓	×
	Pan-DCP [Pan et al., 2018]	Inductive bias	×	✓	×
Self-supervised	SelfDeblur [Ren et al., 2020]	Inductive bias	×	✓	×
Supervised	MPRNet [Zamir et al., 2021]	Discriminative	×	×	✓
	DeblurGANv2 [Kupyn et al., 2019]	GAN	×	×	✓
Diffusion-based	BlindDPS [Chung et al., 2023a]	Pixel diffusion	×	×	×
	BIRD [Chihaoui et al., 2024]	Pixel diffusion	×	✓	×
	GibbsDDRM [Murata et al., 2023]	Pixel diffusion	×	✓	×
	LADiBI (Ours)	Text-to-image latent diffusion	✓	✓	✓

Combining the text-conditioned prior with our effective posterior sampling, we introduce **L**anguage-**A**ssisted **Di**ffusion for **B**lind **I**nverse problems (**LADiBI**), a training-free method that leverages large-scale text-to-image diffusion models to solve a broad range of blind image restoration problems with minimal assumptions. LADiBI can be directly applied across diverse data distributions and allows for easy specification of task-specific assumptions through simple prompting. Algorithm 1 and Figure 2 provide an overview of LADiBI, which can be easily adapted from the standard inference algorithm used in popular text-to-image diffusion models. Unlike existing methods, LADiBI requires no model retraining or reselection for different data distributions or operator functions. Instead, all prior parameterization is encoded directly in the prompt, which users can adjust as needed. Notably, we do not assume linearity of the operator, making LADiBI, to the best of our knowledge, the most generalizable approach to blind inverse problem solving in image restoration.

We evaluate LADiBI against state-of-the-art baselines on a range blind image restoration tasks, including linear problem (e.g. motion and Gaussian deblurring) and nonlinear problem(e.g. JPEG decompression), across various image distributions, as illustrated in Figure 1. In the linear setting, our method matches the performance of the state-of-the-art approaches while requiring significantly fewer assumptions. In the nonlinear setting, LADiBI is the only method tested that can successfully perform JPEG decompression without any prior information of the task (such as the compression algorithm, quantization table or quantization factors), relying solely on observations of the compressed images.

2 Background & Related Works

Diffusion for Inverse Problem Solving Diffusion models [Song et al., 2021b, Ho et al., 2020] generate clean data samples $\boldsymbol{x}_0$ by iteratively refining noisy samples $\boldsymbol{x}_t$ using a time-dependent score function $\nabla_{\boldsymbol{x}_t} \log p_t(\boldsymbol{x}_t)$. This score function is usually parametrized as a noise predictor $\epsilon_\theta(\boldsymbol{x}_t, t)$ and can be used to produce the clean data samples through iteratively denoising. Particularly, a popular sampling algorithm DDIM [Song et al., 2021a] adopts the update rule

$$\boldsymbol{x}_{t-1} = \sqrt{\bar{\alpha}_{t-1}} \underbrace{\left(\frac{\boldsymbol{x}_t - \sqrt{1-\bar{\alpha}_t}\epsilon_\theta(\boldsymbol{x}_t, t)}{\sqrt{\bar{\alpha}_t}} \right)}_{\text{intermediate estimation of } \boldsymbol{x}_0, \text{ denoted as } \boldsymbol{x}_{0|t}} + \sqrt{1-\bar{\alpha}_{t-1}-\sigma_t^2}\epsilon_\theta(\boldsymbol{x}_t, t) + \sigma_t \epsilon \tag{1}$$

that consists of an intermediate estimation of the clean data in order to perform fast sampling.

Many efforts attempt to use unconditionally pretrained diffusion for conditional generation [Song et al., 2021b, Meng et al., 2022, Dhariwal and Nichol, 2021], especially inverse problem solving [Chung et al., 2023b, Kawar et al., 2022, Song et al., 2022], without additional training. Generally, when sampling from $p(\boldsymbol{x}|\boldsymbol{y})$, they decompose its score as

$$\nabla_{\boldsymbol{x}_t} \log p_t(\boldsymbol{x}_t|\boldsymbol{y}) = \nabla_{\boldsymbol{x}_t} \log p_t(\boldsymbol{x}_t) + \nabla_{\boldsymbol{x}_t} \log p_t(\boldsymbol{y}|\boldsymbol{x}_t) \tag{2}$$

Since $\nabla_{\boldsymbol{x}_t} \log p_t(\boldsymbol{x}_t)$ can be obtained from an unconditionally pre-trained diffusion model, these methods usually aim at deriving an accurate approximation for $\nabla_{\boldsymbol{x}_t} \log p_t(\boldsymbol{y}|\boldsymbol{x}_t)$.

Recent work has also explored text-to-image latent diffusion models for inverse problem-solving [Saharia et al., 2022, Balaji et al., 2023, Zhang et al., 2024]. In particular, MPGD [He et al., 2024] addresses inverse problems by leveraging the manifold preserving property of the latent diffusion models. Specifically, it modifies the intermediate clean latent estimate $\boldsymbol{z}_{0|t}$ with

$$\boldsymbol{z}_{0|t} = \boldsymbol{z}_{0|t} - c_t \nabla_{\boldsymbol{z}_{0|t}} \|\boldsymbol{y} - \mathcal{A}_\phi(\mathbf{D}(\boldsymbol{z}_{0|t}))\|_2^2 \tag{3}$$

where $\mathbf{D}(\boldsymbol{z}_{0|t})$ represents the decoded intermediate clean image estimation and c_t is the step size hyperparameter. The L_2 loss $\|\boldsymbol{y} - \mathcal{A}_\phi(\mathbf{D}(\boldsymbol{z}_{0|t}))\|_2^2$ is induced by the additive Gaussian noise assumption from conventional inverse problem setting. Although these diffusion-based methods perform well on diverse data distributions and tasks, they all require the operator $\mathcal{A}_\phi$ to be known.

Blind Inverse Problem Blind inverse problems aim to recover unknown data $\boldsymbol{x} \in \mathbb{R}^d$ from measurements $\boldsymbol{y} \in \mathbb{R}^m$, typically modeled as:

$$\boldsymbol{y} = \mathcal{A}_\phi(\boldsymbol{x}) + \boldsymbol{n} \tag{4}$$

where $\mathcal{A}_\phi : \mathbb{R}^d \to \mathbb{R}^m$ is the forward degradation operator parameterized by unknown function ϕ, and $\boldsymbol{n} \sim \mathcal{N}(0, \sigma^2 I) \in \mathbb{R}^m$ represents measurement noise with variance $\sigma^2 I$. Blind inverse problems are more difficult due to the joint estimation of $\boldsymbol{x}$ and $\mathcal{A}_\phi$, and are inherently ill-posed without further assumptions. As a result, existing methods typically incorporate different assumptions about the priors of the target image data as well as the unknown operator.

Conventional methods use hand-crafted functions as image and operator prior constraints [Chan and Wong, 1998, Pan et al., 2018, 2017, Krishnan et al., 2011, Levin et al., 2009]. They often obtain these functional constraints by observing certain characteristics (e.g. clear edges and sparsity) unique to distributions that are usually considered as the target image (e.g. high definition natural images) and the operator (e.g. blurring kernels). However, not only are these hand-made functions not generalizable, they also often require significant manual tuning for each individual image.

With the rise of deep learning, neural network has become a popular choice for parameterizing priors [Ulyanov et al., 2018, Gandelsman et al., 2019, Ren et al., 2020, Kupyn et al., 2019, Zamir et al., 2021]. These methods offer significant improvement over traditional approaches. However, their learning procedures require separate data collection and model training for each image distribution and task, making them resource-intensive.

Recently, inspired by advances in diffusion models for inverse problem solving, numerous efforts have incorporated the priors from pre-trained diffusion models. However, most of these methods still lack generalizability [Chung et al., 2023a, Murata et al., 2023, Sanghvi et al., 2023, Laroche et al., 2024, Tu et al., 2024, WeiminBai et al., 2025]. For instance, Chung et al. [2023a], Sanghvi et al. [2023] require training separate operator priors, while Murata et al. [2023] remains training-free but rely on the operator kernel's SVD for feasible optimization. These approaches generally depend on linear assumptions about the operator and well-trained diffusion models tailored to specific image distributions, limiting their ability to generalize across diverse image and operator types.

Table 1 summarizes the conceptual difference between our method and popular approaches in current literature. By leveraging large-scale pre-trained text-to-image latent diffusion models and our new posterior sampling algorithm, our method offers the most generalizability across diverse image and operator distributions with no additional training.

3 Method

In this paper, we aim to tackle the problem of blind inverse problem solving defined in Equation 4. Our solution has the following desiderata: (1) **No additional training:** it should not require data collection or model training, (2) **Adaptability to diverse image priors:** the same model should apply to various image distributions, (3) **Flexible operators:** no assumptions about the operator's functional form, such as linearity or task-specific update rules, should be necessary. To make this problem feasible, we assume access to open-sourced pre-trained models.

To tackle this problem, we first formulate it as a Bayesian inference problem where the optimal solution is to sample from the posterior

$$p(\boldsymbol{x}, \mathcal{A}_\phi | y) \propto p(\boldsymbol{y} | \boldsymbol{x}, \mathcal{A}_\phi) p(\boldsymbol{x}, \mathcal{A}_\phi). \tag{5}$$

This formulation allows us to decompose this problem into two parts: obtaining sampling to maximizes the measurement likelihood $p(\boldsymbol{y} | \boldsymbol{x}, \mathcal{A}_\phi)$. This makes diffusion-based framework the ideal approach as its posterior sampling process naturally separates these two stages.

3.1 Obtaining the prior score

As established in Equation 2, diffusion-based approaches to inverse problems require access to the score of the prior distribution. For blind inverse problems, this extends to estimating the score of the

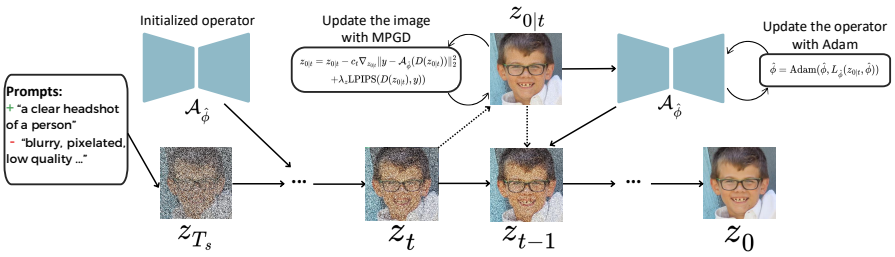


Figure 2: A schematic overview of LADiBI (Algorithm 1).

joint distribution $\nabla_{\boldsymbol{x}_t} \log p(\boldsymbol{x}_t, \mathcal{A}_\phi)$, which presents significant challenges beyond standard image priors. While strong pre-trained diffusion models usually exist for target image distributions $p(\boldsymbol{x})$, appropriate models or even training data for degradation operators $\mathcal{A}_\phi$ are largely unavailable, making estimations of the joint distribution $p(\boldsymbol{x}, \mathcal{A}_\phi)$ particularly difficult.

Therefore, most existing diffusion-based methods further decompose $p(\boldsymbol{x}, \mathcal{A}_\phi) \approx p(\boldsymbol{x})p(\mathcal{A}_\phi)$ and impose restrictive assumptions, such as linearity constraints, on the operator. To estimate the operator distribution, they either rely on simulated training data or perform constrained optimization under fixed operator distributions. In addition, these methods typically require different pre-trained diffusion models for different image domains (e.g., deblurring human v.s. animal faces necessitate different prior models). This reliance on domain-specific models and strong operator assumptions fundamentally limits their generalizability, particularly violating the second and third desiderata.

Motivated by these limitations, we introduce an alternative approach inspired by the following key insights. First, many image restoration tasks can be intuitively described using natural language. In addition, large pre-trained text-to-image diffusion models like Stable Diffusion [Rombach et al., 2021] already capture rich distributions of both target images and common degradation artifacts that can be desribed by text. Based on these insights, we propose to approximate the joint prior's score using large-scale pre-trained text-to-image diffusion models through classifier-free guidance [Ho and Salimans, 2021]. In particular, we encode the desired target image characteristics using the positive prompts, and the unwanted degradation artifacts via negative prompts. For example, when restoring a Gaussian blurred human face image, a suitable positive prompt can be "a clear headshot of a person", and the corresponding negative prompt could be "blurry, low-quality image".

Formally, denoting the positive prompt as $\boldsymbol{c}_+$ and the negative prompt as $\boldsymbol{c}_-$,

$$\nabla_{\boldsymbol{x}_t} \log p(\boldsymbol{x}_t, \mathcal{A}_\phi) \approx \nabla_{\boldsymbol{x}_t} \log p(\boldsymbol{x}_t|\boldsymbol{c}_-) + \gamma(\nabla_{\boldsymbol{x}_t} \log p(\boldsymbol{x}_t|\boldsymbol{c}_+) - \nabla_{\boldsymbol{x}_t} \log p(\boldsymbol{x}_t|\boldsymbol{c}_-)) \tag{6}$$

where $\gamma > 1$ is a weighting hyperparameter. When using latent diffusion models parameterized by θ, this translates to the empirical noise prediction

$$\hat{\epsilon}_\theta(\boldsymbol{z}_t, t) = \epsilon_\theta(\boldsymbol{z}_t, t, \boldsymbol{c}_-) + \gamma(\epsilon_\theta(\boldsymbol{z}_t, t, \boldsymbol{c}_+) - \epsilon_\theta(\boldsymbol{z}_t, t, \boldsymbol{c}_-)) \tag{7}$$

This straightforward approximation provides access to the otherwise intractable joint prior score, and enables versatile applications across diverse image and operator distributions. By leveraging the knowledge encoded in large pre-trained text-to-image models, our method bypasses the need for task-specific training, operator restriction or model re-selection.

3.2 Sampling from the posterior

In addition to a strong prior score, our output image should also satisfy the measurement constraint. Given the problem setup in Equation 4, the measurements are subject to additive Gaussian noise $\boldsymbol{n}$, hence $\log p(\boldsymbol{y}|\boldsymbol{x}, \mathcal{A}_\phi) = -\frac{1}{2\sigma^2}\|\boldsymbol{y} - \mathcal{A}_\phi(\boldsymbol{x})\|_2^2$. When $\mathcal{A}_\phi$ is known, we can use the MPGD update rule in Equation 3 for posterior sampling.

However, since $\mathcal{A}_\phi$ is unknown, the true parameters ϕ is often approximated by another set of parameters $\hat{\phi}$. This approximation is usually addressed by one of the two strategies: an alternating optimization scheme that jointly approximates $\boldsymbol{x}$ and ϕ, or obtaining a reliable $\hat{\phi}$ first then solving a non-blind inverse problem. The first approach is well-suited for training-free settings, but it is often highly sensitive to $\hat{\phi}$ initialization and tuning. The second approach can perform well if a strong $\hat{\phi}$ is obtained, though it usually requires training and additional restrictions. We propose a hybrid strategy: we first obtain a reliable initial $\hat{\phi}$, and then perform an alternating optimization to iteratively refine both the operator parameter and data estimations throughout the diffusion process.

Algorithm 1 Our algorithm LADiBI

/* Initialize latent with encoded measurement $\boldsymbol{y}$ & SDEdit */
$\boldsymbol{z}_{T_s} = \sqrt{\bar{\alpha}_{T_s}}\mathbf{E}(y) + \sqrt{1-\bar{\alpha}_{T_s}}\epsilon_{T_s}$ for $\epsilon_{T_s} \sim \mathcal{N}(0, I)$
Initialize $\hat{\phi}$ with a fixed operator prior or Algorithm 2
for $t = T_s, \ldots, 1$ **do**
  /* Use time-traveling for more stable results */
  for $j = 1, \ldots, M$ **do**
    /* Use CFG to obtain accurate prior */
    Calculate $\hat{\epsilon}_\theta(\boldsymbol{z}_t, t)$ with Eq. 7
    $\boldsymbol{z}_{0|t} = \frac{1}{\sqrt{\bar{\alpha}_t}}(\boldsymbol{z}_t - \sqrt{1-\bar{\alpha}_t}\hat{\epsilon}_\theta(\boldsymbol{z}_t, t))$
    if $j \equiv 0 \pmod 2$ **then**
      /* Perform MPGD with the estimated $\mathcal{A}_{\hat{\phi}}$ */
      Update $\boldsymbol{z}_{0|t}$ with Eq. 8
    end if
    $\epsilon_t, \epsilon'_t \sim \mathcal{N}(0, I)$
    $\boldsymbol{z}_{t-1} = \sqrt{\bar{\alpha}_{t-1}}\boldsymbol{z}_{0|t} + \sqrt{1-\bar{\alpha}_{t-1}-\sigma_t^2}\epsilon_\theta(\boldsymbol{z}_t, t) + \sigma_t\epsilon_t$
    $\boldsymbol{z}_t = \sqrt{\frac{\bar{\alpha}_t}{\bar{\alpha}_{t-1}}}\boldsymbol{z}_{t-1} + \sqrt{1-\frac{\bar{\alpha}_t}{\bar{\alpha}_{t-1}}}\epsilon'_t$
  end for
  /* Periodically update $\hat{\phi}$ with $\boldsymbol{z}_{0|t}$ and Eq. 9 */
  if $t \equiv 0 \pmod 5$ **then**
    for $k = 1, \ldots, K$ **do**
      $\hat{\phi} = \text{Adam}(\hat{\phi}, L_\phi(\boldsymbol{z}_{0|t}, \hat{\phi}))$
    end for
  end if
end for
return $\boldsymbol{x}_0 = \mathbf{D}(\boldsymbol{z}_0)$

Algorithm 2 General Operator Initialization

for $j = 1, \ldots, M$ **do**
  /* Initialize latent batch with encoded $\boldsymbol{y}$ & SDEdit */
  $\epsilon_{T_s}^{(i)} \sim \mathcal{N}(0, I)$ for $i = 1, 2, \ldots, N$
  $\boldsymbol{z}_{T_s}^{(i)} = \sqrt{\bar{\alpha}_{T_s}}\mathbf{E}(y) + \sqrt{1-\bar{\alpha}_{T_s}}\epsilon_{T_s}^{(i)}$
  /* Use MPGD to obtain latent estimations $\{\boldsymbol{z}_{0|t}^{(i)}\}_{i=1}^N$ */
  for $t = T_j, \ldots, 1$ and all i in parallel **do**
    Calculate $\hat{\epsilon}_\theta(\boldsymbol{z}_t^{(i)}, t)$ with Eq. 7
    $\boldsymbol{z}_{0|t}^{(i)} = \frac{1}{\sqrt{\bar{\alpha}_t}}(\boldsymbol{z}_t^{(i)} - \sqrt{1-\bar{\alpha}_t}\hat{\epsilon}_\theta(\boldsymbol{z}_t^{(i)}, t))$
    if $j \neq 1$ **then**
      Update $\boldsymbol{z}_{0|t}^{(i)}$ with Eq. 8
    end if
    $\boldsymbol{z}_{t-1}^{(i)} = \sqrt{\bar{\alpha}_{t-1}}\boldsymbol{z}_{0|t}^{(i)} + \sqrt{1-\bar{\alpha}_{t-1}-\sigma_t^2}\epsilon_\theta(\boldsymbol{z}_t^{(i)}, t) + \sigma_t\epsilon_t^{(i)}$ for $\epsilon_t^{(i)} \sim \mathcal{N}(0, I)$
  end for
  /* Update $\hat{\phi}$ with $\{\boldsymbol{z}_{0|t}^{(i)}\}_{i=1}^N$ and Eq. 9 */
  for $k = 1, \ldots, K$ **do**
    $\hat{\phi} = \text{Adam}(\hat{\phi}, \frac{1}{N}\sum_{i=1}^N L_\phi(\boldsymbol{z}_0^{(i)}, \hat{\phi}))$
  end for
end for
return $\hat{\phi}$

$\hat{\phi}$ Initialization Sometimes initializing the operator parameters is straightforward – for example, for a slight blur, an identity function can be a reasonable starting point. Alternatively, reliable operator priors, such as the pre-trained prior in BlindDPS or the Gaussian prior in GibbsDDRM for linear operators, can also provide effective initializations for simple kernels. However, for complex operator estimator like neural networks, obtaining good initializations becomes more challenging.

To address this, we propose a new algorithm for general operator initialization. Since the goal here is only to initialize $\hat{\phi}$, the quality of intermediate $\boldsymbol{x}$ estimates is less critical as long as they provide useful signals. Unlike other diffusion-based methods that alternate optimization targets at each diffusion step, we use SDEdit [Meng et al., 2022] and MPGD [He et al., 2024] with very few diffusion steps to quickly obtain a batch of $\boldsymbol{x}$ estimates. We then perform maximum likelihood estimation (MLE) on these estimates to update $\hat{\phi}$ using Adam optimizer. As detailed in Algorithm 2, repeating this process can leverage the diffusion prior to quickly converge to a reliable starting point for $\hat{\phi}$.

Notice that we only assume that ϕ can be approximated by differentiable functions, allowing $\mathcal{A}_{\hat{\phi}}$ to be parameterized by general model families such as neural networks. Moreover, as mentioned earlier, any well-performing operator priors can be seamlessly integrated into our framework.

Iterative Refinement After $\hat{\phi}$ initialization, we perform another alternating optimization to refine both the operator and the image. Throughout the diffusion process, we alternate between updating $\boldsymbol{z}_{0|t}$ using MPGD with $\mathcal{A}_{\hat{\phi}}$ fixed and updating the MLE estimation of $\hat{\phi}$ using Adam with $\boldsymbol{z}_{0|t}$ fixed.

Since unlike GibbsDDRM, which uses Langevin dynamics to update the operator, we solve for a local optimum. Therefore, it's reasonable not to update the operator too frequently. Empirically, we find that periodic updates to the operator combined with time-traveling [Lugmayr et al., 2022, Yu et al., 2023] yield the best results. In addition, since MPGD supports any differentiable loss function, we can incorporate regularization to further improve the visual quality. In practice, we use

$$\boldsymbol{z}_{0|t} = \boldsymbol{z}_{0|t} - c_t\nabla_{\boldsymbol{z}_{0|t}}(\|\boldsymbol{y} - \mathcal{A}_{\hat{\phi}}(\mathbf{D}(\boldsymbol{z}_{0|t}))\|_2^2 + \lambda_{\boldsymbol{z}}\text{LPIPS}(\mathbf{D}(\boldsymbol{z}_{0|t}), \boldsymbol{y})) \tag{8}$$

as the MPGD update rule and

$$L_\phi(\boldsymbol{z}_{0|t}, \hat{\phi}) = \|\boldsymbol{y} - \mathcal{A}_{\hat{\phi}}(\mathbf{D}(\boldsymbol{z}_{0|t}))\|_2^2 + \lambda_\phi\|\hat{\phi}\|_1 \tag{9}$$

as the Adam objective. Here $\text{LPIPS}(\cdot)$ denotes the LPIPS distance between two images and $\|\cdot\|_1$ denotes the L_1 regularization term. $\lambda_{\boldsymbol{z}}$ and λ_ϕ are adjustable hyperparameters.

Combining large-scale language-conditioned diffusion priors, effective operator initializations and the iterative refinement process described above, we introduce LADiBI, a new training-free algorithm for blind inverse problem solving that supports diverse target image distributions and flexible operators.

Table 2: Quantitative results on blind linear deblurring tasks. "GibbsDDRM" denotes the results from directly running their open-sourced code and "GibbsDDRM*" denotes the results after adjusting their code to match with the ground truth kernel padding.

Method	FFHQ						AFHQ					
	Motion			Gaussian			Motion			Gaussian		
	LPIPS ↓	PSNR ↑	KID ↓	LPIPS ↓	PSNR ↑	KID ↓	LPIPS ↓	PSNR ↑	KID ↓	LPIPS ↓	PSNR↑	KID ↓
DPS w/ GT kernel	0.164	22.82	0.0046	0.138	24.48	0.0052	0.367	21.28	0.1120	0.330	23.52	0.0813
SelfDeblur	0.732	9.05	0.1088	0.733	8.87	0.0890	0.742	9.04	0.0650	0.736	8.84	0.0352
MPRNet	0.292	22.42	0.0467	0.334	23.23	0.0511	0.324	22.09	0.0382	0.379	21.97	0.0461
DeblurGANv2	0.309	**22.55**	0.0411	0.325	26.61	0.0227	0.323	**22.74**	0.0350	0.340	**27.12**	0.0073
Pan-l0	0.389	14.10	0.1961	0.265	20.68	0.1012	0.414	14.16	0.1590	0.276	21.04	0.0320
Pan-DCP	0.325	17.64	0.1323	0.235	24.93	0.0490	0.371	17.63	0.1377	0.297	25.11	0.0263
BlindDPS	0.246	20.93	0.0153	0.216	25.96	0.0205	0.393	20.14	0.0913	0.330	24.79	0.0268
BIRD	0.294	19.23	0.0491	0.212	21.95	0.0414	0.438	18.92	0.0286	0.320	21.87	0.0089
GibbsDDRM	0.293	20.52	0.0746	0.216	27.03	0.0430	0.303	19.44	0.0265	0.257	24.01	**0.0040**
LADiBI (Ours)	0.230	20.96	**0.0084**	0.197	21.08	**0.0068**	**0.262**	21.20	**0.0132**	**0.204**	24.33	0.0065
GibbsDDRM*	**0.199**	22.36	0.0309	**0.155**	**27.65**	0.0252	0.278	19.00	0.0180	0.224	21.62	**0.0040**

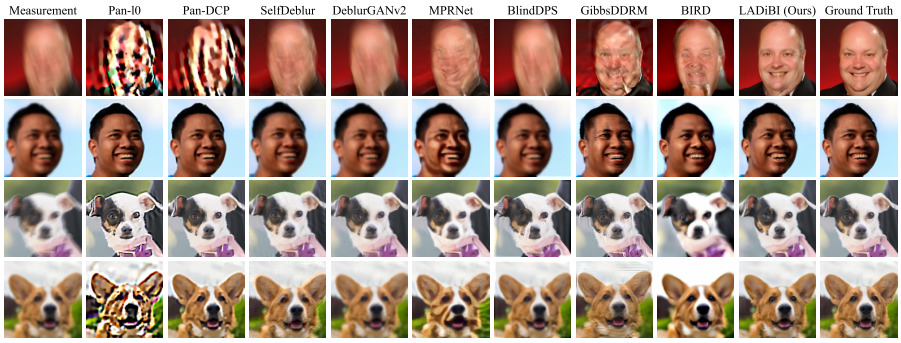


Figure 3: Qualitative results on blind linear deblurring tasks. From top to bottom we showcase examples from motion deblur on FFHQ, Gaussian deblur on FFHQ, motion deblur on AFHQ, and Gaussian deblur on AFHQ respectively.

4 Experiments

4.1 Experimental Setup

We empirically verify the performance of our method with two linear deblurring tasks, Gaussian deblurring and motion deblurring, and a non-linear restoration task, JPEG decompression.

Setup We conduct quantitative comparisons on FFHQ 256 $\times$ 256 [Karras et al., 2019] and AFHQ-dog 256 $\times$ 256 [Choi et al., 2020]. Following Murata et al. [2023], we use 1000 images for FFHQ and 500 images for AFHQ. We use LPIPS [Zhang et al., 2018], PSNR and KID [Bińkowski et al., 2018] to measure the perceptual similarity, pixel accuracy, and image fidelity respectively.

Baselines We compare our method against other state-of-the-art approaches as baselines. Specifically, we choose Pan-l0 [Pan et al., 2017] and Pan-DCP [Pan et al., 2018] as the optimization-based method, SelfDeblur [Ren et al., 2020] as the self-supervised approach, PRNet [Zamir et al., 2021] and DeblurGANv2 [Kupyn et al., 2019] as the supervised baselines, and BlindDPS [Chung et al., 2023a], BIRD [Chihaoui et al., 2024] and GibbsDDRM [Murata et al., 2023] as the diffusion-based methods. All baselines are experiments using their open-sourced code and pre-trained models.

4.2 Blind Linear Deblurring

We first conduct experiments on linear deblurring, which is the design space of most baselines. We evaluate all methods on two blurring kernels: Gaussian blur and motion blur. We apply random motion blur kernels with intensity 0.5 and Gaussian blur kernels with standard deviation 3. Measurements are derived by applying a pixel-wise Gaussian noise with $\sigma = 0.02$. We use a 61×61 convolutional matrix as $\hat{\phi}$ and initialize it as a Gaussian kernel of intensity 6.0 following Murata et al. [2023]

As shown in Table 2 and Figure 3, our method matches the performance of the state-of-the-art method GibbsDDRM, which is explicitly designed to solve linear problems using SVDs. In fact, GibbsDDRM's highly specialized design makes it so sensitive that even small discrepancies (e.g.

Table 3: Quantitative results (the mean and standard deviation) on blind JPEG decompression task.

Method	FFHQ			AFHQ		
	LPIPS ↓	PSNR ↑	KID ↓	LPIPS ↓	PSNR ↑	KID ↓
Pan-I0	0.787 ± 0.084	12.72 ± 2.53	0.2189	0.825 ± 0.083	13.15 ± 2.68	0.2168
Pan-DCP	0.710 ± 0.067	14.91 ± 2.63	0.2318	0.673 ± 0.055	18.27 ± 1.60	0.2472
SelfDeblur	0.676 ± 0.054	8.84 ± 1.93	0.1959	0.703 ± 0.052	8.89 ± 1.72	0.1173
MPRNet	0.785 ± 0.048	5.99 ± 1.81	0.8008	0.769 ± 0.048	5.93 ± 1.78	0.6809
DeblurGAN	0.473 ± 0.058	<u>21.48</u> ± 1.85	0.2207	0.502 ± 0.061	**21.76** ± 1.84	0.1527
BlindDPS	0.431 ± 0.079	**21.55** ± 1.96	0.1791	<u>0.397</u> ± 0.069	20.87 ± 1.89	0.2108
BIRD	<u>0.406</u> ± 0.047	20.68 ± 1.09	<u>0.0525</u>	0.425 ± 0.068	21.08 ± 1.50	<u>0.0673</u>
GibbsDDRM	0.841 ± 0.057	13.91 ± 1.26	0.2915	0.775 ± 0.059	14.59 ± 1.48	0.2915
Ours	**0.268** ± 0.070	21.40 ± 1.18	**0.0172**	**0.315** ± 0.075	<u>21.12</u> ± 1.27	**0.0216**

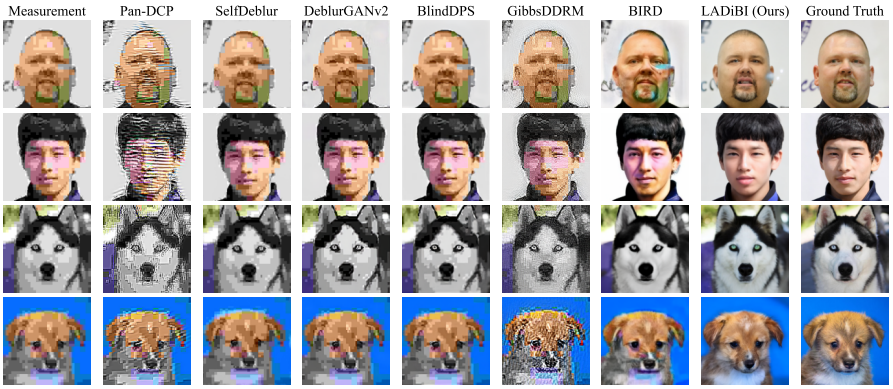


Figure 4: Qualitative results on the blind JPEG decompression task.

kernel padding) between their modeling assumption and the ground truth operator can result in significant performance degradation. Moreover, although BlindDPS and GibbsDDRM use diffusion models that are trained on the exact data distributions tested, our method can still match or outperform them with the large-scale pre-trained model. While supervised methods obtain higher PSNR, our method produces the fewest artifacts, validated by LPIPS and KID scores. Overall, our method offers a competitive performance on linear tasks, even though it is designed for more general applications.

4.3 Blind JPEG Decompression

We further compare all methods on JPEG decompression, a particularly challenging task due to its non-linear, non-differentiable nature. Unlike traditional settings, our experiments provide no task-specific knowledge, such as the compression algorithm, quantization table, or factors – algorithms rely solely on the measurement images. We generate measurements using JPEG compression with a quantization factor of 2. We use a neural network with a 3-layer U-net to parametrize the operator. The operator is initialized using Algorithm 2 with $M = 8$ and $N = 4$.

Table 3 and Figure 4 present the results on the JPEG decompression task. It is evident through both quantitative and qualitative results that our method is the only one capable of enhancing the fidelity of the images and maintaining consistency to measurements. Unlike the baselines, which struggle with this task due to their limited posterior formulations, our flexible framework adapts to approximate this challenging operator and produces high quality images.

5 Conclusion

In this work, we propose LADiBI, a new training-free algorithm to solving blind inverse problems in image restoration using large-scale pre-trained text-to-image diffusion models. With unknown degradation operators, our method leverages text prompts as well as posterior guidance on intermediate diffusion steps to restore desired images based on the measurements. Experiments demonstrate that LADiBI's effectiveness on diverse operator and image distributions.

References

Y. Balaji, S. Nah, X. Huang, A. Vahdat, J. Song, Q. Zhang, K. Kreis, M. Aittala, T. Aila, S. Laine, B. Catanzaro, T. Karras, and M.-Y. Liu. ediff-i: Text-to-image diffusion models with an ensemble of expert denoisers, 2023. URL https://arxiv.org/abs/2211.01324.

A. Bansal, H.-M. Chu, A. Schwarzschild, S. Sengupta, M. Goldblum, J. Geiping, and T. Goldstein. Universal guidance for diffusion models. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 843–852, 2023.

M. Bińkowski, D. J. Sutherland, M. Arbel, and A. Gretton. Demystifying MMD GANs. 2018.

T. Chan and C.-K. Wong. Total variation blind deconvolution. *IEEE Transactions on Image Processing*, 7(3):370–375, 1998. doi: 10.1109/83.661187.

H. Chihaoui, A. Lemkhenter, and P. Favaro. Blind image restoration via fast diffusion inversion. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024. URL https://openreview.net/forum?id=HfSJlBRkKJ.

Y. Choi, Y. Uh, J. Yoo, and J.-W. Ha. Stargan v2: Diverse image synthesis for multiple domains. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020.

H. Chung, J. Kim, S. Kim, and J. C. Ye. Parallel diffusion models of operator and image for blind inverse problems. *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023a.

H. Chung, J. Kim, M. T. Mccann, M. L. Klasky, and J. C. Ye. Diffusion posterior sampling for general noisy inverse problems. 2023b. URL https://openreview.net/forum?id=OnD9zGAGT0k.

H. Chung, J. C. Ye, P. Milanfar, and M. Delbracio. Prompt-tuning latent diffusion models for inverse problems. *arXiv preprint arXiv:2310.01110*, 2023c.

P. Dhariwal and A. Nichol. Diffusion models beat GANs on image synthesis. volume 34, pages 8780–8794, 2021.

Y. Gandelsman, A. Shocher, and M. Irani. " double-dip": unsupervised image decomposition via coupled deep-image-priors. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11026–11035, 2019.

H. Greenspan. Super-resolution in medical imaging. *The Computer Journal*, 52(1):43–63, 2009. doi: 10.1093/comjnl/bxm075.

Y. He, N. Murata, C.-H. Lai, Y. Takida, T. Uesaka, D. Kim, W.-H. Liao, Y. Mitsufuji, J. Z. Kolter, R. Salakhutdinov, and S. Ermon. Manifold preserving guided diffusion. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=o3BxOLoxm1.

J. Ho and T. Salimans. Classifier-free diffusion guidance. In *NeurIPS 2021 Workshop on Deep Generative Models and Downstream Applications*, 2021. URL https://openreview.net/forum?id=qw8AKxfYbI.

J. Ho, A. Jain, and P. Abbeel. Denoising diffusion probabilistic models. 33:6840–6851, 2020.

J. S. Isaac and R. Kulkarni. Super resolution techniques for medical image processing. In *2015 International Conference on Technologies for Sustainable Development (ICTSD)*, pages 1–6, 2015. doi: 10.1109/ICTSD.2015.7095900.

T. Karras, S. Laine, and T. Aila. A style-based generator architecture for generative adversarial networks. pages 4401–4410, 2019.

B. Kawar, M. Elad, S. Ermon, and J. Song. Denoising diffusion restoration models. 2022.

D. Krishnan, T. Tay, and R. Fergus. Blind deconvolution using a normalized sparsity measure. In *CVPR 2011*, pages 233–240, 2011. doi: 10.1109/CVPR.2011.5995521.

O. Kupyn, T. Martyniuk, J. Wu, and Z. Wang. Deblurgan-v2: Deblurring (orders-of-magnitude) faster and better. In *The IEEE International Conference on Computer Vision (ICCV)*, Oct 2019.

C. Laroche, A. Almansa, and E. Coupeté. Fast diffusion em: a diffusion model for blind inverse problems with application to deconvolution. In *IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, 2024.

A. Levin, Y. Weiss, F. Durand, and W. T. Freeman. Understanding and evaluating blind deconvolution algorithms. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1964–1971, 2009. doi: 10.1109/CVPR.2009.5206815.

A. Lugmayr, M. Danelljan, A. Romero, F. Yu, R. Timofte, and L. Van Gool. RePaint: Inpainting using denoising diffusion probabilistic models. pages 11461–11471, 2022.

C. Meng, Y. He, Y. Song, J. Song, J. Wu, J.-Y. Zhu, and S. Ermon. SDEdit: Guided image synthesis and editing with stochastic differential equations. In *International Conference on Learning Representations*, 2022.

N. Murata, K. Saito, C.-H. Lai, Y. Takida, T. Uesaka, Y. Mitsufuji, and S. Ermon. GibbsDDRM: A partially collapsed gibbs sampler for solving blind inverse problems with denoising diffusion restoration. In *International Conference on Machine Learning*, 2023.

J. Pan, Z. Hu, Z. Su, and M.-H. Yang. l_0 -regularized intensity and gradient prior for deblurring text images and beyond. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39(2): 342–355, 2017. doi: 10.1109/TPAMI.2016.2551244.

J. Pan, D. Sun, H. Pfister, and M.-H. Yang. Deblurring images via dark channel prior. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 40(10):2315–2328, 2018. doi: 10.1109/TPAMI.2017.2753804.

D. Ren, K. Zhang, Q. Wang, Q. Hu, and W. Zuo. Neural blind deconvolution using deep priors. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020.

J. Rim, H. Lee, J. Won, and S. Cho. Real-world blur dataset for learning and benchmarking deblurring algorithms. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2020.

R. Rombach, A. Blattmann, D. Lorenz, P. Esser, and B. Ommer. High-resolution image synthesis with latent diffusion models, 2021.

O. Ronneberger, P. Fischer, and T. Brox. U-net: Convolutional networks for biomedical image segmentation, 2015. URL `https://arxiv.org/abs/1505.04597`.

C. Saharia, W. Chan, S. Saxena, L. Li, J. Whang, E. L. Denton, K. Ghasemipour, R. Gontijo Lopes, B. Karagol Ayan, T. Salimans, J. Ho, D. J. Fleet, and M. Norouzi. Photorealistic text-to-image diffusion models with deep language understanding. In S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh, editors, *Advances in Neural Information Processing Systems*, volume 35, pages 36479–36494. Curran Associates, Inc., 2022. URL `https://proceedings.neurips.cc/paper_files/paper/2022/file/ec795aeadae0b7d230fa35cbaf04c041-Paper-Conference.pdf`.

Y. Sanghvi, Y. Chi, and S. H. Chan. Kernel diffusion: An alternate approach to blind deconvolution. *arXiv preprint arXiv:2312.02319*, 2023.

J. Song, C. Meng, and S. Ermon. Denoising diffusion implicit models. 2021a. URL `https://openreview.net/forum?id=St1giarCHLP`.

J. Song, A. Vahdat, M. Mardani, and J. Kautz. Pseudoinverse-guided diffusion models for inverse problems. 2022.

Y. Song, J. Sohl-Dickstein, D. P. Kingma, A. Kumar, S. Ermon, and B. Poole. Score-based generative modeling through stochastic differential equations. 2021b.

S. Tu, W. Yang, and B. Fei. Taming generative diffusion for universal blind image restoration. *arXiv preprint arXiv:2408.11287*, 2024.

D. Ulyanov, A. Vedaldi, and V. Lempitsky. Deep image prior. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 9446–9454, 2018.

WeiminBai, S. Chen, W. Chen, and H. Sun. Blind inversion using latent diffusion priors, 2025. URL https://openreview.net/forum?id=jMffFIWHic.

J. Yu, Y. Wang, C. Zhao, B. Ghanem, and J. Zhang. FreeDoM: Training-free energy-guided conditional diffusion model. *arXiv:2303.09833*, 2023.

L. Yuan, J. Sun, L. Quan, and H.-Y. Shum. Image deblurring with blurred/noisy image pairs. In *ACM SIGGRAPH 2007 Papers*, SIGGRAPH '07, page 1–es, New York, NY, USA, 2007. Association for Computing Machinery. ISBN 9781450378369. doi: 10.1145/1275808.1276379. URL https://doi.org/10.1145/1275808.1276379.

S. W. Zamir, A. Arora, S. Khan, M. Hayat, F. S. Khan, M.-H. Yang, and L. Shao. Multi-stage progressive image restoration. In *CVPR*, 2021.

C. Zhang, C. Zhang, M. Zhang, I. S. Kweon, and J. Kim. Text-to-image diffusion models in generative ai: A survey, 2024. URL https://arxiv.org/abs/2303.07909.

R. Zhang, P. Isola, A. A. Efros, E. Shechtman, and O. Wang. The unreasonable effectiveness of deep features as a perceptual metric. pages 586–595, 2018.

A Additional Implementation Details

In this section, we offer more detailed information regarding the implementation setup for Algorithms 1 and 2. All experiments are conducted using Stable Diffusion 1.4 [Rombach et al., 2021], DDIM 200-step, and $T_s = 150, M = 4$ on an NVIDIA A6000 GPU. In an effort to make our methodology generalizable and extensible to other inverse problems, we attempt to maintain the same hyper-parameter values across tasks wherever possible. Each hyper-parameter value has been chosen after conducting preliminary experiments for a specific range and opting for the value that offers the best performance. Indicative examples of such experiments are shown in C.

In particular, we run Algorithm 1 for $T_s = 150$ timesteps while performing 4 repetitions as part of the time-traveling strategy. We encode the measurement as $\boldsymbol{x}_{T_s}$ by applying the forward diffusion process up to timestep 800. In parallel with the reverse diffusion process, we update $\hat{\phi}$ every 5 timesteps, each time conducting $K = 150$ gradient steps. We adjust the c_t and the λ_ϕ parameters of Equations 8 and 9 to 30 and 2 respectively. In terms of the operator initialization process described by Algorithm 2, we make use of a batch of $N = 4$ samples and run $M = 8$ iterations, each comprising $T_j = 60$ timesteps.

We also take into consideration that the targets and measurements reside in the 256×256 pixel space, whereas Stable Diffusion v1 operates on images of pixel size 512×512. To address this disparity, we initially upsample the measurement using bilinear interpolation in order to transform it to a 512×512 image, and then downsample the resulting x_0 to map the final estimate back to the original image space.

The schematic overviews of Algorithms 1 and 2 are presented in Figures 2 and 5.

In addition, there are some parameters in our approach for which employing a task-aware setup strategy is essential for state-of-the-art performance, including the prompt as well as the architecture of $\hat{\phi}$. Here we provide details with respect to these parameters according to each restoration task:

Motion Deblurring

- Positive prompts: *"a clear headshot of a person/animal"*
- Negative prompts: *"shaken image, motion blur, pixelated, lowres, text, error, cropped, worst quality, blurry ears, low quality, ugly, duplicate, morbid, mutilated, poorly drawn face, mutation, deformed, blurry, dehydrated, blurry hair, bad anatomy, bad proportions, disfigured, gross proportions"*
- $\hat{\phi}$ architecture: A single 61×61 convolutional block with 3 input and 3 output channels.

Gaussian Deblurring

- Positive prompts: *"a clear headshot of a person/animal"*
- Negative prompts: *"blurry, gaussian blur, lowres, text, error, cropped, worst quality, blurry ears, low quality, ugly, duplicate, morbid, mutilated, text in image, DSLR effect, poorly drawn face, mutation, deformed, dehydrated, blurry hair, bad anatomy, bad proportions, disfigured, gross proportions"*
- $\hat{\phi}$ architecture: A single 61×61 convolutional block with 3 input and 3 output channels.

JPEG Decompression

- Positive prompts: *"a clear headshot of a person/animal"*
- Negative prompts: *"pixelated, lowres, text, error, cropped, worst quality, blurry ears, low quality, jpeg artifacts, ugly, duplicate, morbid, mutilated, text in image, DSLR effect, poorly drawn face, mutation, deformed, blurry, dehydrated, blurry hair, bad anatomy, bad proportions, disfigured, gross proportions"*
- $\hat{\phi}$ architecture: A neural network with a typical 3-layer U-net [Ronneberger et al., 2015] architecture. Each layer consists of 2 convolutional blocks of size 3×3 with ReLU activations and number of input and output channels ranging from 32 to 128.

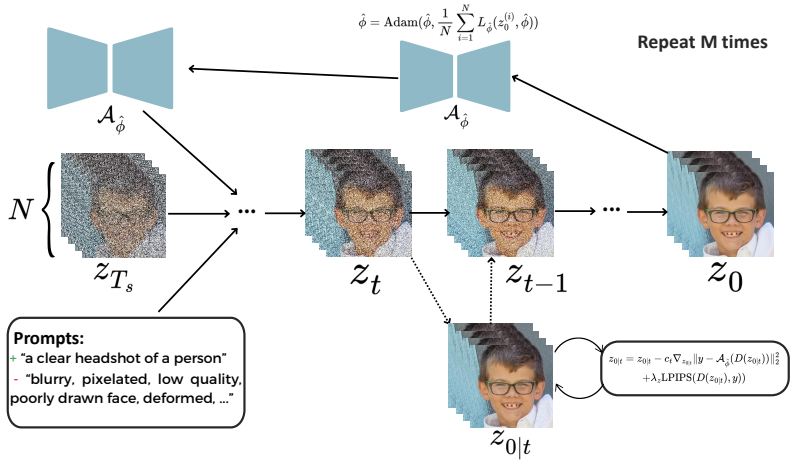


Figure 5: A schematic overview of general operator initialization (Algorithm 2).

We conduct quantitative experiments on FFHQ 256×256 [Karras et al., 2019] and AFHQ 256×256 [Choi et al., 2020]. Images in FFHQ are publicized under Creative Commons BY 2.0, Creative Commons BY-NC 2.0, Public Domain Mark 1.0, Public Domain CC0 1.0, or U.S. Government Works license and AFHQ is publicized under Attribution-NonCommercial 4.0 International license.

The code implementation for LADiBI is linked here.

B Additional Results

In this section, we present additional experimental results on all benchmarks as well as additional tasks and data distributions. Table 4 include the standard deviation of the linear benchmark results. Figures 7, 8 and 9 offer more qualitative comparisons against baseline methods on motion deblurring, gaussian deblurring and JPEG decompression respectively.

In order to show our method's applicability on a wider range of image and operator distributions, we present indicative examples of Gaussian and motion deblurring on Monet paintings in Figure 6. We use the same negative prompts as in the previous experiments and *"a portrait of a person as a Monet style painting"* as positive prompts. We notice that, although the images portray human faces, the baseline method using models trained on realistic human face images cannot accurately solve the problem, while our method effectively generates images consistent with both the measurement image and the painting style present in the ground truth image. In Figure 10, we include additional comparisons against selected baseline methods on the 3 benchmark tasks using a set of Monet and Van Gogh portrait paintings as ground truth images.

We also show additional demonstrations of applying LADiBI to solve colorization and non-linear deblurring problems. Figures 11 and 12 substantiate our claim that our algorithmic scheme is flexible enough to adapt to various inverse tasks. Additionally, thanks to the broad range of image distributions encapsulated by the pretrained model, as well as the minimal assumptions imposed by the proposed methodology, our approach is able to accurately solve inverse problems for a broad range of images. Figures 13 and 14 demonstrate our method's capability to perform Gaussian deblurring, motion deblurring as well as JPEG decompression for images depicting cars and landscapes.

Finally, to assess the real-world performance of our method, we test it on samples from the RealBlur dataset [Rim et al., 2020], which contains naturally captured blurry images using GoPro in real life settings. This dataset presents realistic blur patterns that are significantly more complex and diverse compared to the synthetically generated benchmarks. Figure 15 presents qualitative demos, which highlight our method's ability to recover sharp structures and textures under various real-world blur conditions. These results demonstrate the applicability of our approach to real inverse problem solving tasks and support its generalization capability.

Table 4: Quantitative results on motion and Gaussian deblurring on FFHQ dataset. Mean ± standard deviation shown.

Method	Motion			Gaussian		
	LPIPS ↓	PSNR ↑	KID ↓	LPIPS ↓	PSNR ↑	KID ↓
DPS w/ GT kernel	0.164	22.82	0.0046	0.138	24.48	0.0052
SelfDeblur	0.732 ± 0.147	9.05 ± 2.46	0.1088	0.733 ± 0.094	8.87 ± 2.21	0.0890
MPRNet	0.292 ± 0.101	22.42 ± 3.21	0.0467	0.334 ± 0.068	23.23 ± 2.19	0.0511
DeblurGANv2	0.309 ± 0.111	**22.55** ± 3.44	0.0411	0.325 ± 0.145	26.61 ± 3.28	0.0227
Pan-l0	0.389 ± 0.090	14.10 ± 2.51	0.1961	0.265 ± 0.082	20.68 ± 3.81	0.1012
Pan-DCP	0.325 ± 0.105	17.64 ± 3.63	0.1323	0.235 ± 0.066	24.93 ± 3.60	0.0490
BlindDPS	0.246 ± 0.077	20.93 ± 2.09	0.0153	0.216 ± 0.076	25.96 ± 2.45	0.0205
BIRD	0.294 ± 0.076	19.23 ± 1.88	0.0491	0.212 ± 0.055	21.95 ± 1.62	0.0414
GibbsDDRM	0.293 ± 0.099	20.52 ± 2.81	0.0746	0.216 ± 0.046	27.03 ± 1.87	0.0430
LADiBI (Ours)	0.230 ± 0.076	20.96 ± 2.34	**0.0084**	0.197 ± 0.071	21.08 ± 2.71	**0.0068**
GibbsDDRM*	**0.199** ± 0.110	22.36 ± 3.79	0.0309	**0.155** ± 0.049	**27.65** ± 2.66	0.0252

Table 5: Quantitative results on motion and Gaussian deblurring on AFHQ. Mean ± standard deviation shown.

Method	Motion			Gaussian		
	LPIPS ↓	PSNR ↑	KID ↓	LPIPS ↓	PSNR ↑	KID ↓
DPS w/ GT kernel	0.367	21.28	0.1120	0.330	23.52	0.0813
SelfDeblur	0.742 ± 0.158	9.04 ± 1.84	0.0650	0.736 ± 0.112	8.84 ± 1.45	0.0352
MPRNet	0.324 ± 0.095	22.09 ± 3.01	0.0382	0.379 ± 0.081	21.97 ± 2.55	0.0461
DeblurGANv2	0.323 ± 0.105	**22.74** ± 2.89	0.0350	0.340 ± 0.084	**27.12** ± 2.94	0.0073
Pan-l0	0.414 ± 0.133	14.16 ± 3.97	0.1590	0.276 ± 0.079	21.04 ± 3.39	0.0320
Pan-DCP	0.371 ± 0.147	17.63 ± 5.94	0.1377	0.297 ± 0.086	25.11 ± 3.68	0.0263
BlindDPS	0.393 ± 0.061	20.14 ± 1.67	0.0913	0.330 ± 0.057	24.79 ± 1.76	0.0268
BIRD	0.438 ± 0.110	18.92 ± 2.04	0.0286	0.320 ± 0.079	21.87 ± 1.95	0.0089
GibbsDDRM	0.303 ± 0.114	19.44 ± 3.58	0.0265	0.257 ± 0.121	24.01 ± 4.77	**0.0040**
LADiBI (Ours)	**0.262** ± 0.096	21.20 ± 2.72	**0.0132**	**0.204** ± 0.066	24.33 ± 1.73	0.0065
GibbsDDRM*	0.278 ± 0.099	19.00 ± 3.09	0.0180	0.224 ± 0.091	21.62 ± 3.53	**0.0040**

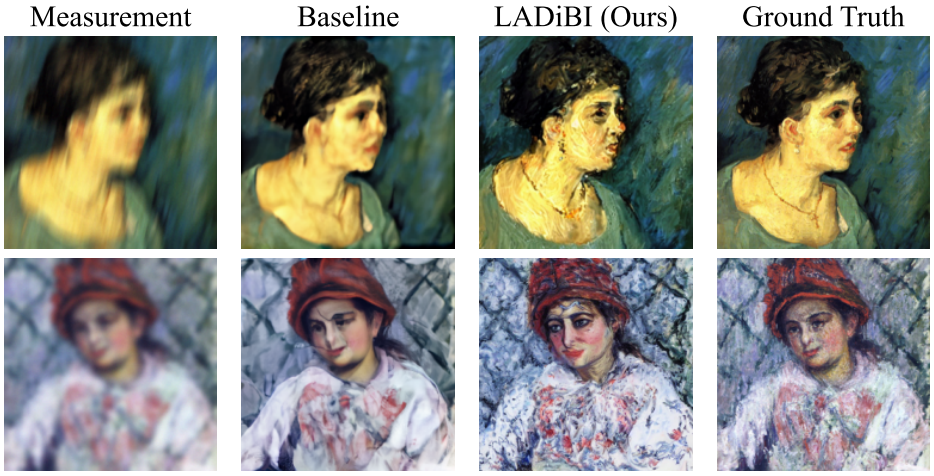


Figure 6: Qualitative results on blind deblurring Monet paintings.

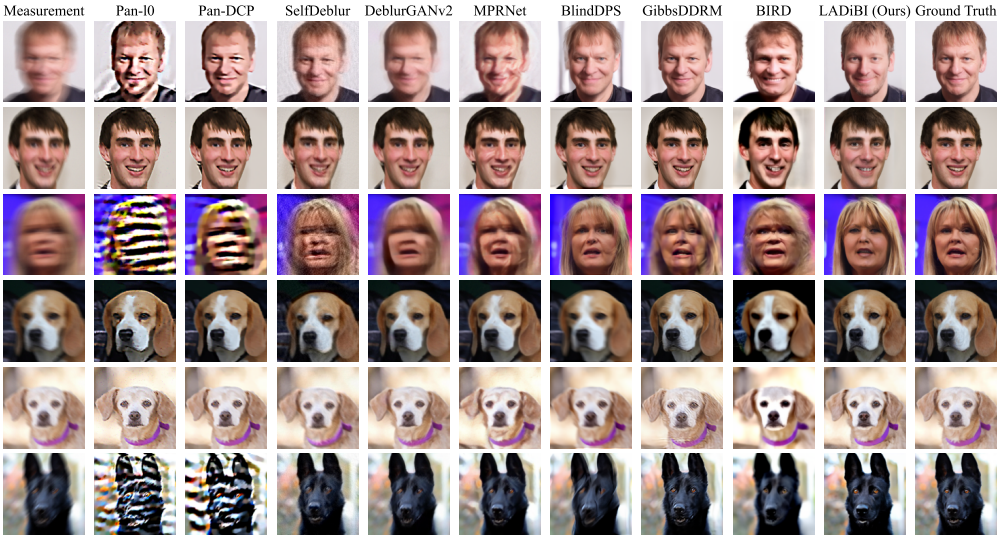


Figure 7: Additional qualitative results on motion deblurring task.

Figure 8: Additional qualitative results on Gaussian deblurring task.

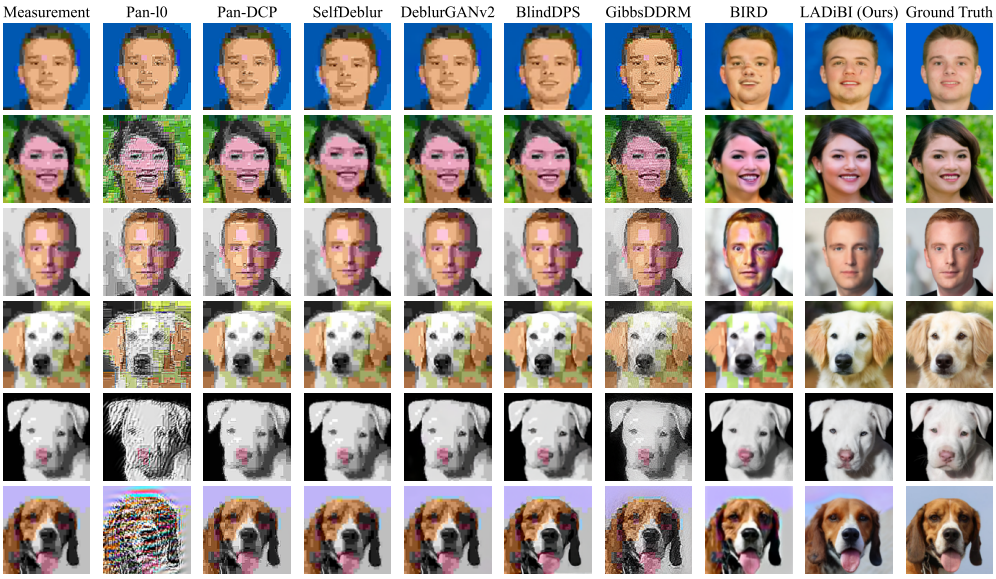


Figure 9: Additional qualitative results on JPEG decompression task.

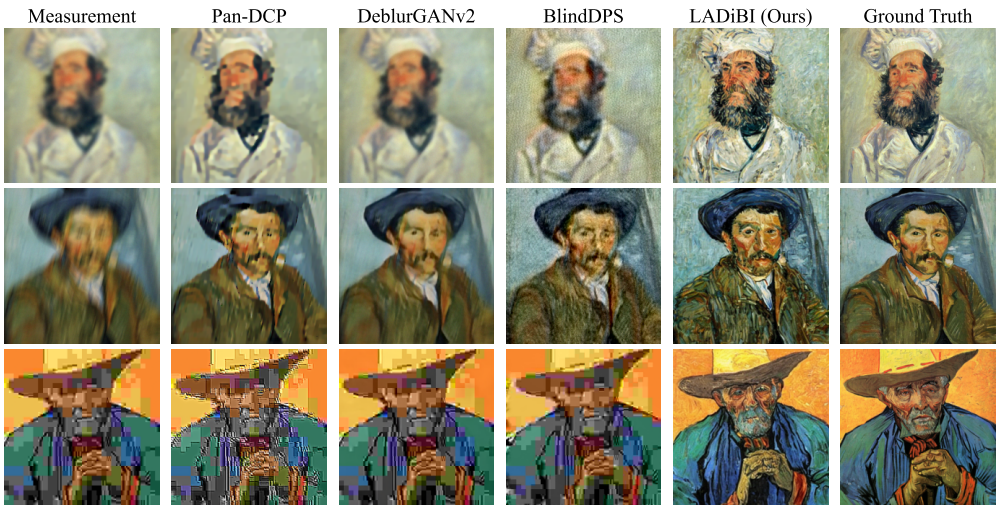


Figure 10: Additional qualitative results on painting restoration.

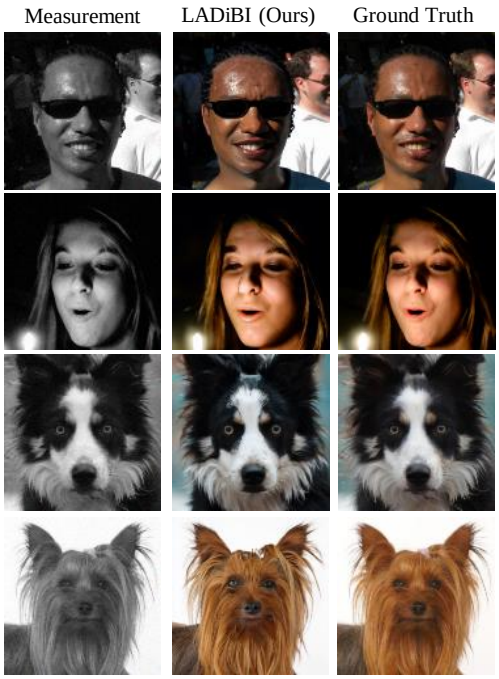


Figure 11: Qualitative results on colorization task.

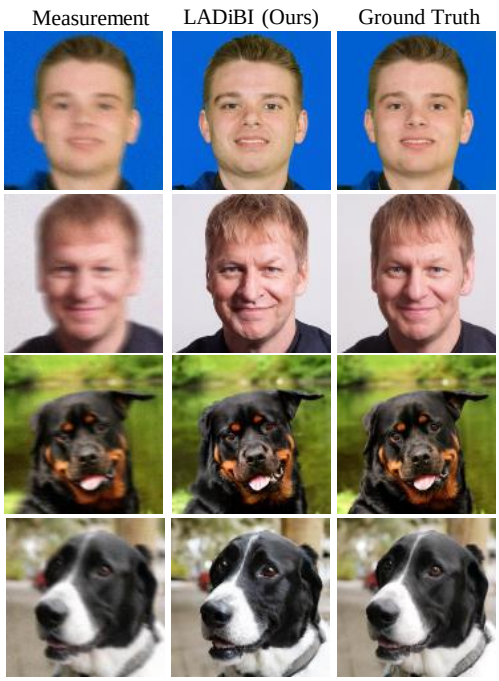


Figure 12: Qualitative results on non-linear deblurring task.

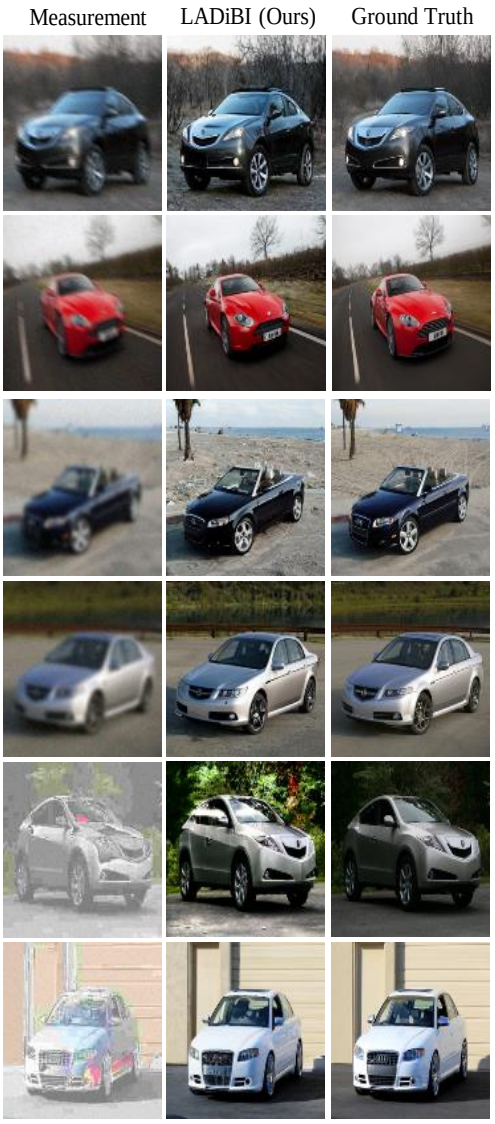


Figure 13: Demo inverse problem solving on car images.

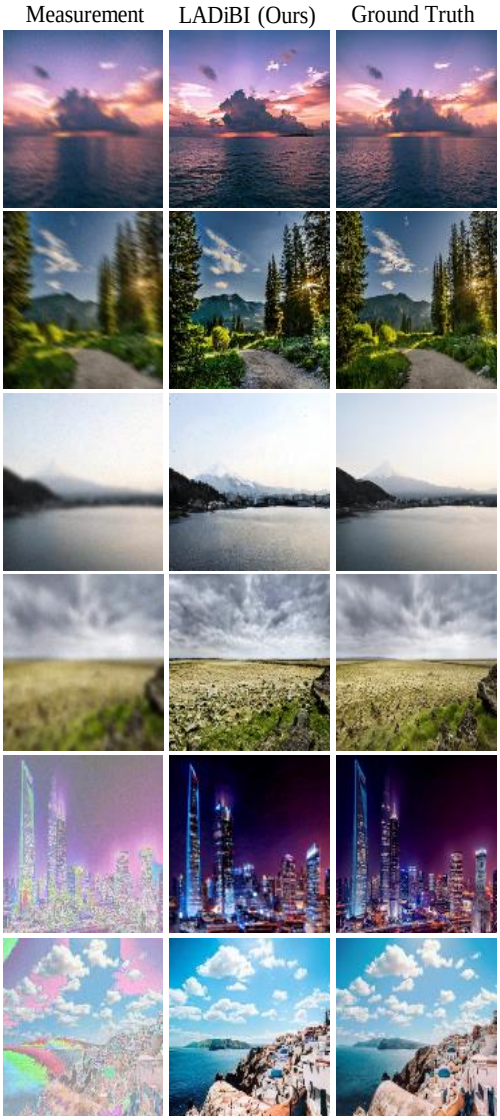


Figure 14: Demo inverse problem solving on landscape images.

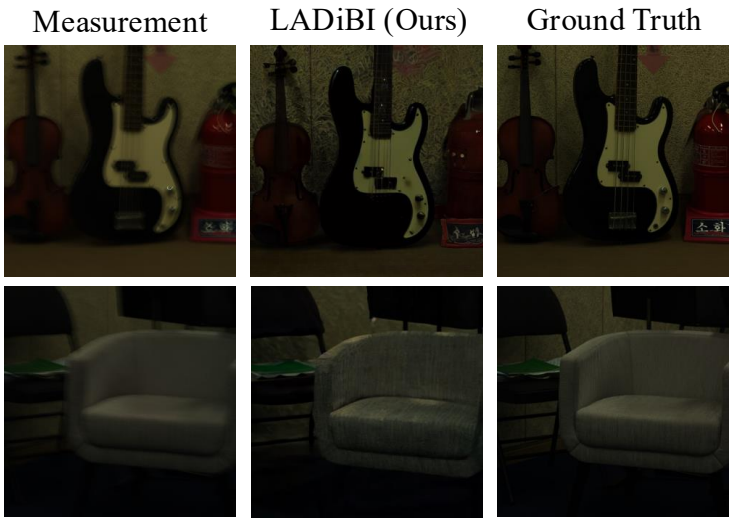


Figure 15: Qualitative samples on the RealBlur dataset.

Table 6: Ablation study on JPEG decompression.

Ablation	LPIPS ↓	PSNR ↑	KID ↓
W/o text prompts	0.508	19.30	0.0319
W/o negative prompts	0.440	19.44	0.0242
Using generic $\boldsymbol{c}_-$	0.425	19.48	0.0255
Using task-irrelevant $\boldsymbol{c}_-$	0.406	19.40	0.0237
W/o MPGD guidance	0.403	19.44	0.0274
W/o regularization	0.307	**21.35**	0.0172
W/o $\hat{\phi}$ Initialization	<u>0.289</u>	20.17	<u>0.0170</u>
LADiBI	**0.262**	<u>21.20</u>	**0.0132**

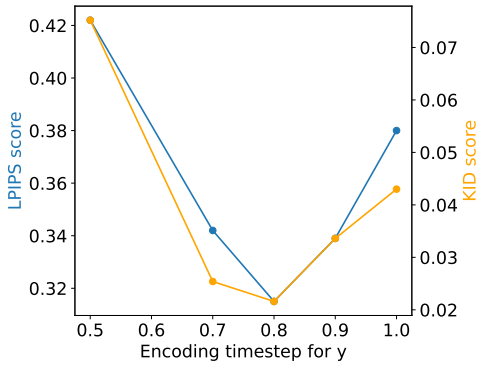


Figure 16: Ablation study on diffusion timestep for encoding the measurement.

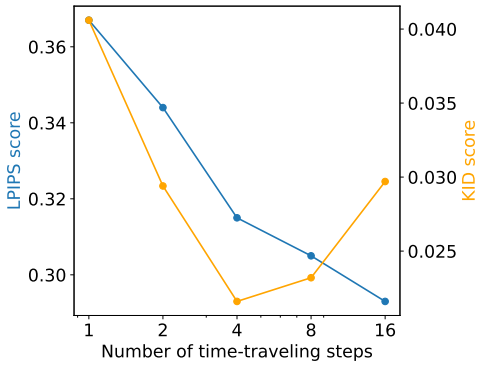


Figure 17: Ablation study on number of time-traveling steps.

C Ablation Study

To showcase the effectiveness of each part of our algorithm, we conduct ablation study on AFHQ dataset with JPEG decompression task. In particular, we test the importance of suitable ad-hoc prompts, the use of MPGD guidance, the MPGD regularization term, and the operator initialization for neural network $\mathcal{A}_{\hat{\phi}}$. We keep SDEdit in all settings to encode the measurement information for fair comparisons. As shown in Table 6, each of the aforementioned components plays a significant role in our scheme and is indispensable for producing high quality results.

To further our ablation, we extend the study on various hyperparameters and components of our LADiBI algorithm on the motion deblurring task using the AFHQ dataset.

Figure 16 presents both the LPIPS and KID score for the value of the timestep T_s at which we encode the measurement using SDEdit [Meng et al., 2022]. Our experiments align with the observation in Meng et al. [2022]: If T_s is too large most of the information about the measurement has been replaced by noise which does not allow the sampling process to leverage useful features of the degraded image. On the other hand, if T_s is too small, the sampling process is not equipped with enough scheduled steps to reach the target distribution.

In addition, Figure 17 presents the effectiveness of taking advantage of the time-traveling strategy. More time-traveling boosts the overall performance up to a specific value, after which we begin to notice a trade-off between perceptual clarity of the image and fidelity to the target distribution.

Finally, we evaluate the performance of our algorithm when using a unconstrained configuration for $\mathcal{A}_{\hat{\phi}}$ in constrained tasks. In particular, we test the neural network architecture for the operator in linear inverse problems and we showcase quantitative results for the motion deblurring task in Table 7, and qualitative samples in Figure 18. We observe that, although performing worse than LADiBI employed with the aligned linear operator architecture, the neural operator is still capable of producing estimates of decent quality while also preserving a design that allows applicability to highly unconstrained image restoration task.

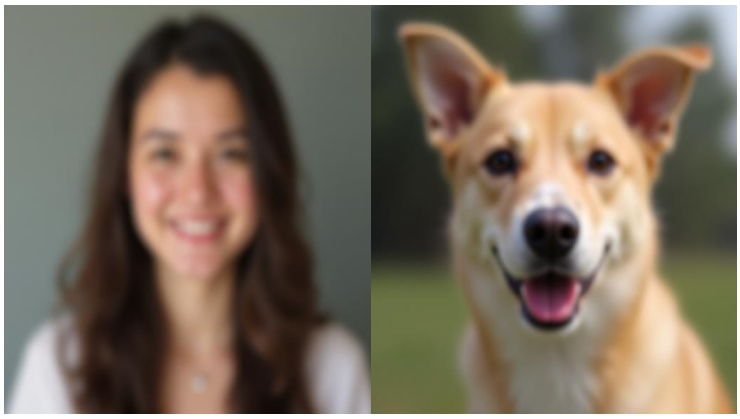
Table 7: Quantitative comparison between linear and deep operator architecture

	AFHQ Motion Blur		
Method	LPIPS ↓	PSNR ↑	KID ↓
Panl0	0.414	14.16	0.1590
PanDCP	0.371	17.63	0.1377
SelfDeblur	0.742	9.04	0.0650
MPRNet	0.324	<u>22.09</u>	0.0382
DeblurGANv2	0.323	**22.74**	0.0350
BlindDPS	0.393	20.14	0.0913
GibbsDDRM	0.303	19.44	0.0265
GibbsDDRM*	<u>0.278</u>	19.00	0.0180
LADiBI (Kernel Operator)	**0.262**	21.20	**0.0132**
LADiBI (U-Net Operator)	0.343	18.93	<u>0.0146</u>

Figure 18: Comparison between linear and deep operator architecture on linear tasks.

D Prior Coverage of Large Pre-trained Text-to-image Diffusion

In this section we provide an empirical justification for using Stable Diffusion v1.4 as our base model. To demonstrate this property, Figure 19 provides qualitative samples of images with Gaussian blur, motion blur and after JPEG compression. These examples show that Stable Diffusion 1.4 already capture rich distributions of both target images and common degradation artifacts.



(a) Generated images with Gaussian blur effect.

(b) Generated images with motion blur effect.

(c) Generated images with JPEG compression effect.

Figure 19: Drawn samples from the posterior distribution using our baseline model.

E Limitations and Future Works

In this section, we discuss the limitation of LADiBI and potential future works to address these problems.

Similar to many training-free diffusion posterior sampling algorithms [Chung et al., 2023a, Murata et al., 2023, Chung et al., 2023b, He et al., 2024], our method is also sensitive to hyper-parameter tuning. We have provided details about our hyperparameter choices in the previous sections, and we will release our code in a public repository upon publication of this paper.

In addition, while generally simple, we do require the users to infer appropriate prompts from the measurement. An interesting future work direction can include automated prompt tuning similar to the method proposed in Chung et al. [2023c].

Another notable drawback of our method is that LADiBI requires significantly longer inference time in comparison to the best performing baseline (i.e. GibbsDDRM [Murata et al., 2023]) in order to obtain high quality restorations. With the general operator initialization, our algorithm can take around 5 minutes to complete on a single NVIDIA A6000 GPU while GibbsDDRM only takes around 30 seconds. Although this is justifiable by the larger optimization space that we operate on, investigating on how to reduce the inference time requirement is an interesting and critical next step for our work.

Lastly we would like to note that, while neural networks, as demonstrated in the previous sections, can serve as a general model family for various operator functional classes and achieve satisfactory results, obtaining state-of-the-art performance for linear tasks within a reasonable inference time still requires resorting to a linear kernel as the estimated operator. Exploring neural network architectures that can easily generalize across different operator functional classes while achieving state-of-the-art results efficiently remains an exciting direction for future work.

F Impact Statement

Lastly, since our algorithm leverages large pre-trained image generative models, we would like to address the ethical concerns, societal impact as well as the potential harm that can be caused by our method when being used inappropriately.

As a consequence of using large pre-trained text-to-image generative models, our method also inherits potential risks associated with these pre-trained models, including the propagation of biases, copyright infringement and the possibility of generating harmful content. We recognize the significance of these ethical challenges and are dedicated to responsible AI research practices that prevent reinforcing these ethical considerations. Upon the release our code, we are committed to implement and actively update safeguards in our public repository to ensure safer and more ethical content generation practices.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: Our abstract and introduction accurately reflect the paper's main claims.

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: The limitation is discussed in Section E.

   Guidelines:

   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [NA]

Justification: This paper does not include theoretical results.

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: All experimental details are disclosed in Section **??** and Section A.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [Yes]

Justification: We have provided the code implementation of our proposed method here.

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

Answer: [Yes]

Justification: All experimental details are disclosed in Section **??** and Section A.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
- The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

Answer: [Yes]

Justification: We have provided the standard deviation results as the error bars.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.

- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: All experimental details are disclosed in Section **??** and Section A.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: The research is conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics.

   Guidelines:

   - The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: The societal impacts of our work is discussed in Section F.

    Guidelines:

    - The answer NA means that there is no societal impact of the work performed.
    - If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.

- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

    Answer: [Yes]

    Justification: The safeguard implementation for the code release of our work is discussed in Section F.

    Guidelines:

    - The answer NA means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: We have cited the dataset used in this paper in Section **??** and Section A.

    Guidelines:

    - The answer NA means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.

- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [Yes]

    Justification: The only new asset we create in this paper is the code implementation of our method. We have provided the code implementation of our proposed method here.

    Guidelines:

    - The answer NA means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [NA]

    Justification: This paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [NA]

    Justification: This paper does not involve crowdsourcing nor research with human subjects

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does not impact the core methodology, scientific rigorousness, or originality of the research, declaration is not required.

Answer: [NA]

Justification: The core method development in this research does not involve LLMs as any important, original, or non-standard components.

Guidelines:

- The answer NA means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy (`https://neurips.cc/Conferences/2025/LLM`) for what should or should not be described.