

---

# Blind Image Restoration via Fast Diffusion Inversion

---

**Hamadi Chihaoui   Abdelhak Lemkhenter   Paolo Favaro**

Computer Vision Group, Institute of Informatics, University of Bern, Switzerland  
 {hamadi.chihaoui, abdelhak.lemkhenter, paolo.favaro}@unibe.ch

## Abstract

Recently, various methods have been proposed to solve Image Restoration (IR) tasks using a pre-trained diffusion model leading to state-of-the-art performance. However, most of these methods assume that the degradation operator in the IR task is completely known. Furthermore, a common characteristic among these approaches is that they alter the diffusion sampling process in order to satisfy the consistency with the degraded input image. This choice has recently been shown to be sub-optimal and to cause the restored image to deviate from the data manifold. To address these issues, we propose Blind Image Restoration via fast Diffusion inversion (BIRD) a blind IR method that jointly optimizes for the degradation model parameters and the restored image. To ensure that the restored images lie onto the data manifold, we propose a novel sampling technique on a pre-trained diffusion model. A key idea in our method is not to modify the reverse sampling, *i.e.*, not to alter all the intermediate latents, once an initial noise is sampled. This is ultimately equivalent to casting the IR task as an optimization problem in the space of the input noise. Moreover, to mitigate the computational cost associated with inverting a fully unrolled diffusion model, we leverage the inherent capability of these models to skip ahead in the forward diffusion process using large time steps. We experimentally validate BIRD on several image restoration tasks and show that it achieves state of the art performance on all of them. Our code is available at <https://github.com/hamadichihaoui/BIRD>.

## 1 Introduction

Recent advances in the field of generative learning due to diffusion models [6, 17] and better neural network architecture choices [7, 23] have led to models capable of generating detailed high-resolution images. In addition to their data generation capabilities, these models also provide a smooth representation of the learned data distribution, which can be used for various applications, such as image restoration. Indeed, different approaches [8, 11, 20] have emerged to solve inverse problems using pretrained diffusion models. Those approaches start from a random noise vector as the diffusion input and recursively alter the diffusion sampling process. After each diffusion reverse step, a projection-based step is added to enforce the consistency with the corrupted image. [3] shows that this procedure may cause the generated image to leave the data manifold during the iterative denoising process, and thus produce unrealistic images. The authors propose an additional correction at each reverse step term to better satisfy the manifold constraint. The recent work [2] proposes a more general framework to handle both non-linear and noisy cases. However, these methods [2, 3] are non-blind as they assume to have complete knowledge of the degradation model, which is not practical for real-world applications. BlindDPS [1] recently proposes an extension of [2] to the case of blind deblurring. They train a score-based/diffusion model for the blur operator. At test-time, they jointly optimize for both the sharp image and blur operator by running the diffusion reverse process. However, training a model for each new degradation operator can be time-consuming.

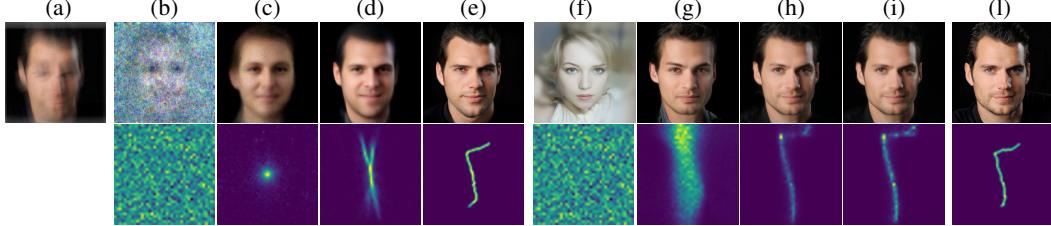


Figure 1: Blind image deblurring with unknown motion blur. (a): blurry input image. From (b) to (e): The top row shows the predictions of BlindDPS [1] as the iterations increase; the bottom row shows the corresponding estimated blur kernel. From (f) to (i): Same estimates as in (b) to (e), but obtained from BIRD, our proposed method. (l) is the ground truth sharp image (top) and ground truth blur kernel (bottom). Notice that BlindDPS [1] trains a score-based model for the kernel estimation, while BIRD does not use any training and can adapt to any new kernel directly at test time. BIRD yields always natural images at every iteration of the reconstruction procedure. Finally, notice that despite recovering a suboptimal blur kernel, the image reconstructed with BIRD is more similar to the ground truth image than with BlindDPS thanks to the robustness of our image generation procedure.

To overcome all these challenges, we propose a novel blind image restoration (IR) method that jointly optimizes the degradation model parameters and the restored image only at test time (and thus separately for each new image). We call this method BIRD, which stands for Blind Image Restoration via fast Diffusion inversion. Since BIRD does not need to pre-train a model for the degradation operator, it can be used immediately on a wide range of IR tasks such as Gaussian deblurring, motion deblurring, super-resolution and JPEG de-artifacting (see Figure 2). In this work, we show that having a strong image prior and ensuring that the restored image never leaves the data manifold during the reconstruction process are fundamental properties to generalize to a wide range of image degradation problems. As shown in Figure 1, BIRD (see (f) to (i)) reconstructs realistic images at each iteration, while BlindDPS (see (b) to (e)) may initially reconstruct non realistic images. We observe that even though we have a more primitive blur reconstruction procedure (we directly regress the motion blur kernel), the reconstructed image is still more similar to the ground truth one (see column (l)) than the final image estimate from BlindDPS (see (e) top), which uses a quite accurate blur kernel estimate (see (e) bottom). This speaks of the importance of defining a strong image prior first and foremost.

To define the image prior, BIRD uses a pre-trained Denoising Diffusion Implicit Model (DDIM) [16] and exploits the existing deterministic correspondence between noise and images in DDIMs by casting the inverse restoration problem as a latent estimation problem, *i.e.*, where the latent variable is the input noise to the diffusion model. In contrast to prior work, we do not alter the reverse sampling, *i.e.*, all the intermediate latents, once an initial noise is sampled. To then apply our image prior, we cast the IR task as an alternating optimization between the unknown restored image and the unknown parameters of the degradation model. However, a direct implementation of the optimization procedure in the case of a diffusion model used as a black box would be computationally demanding and ultimately impractical. To mitigate the substantial computational cost associated with inverting a fully unrolled diffusion model, we leverage, for the first time in IR tasks, the inherent capability of these models to skip ahead in the forward diffusion process using arbitrarily large time steps. We show that our method is able to achieve state of the art performance across a wide range of blind image restoration tasks. Our main contributions can be summarized as follows

1. BIRD is the first to cast an IR task as a latent optimization problem (where only the initial noise is optimized) in the context of diffusion models; our optimization procedure aims at generating images that lie on the data manifold at every iteration;
2. BIRD is computationally efficient; we propose a fast diffusion model inversion in the context of image restoration, without fine-tuning nor retraining;
3. We achieve state of the art results on CelebA and ImageNet for different blind IR tasks, such as Gaussian and motion deblurring, super-resolution, and JPEG de-artifacting.

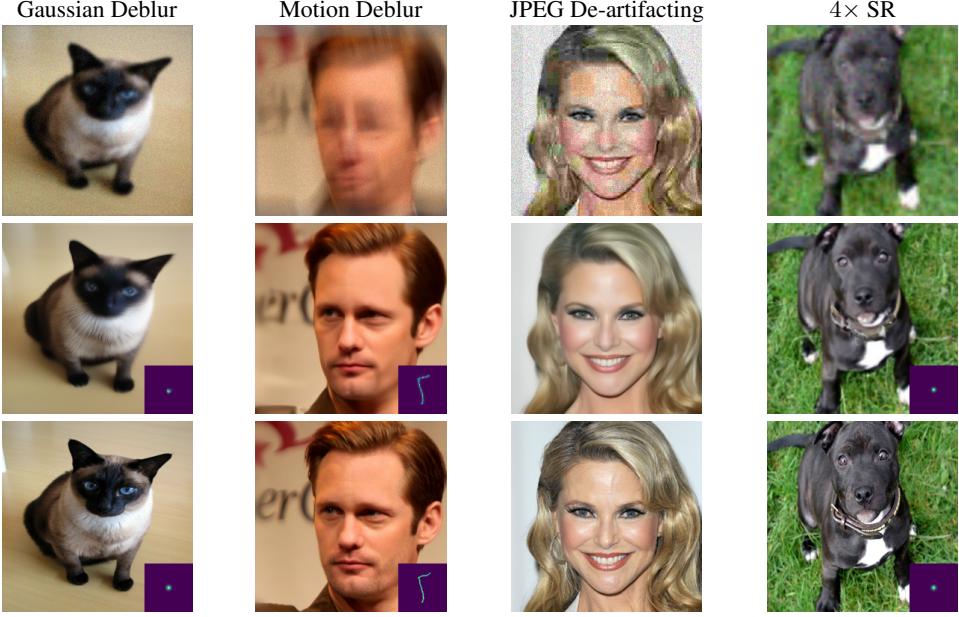


Figure 2: We demonstrate BIRD on several **blind** image restoration problems (*i.e.*, when the values of the degradation model are unknown): Gaussian deblurring, motion deblurring, JPEG restoration (de-artifacting), and superresolution (SR) (it includes additional Gaussian blur). BIRD is applicable to a single degraded image and does not require re-training or fine-tuning of the prior model (we use a diffusion model). Although some of the generated degraded images use Gaussian blur, BIRD recovers a generic blur kernel (without making any Gaussianity assumption).

## 2 Related Work

**Blind and non-blind image restoration methods.** An image restoration task is often cast as an inverse problem, where the degradation model is explicitly known. For example, in image deblurring the degradation model can be described by a convolution with a blur kernel. Methods that explicitly assume the exact knowledge of the blur kernel are called *non-blind* methods. More general methods that assume only knowledge of the type of degradation (*e.g.*, blurring vs JPEG compression), but not the values of its parameters, are instead called *blind*. Some recent examples of non-blind methods are [3] and DPS [2]. [3] points out that relying on an iterative procedure consisting of reverse diffusion steps and a projection-based consistency step runs the risk of stepping outside of the data manifold, a risk they mitigate using an additional correction term. DPS [2] proposes a more general framework to handle both the non-linear and noisy cases. DDNM [20] proposes a zero-shot framework for IR tasks based on the range-null space decomposition. The method works by refining only the null-space contents during the reverse diffusion process, to satisfy both data consistency and realness. Some recent examples of blind methods are BlindDPS [1] and [5]. [5] leverage DDPM and solve inverse problems via hierarchical guidance and a patch-based method. In contrast to these works, in BIRD we introduce a blind image restoration method based on a novel diffusion inversion for DDIM. Our method is computationally efficient, as it requires only a few reverse diffusion steps, and generates realistic samples by only estimating the initial noise.

**Methods based on GAN inversion.** Since our proposed method is based on inverting a diffusion process, we also briefly review related work in the literature and discuss how it differs from our approach. Because Generative Adversarial Networks (GANs) were among the best image generation models, a number of methods [13, 14, 21, 22] invert pre-trained GANs to solve image restoration problems. DPGGAN [13] performs the GAN inversion on a single new corrupted image. The authors show that the inversion is not easily obtainable just with the direct optimization of the latent input to the frozen GAN model. Thus, they propose to also fine-tune the (unfrozen) GAN, while optimizing for the latent vector. [22] instead train an encoder to invert a pre-trained GAN on a dataset of corrupted images (through masking). At test time they directly apply the trained encoder and



Figure 3: Illustration of our proposed accelerated image sampling of the pre-trained DDIM. (a) initial noise  $x_T \sim \mathcal{N}(0, \mathbf{I})$  ( $T = 1000$ ). (b), (c), (d), (e) and (f) are samples  $x_0$  generated using  $\text{DDIMReverse}(x_T, \delta t)$  with  $\delta t = 100, 50, 20, 1$  respectively. Notice how the generated images are all realistic regardless of the choice of the step size  $\delta t$ .

the GAN on a new corrupted image without further training. BIRD shares the aim of inverting a generative model with the above prior work. However, as demonstrated in the literature for inverting GAN models, the inversion of generative models is far from a straightforward task. Moreover, the inversion of GAN and diffusion models are fundamentally different in nature. For example, while GANs generate samples in “one forward pass,” diffusion models require multiple (denoising) steps. Also, all IR methods based on GAN inversion, either train an auxiliary network (*e.g.*, an encoder) or fine-tune the pre-trained GAN network. In our method, the diffusion model is not fine-tuned/trained.

### 3 Image Restoration via BIRD

We introduce the classic Bayesian formulation of inverse problems applied to a single degraded image. An important component in this formulation is the characterization of the image prior, which we do via DDIM diffusion models.

#### 3.1 Problem Formulation

Image restoration (IR) can be cast as a Maximum a Posteriori (MAP) optimization problem [24]

$$\hat{x} = \arg \max_x \log p(y|x) + \log p(x), \quad (1)$$

where  $\hat{x} \in \mathbb{R}^{N_x \times M_x}$  is the restored image of size  $N_x \times M_x$  pixels,  $y \in \mathbb{R}^{N_y \times M_y}$  is the degraded image of size  $N_y \times M_y$  pixels, where  $p(y|x)$  is the so-called *likelihood* and  $p(x)$  the *prior* distribution. Inverse problems describe IR tasks with a degradation operator  $H_\eta : \mathbb{R}^{N_x \times M_x} \rightarrow \mathbb{R}^{N_y \times M_y}$  and a noise image  $n \in \mathbb{R}^{N_y \times M_y}$ , such that the observed degraded image  $y$  can be written as

$$y = H_\eta(x) + n. \quad (2)$$

$\eta$  is a vector with all the parameters that define the operator  $H_\eta$ . IR tasks such as image denoising, deblurring and super-resolution can all be described using specific choices of  $H_\eta$ . For instance, in the case of image deblurring, the  $H_\eta$  operator can be described as a convolution with a blur kernel  $k$  such that  $H_\eta(x) = k * x$ , where  $*$  denotes the convolution operation. In the blind deblurring problem, one assumes that  $H_\eta$  takes the form of a convolution, but without knowing its parameters  $\eta$ , which, in this case, are the values of the true blur kernel  $k$ . Thus, we also need to estimate the parameters  $\eta$  as part of the optimization procedure. Only in the special case of JPEG de-artifacting we use  $H_\eta(x) = x$ , *i.e.*, the identity function, for simplicity. By assuming that the noise  $n$  is zero mean Gaussian in the MAP formulation, one can rewrite eq. (1) as

$$\hat{x}, \hat{\eta} = \arg \min_{x \in \mathbb{R}^{N_x \times M_x}, \eta} \|y - H_\eta(x)\|^2 + \lambda \mathcal{R}(x), \quad (3)$$

where  $\mathcal{R}(x) = -\log p(x)$  is also called a *regularization* term, and  $\lambda > 0$  is a coefficient that regulates the interplay between the likelihood and the prior.

---

**Algorithm 1** BIRD: Image Restoration

---

**Require:** Degraded image  $y$ , step size  $\delta t$ , the dimension of  $x$   $d = N_x \times N_y$ , learning rate  $\alpha$ , the stop threshold  $\varepsilon$   
**Ensure:** Return  $\hat{x}_0$

- 1: Initialize  $x_T^0 \sim \mathcal{N}(0, \mathbf{I})$  and  $H_{\eta^0}$  with random parameters
- 2: **while**  $k : 1 \rightarrow N$  and  $\mathcal{L}_{IR}(x_0^k, H_{\eta^k}) > \varepsilon$  **do**
- 3:    $x_0^k = \text{DDIMReverse}(x_T^k, \delta t)$
- 4:    $x_T^{k+1} = x_T^k - \alpha \nabla_{x_T} \mathcal{L}_{IR}(x_0^k, H_{\eta^k})$
- 5:    $x_T^{k+1} = \frac{x_T^{k+1}}{\|x_T^{k+1}\|} \sqrt{d}$
- 6:    $\eta^{k+1} = \eta^k - \alpha \nabla_{\eta} \mathcal{L}_{IR}(x_0^k, H_{\eta^k})$
- 7: **end while**
- 8: **return**  $\hat{x}_0 = \text{DDIMReverse}(x_T^N, \delta t)$

---

**Algorithm 2** DDIMReverse ( $x_T, \delta t$ )

---

- 1:  $t = T$
- 2: **while**  $t > 0$  **do**
- 3:    $\hat{x}_{0|t} = (x_t - \sqrt{1 - \bar{\alpha}_t} \epsilon_{\theta}(x_t, t)) / \sqrt{\bar{\alpha}_t}$
- 4:    $x_{t-\delta t} = \sqrt{\bar{\alpha}_{t-\delta t}} \hat{x}_{0|t} + \sqrt{1 - \bar{\alpha}_{t-\delta t}} \cdot \frac{x_t - \sqrt{\bar{\alpha}_t} \hat{x}_{0|t}}{\sqrt{1 - \bar{\alpha}_t}}$
- 5:    $t \leftarrow t - \delta t$
- 6: **end while**
- 7: **return**  $\hat{x}_0$

---

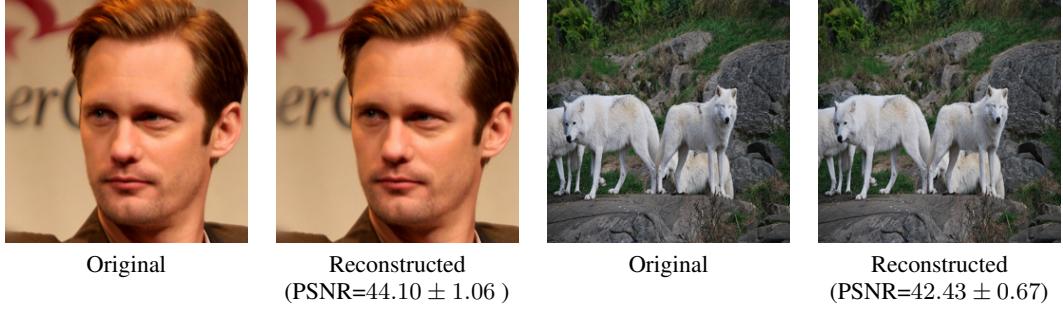


Figure 4: Reconstruction results using BIRD on samples from CelebA and ImageNet validation datasets. The PSNR mean and standard deviation are computed over 10 runs.

Under some mild assumptions, the formulation eq. (3) is known to be equivalent to the constrained optimization

$$\hat{x}, \hat{\eta} = \arg \min_{x \in \mathbb{R}^{N_x \times M_x}, \eta} \|y - H_{\eta}(x)\|^2 \quad (4)$$

$$\text{subject to } \mathcal{R}(x) \leq \rho, \quad (5)$$

for a certain coefficient  $\rho$ . Let us now call  $g$  the mapping that induces the prior  $p(x)$  as the pushforward density of the Normal probability density function [9]. That is,  $g_* q = p$  with  $q(z) = \mathcal{N}(z; 0, \mathbf{I})$ . Then, we can rewrite problem (4) as

$$\hat{x}, \hat{\eta} = \arg \min_{z \in \mathbb{R}^{N_x \times M_x}, \eta} \|y - H_{\eta}(g(z))\|^2 \quad (6)$$

$$\text{subject to } \mathcal{R}(g(z)) \leq \rho. \quad (7)$$

We assume that the mapping  $g$  is such that  $p(g(z)) \geq e^{-\rho}$  when  $\|z\|^2 = N_x \times M_x$ . In high dimensions,  $\frac{1}{N_x \times M_x} \|z\|^2 = \frac{1}{N_x \times M_x} \sum_{i,j} z[i, j]^2 \rightarrow \text{Variance}(z) = 1$  as  $N_x \times M_x \mapsto \infty$ . In fact, most of the density of a high-dimensional Normal random variable is around  $\|z\|^2 = N_x \times M_x$  [12, 19]. We thus assume that  $g$  maps  $z$  with high density  $q(z)$  to  $x = g(z)$  with high density  $p(x)$ . Under our assumption, we can approximate problem (6) with

$$\hat{z}, \hat{\eta} = \arg \min_{z: \|z\|^2 = N_x \times M_x, \eta} \|y - H_{\eta}(g(z))\|^2, \quad (8)$$

and then use  $\hat{x} = g(\hat{z})$ . As  $g$  we adopt a pre-trained diffusion model and choose the Denoising Diffusion Implicit Model (DDIM) [16]. To solve problem (8), we then use an alternating minimization with gradient descent and a normalization step to implement the constraint on  $z$  (see Algorithm 1). Since gradient descent requires computing the derivatives of  $g$ , a diffusion model, with respect to its argument  $z$ , we discuss how to do that efficiently in the next sections.

### 3.2 An Efficient Diffusion Inversion

As previously mentioned, we adopt DDIM as our pre-trained generative process and set it as a fully deterministic diffusion process. We propose an optimization-based iterative procedure to solve problem (8). Thus, we jointly estimate a realistic image  $x$  (*i.e.*, with a high  $p(x)$ ) and the forward degradation model  $H_\eta$  such that  $H_\eta(x) \approx y$ . In order to find  $x$ , we explicitly exploit the correspondence between image samples  $x$  and their associated initial latent image  $z$  in DDIM, in contrast to [8, 11, 20]. Thus, we aim to find the initial noise sample  $z$  that can generate the image  $x$  when applied to DDIM. To keep the notation consistent with the DDIM formalism, we denote images  $x$  with  $x_0$  and the samples  $z$  with  $x_T$ , where  $T$  is the number of iterations in the diffusion model. To make our presentation self-contained, we briefly revise the notation and notions of diffusion models.

#### 3.2.1 Background: Denoising Diffusion Probabilistic and Implicit Models

Denoising Diffusion Probabilistic Models (DDPM) [6] leverage diffusion processes in order to generate high quality image samples. The aim is to reverse the forward diffusion process that maps images to noise, either by relying on a stochastic iterative denoising process or by learning the explicit dynamics of the reverse process, *e.g.*, through an ODE [16]. More precisely the forward diffusion process maps an image  $x_0 \sim p(x)$  to a zero-mean Gaussian  $x_T \sim \mathcal{N}(0, \mathbf{I})$  by generating intermediate images  $x_t$  for  $t \in (0, T]$  which are progressively noisier versions of  $x_0$ . DDPM [6] adopts a Markovian diffusion process, where  $x_t$  only depends on  $x_{t-1}$ . Given a non-increasing sequence  $\alpha_{1:T} \in (0, 1]$ , the joint and marginal distributions of the forward diffusion process are described by

$$q(x_{1:T}|x_0) = \prod_{t=1}^T q(x_t|x_{t-1}), \text{ where } q(x_t|x_{t-1}) = \mathcal{N}\left(\sqrt{\alpha_t}x_{t-1}, (1 - \alpha_t)\mathbf{I}\right), \quad (9)$$

which implies that we can sample  $x_t$  simply by conditioning on  $x_0$  with  $\bar{\alpha}_t = \prod_{s \leq t} \alpha_s$  via

$$q(x_t|x_0) = \mathcal{N}\left(\sqrt{\bar{\alpha}_t}x_0, (1 - \bar{\alpha}_t)\mathbf{I}\right). \quad (10)$$

To invert the forward process, one can train a model  $\epsilon_\theta$ , with parameters  $\theta$ , to minimize the objective

$$\min_{\theta} \mathbb{E}_{t \sim \mathcal{U}(0,1); x_0 \sim q(x); \epsilon \sim \mathcal{N}(0, \mathbf{I})} \left[ \|\epsilon - \epsilon_\theta(\sqrt{\bar{\alpha}_t}x_0 + \sqrt{1 - \bar{\alpha}_t}\epsilon, t)\|^2 \right]. \quad (11)$$

Given the initial noise  $x_T$ , image samples  $x_0$  are obtained by iterating for  $t \in [1, T]$  the denoising update

$$x_{t-1} = \frac{1}{\sqrt{\alpha_t}} \left( x_t - \epsilon_\theta(x_t, t) \times \frac{(1 - \alpha_t)}{\sqrt{1 - \bar{\alpha}_t}} \right) + \sigma_t z, \quad (12)$$

with  $z \sim \mathcal{N}(0, \mathbf{I})$ , and  $\sigma_t^2 = \frac{1 - \bar{\alpha}_{t-1}}{1 - \bar{\alpha}_t} (1 - \alpha_t)$ .

The authors of [16] point out that the quality of generated images improves as the total number of denoising steps  $T$  increases. Thus, the inference loop using eq. (12) becomes computationally expensive. To reduce the computational cost, they propose to use instead  $q(x_{1:T}|x_0) = q(x_T|x_0) \prod_{t=2}^T q(x_{t-1}|x_t, x_0)$ , a Denoising Diffusion Implicit Model (DDIM) [16], which foregoes the Markovian assumption in favor of a diffusion process where  $q(x_T|x_0) = \mathcal{N}\left(\sqrt{\bar{\alpha}_T}x_0, (1 - \bar{\alpha}_T)\mathbf{I}\right)$  and

$$q(x_{t-1}|x_t, x_0) = \mathcal{N}\left(\sqrt{\bar{\alpha}_{t-1}}x_0 + \sqrt{1 - \bar{\alpha}_{t-1} - \sigma_t^2} \cdot \frac{(x_t - \sqrt{\bar{\alpha}_t}x_0)}{\sqrt{1 - \bar{\alpha}_t}}, \sigma_t^2 \mathbf{I}\right). \quad (13)$$

When  $\sigma_t = 0$  for all  $t$ , the diffusion process is **fully deterministic**. This means that when we start from the same noise sample  $x_T$  we obtain the same generated image sample. Given  $x_t$ , one can first predict the denoised observation  $\hat{x}_0$ , which is a prediction of  $x_0$  given  $x_t$

$$\hat{x}_0 = \frac{(x_t - \sqrt{1 - \bar{\alpha}_t}\epsilon_\theta(x_t, t))}{\sqrt{\bar{\alpha}_t}}. \quad (14)$$

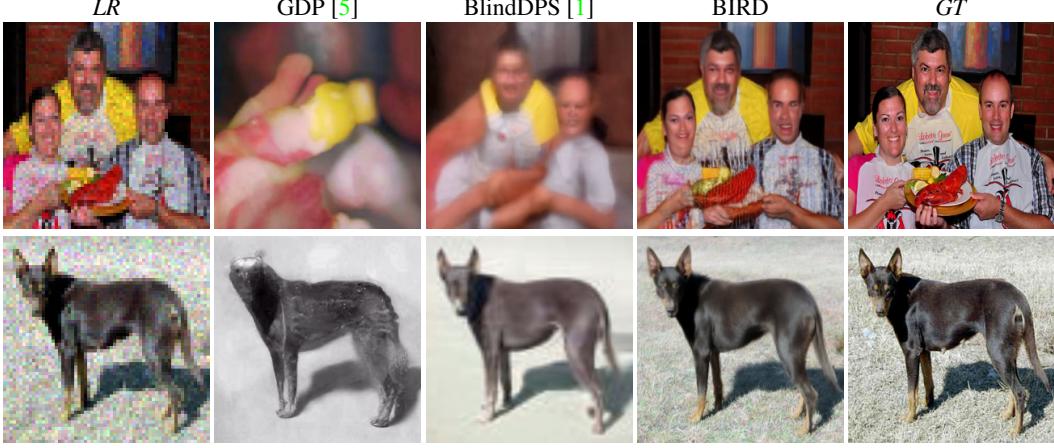


Figure 5: Qualitative comparisons of  $4\times$  SR on ImageNet. Each column shows two examples. From left to right: input low-resolution images, GPD [5], BlindDPS [1], BIRD (our method), and the ground truth high-resolution and noise-free image.

Then, we can predict  $x_{t-1}$  from  $x_t$  and  $\hat{x}_0$  using eq. (13) by setting  $\sigma_t = 0$

$$x_{t-1} = \sqrt{\bar{\alpha}_{t-1}}\hat{x}_0 + \sqrt{1 - \bar{\alpha}_{t-1}}\hat{\omega} \quad (15)$$

with  $\hat{\omega} = \frac{x_t - \sqrt{\bar{\alpha}_t}\hat{x}_0}{\sqrt{1 - \bar{\alpha}_t}}$  a direction pointing to  $x_t$ . [16] shows that this formulation allows DDIM to use fewer time steps at inference by directly predicting  $x_{t-\tau}$  with  $\tau > 1$ , which results in a more computationally efficient process using

$$x_{t-\tau} = \sqrt{\bar{\alpha}_{t-\tau}}\hat{x}_0 + \sqrt{1 - \bar{\alpha}_{t-\tau}}\hat{\omega}. \quad (16)$$

### 3.2.2 Accelerated DDIM Sampling

We explore the advantage of DDIMs of allowing fewer steps than in other diffusion models during the sampling process, as discussed in section 3.2.1. We start from  $x_T \sim \mathcal{N}(0, \mathbf{I})$ . Instead of denoising  $x_T$  iteratively for all the steps used in the pre-training, we make larger steps by using intermediate estimates of  $\hat{x}_{0|t}$  from  $x_t$  using the pre-trained DDIM model  $\epsilon_\theta$  via

$$\hat{x}_{0|t} = \frac{x_t - \sqrt{1 - \bar{\alpha}_t}\epsilon_\theta(x_t, t)}{\sqrt{\bar{\alpha}_t}}. \quad (17)$$

We define the hyper-parameter  $\delta t$  that controls the number of denoising steps, and we can directly jump to estimate  $x_{t-\delta t}$  from  $\hat{x}_{0|t}$  and  $x_t$  using eq. (18) below

$$x_{t-\delta t} = \sqrt{\bar{\alpha}_{t-\delta t}}\hat{x}_{0|t} + \sqrt{1 - \bar{\alpha}_{t-\delta t}} \cdot \frac{x_t - \sqrt{\bar{\alpha}_t}\hat{x}_{0|t}}{\sqrt{1 - \bar{\alpha}_t}}. \quad (18)$$

A larger step size  $\delta t$  allows us to favor speed, while a lower one favors precision or fidelity. This iterative procedure, which we summarize in Algorithm 2, results in an estimate of  $x_0$  that is differentiable in  $x_T$ . We denote by  $\text{DDIMReverse}(\cdot, \delta t)$  the mapping function between  $x_T$  and  $x_0$  that is parameterized with the step size  $\delta t$  (this is essentially our choice of  $g$  in problem (8)). Figure 3 shows sampled images  $x_0$  using different  $\delta t$  starting from the same initial noise  $x_T$  through  $\text{DDIMReverse}(\cdot, \delta t)$ .

### 3.2.3 BIRD: Blind Image Restoration via Fast Diffusion Inversion

BIRD performs an iterative minimization of problem (8) with a running index  $k$ , which we append as a superscript to the estimated unknowns. It starts from an initial noise instance  $x_T^0 \sim \mathcal{N}(0, \mathbf{I})$  and a parametric degradation model  $H_{\eta^0}$ , where  $\eta^0$  is randomly initialized. At each iteration  $k$ , we first compute our estimate of the clean image  $x_0^k$  by mapping  $x_T^k$  through  $\text{DDIMReverse}(\cdot, \delta t)$

$$x_0^k = \text{DDIMReverse}(x_T^k, \delta t). \quad (19)$$

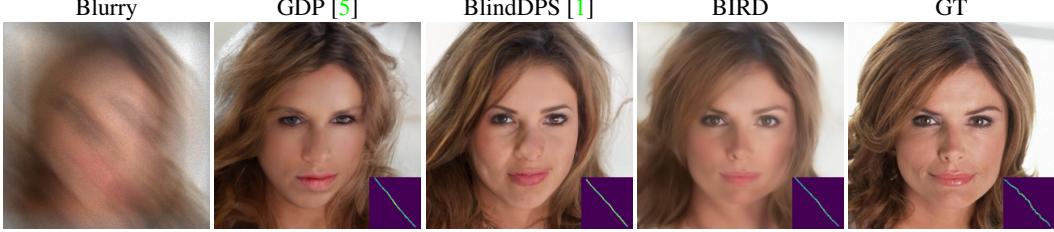


Figure 6: Qualitative comparisons of Gaussian deblurring on CelebA. From left to right: input blurry image, GDP [5], BlindDPS [1], BIRD (our method), and the ground truth sharp image.

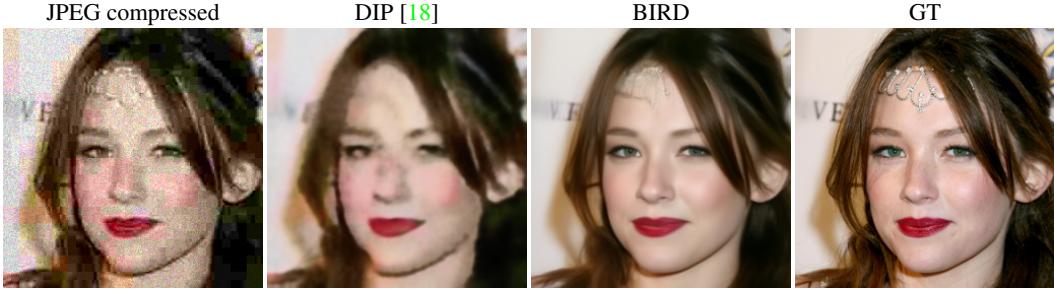


Figure 7: Qualitative comparisons of JPEG de-artifacting on CelebA. From left to right: input JPEG compressed image, GDP [5], BlindDPS [1], BIRD (our method), and the original image.

We then compute the restoration loss  $\mathcal{L}_{IR}$

$$\mathcal{L}_{IR}(x_0^k, H_\eta) = \|y - H_\eta(x_0^k)\|^2. \quad (20)$$

Given an initial noise  $x_T^k$  and  $H_{\theta^k}$ , the optimization iteration is simply derived through gradient descent with a learning rate  $\alpha$ .

$$x_T^{k+1} = x_T^k - \alpha \nabla_{x_T} \mathcal{L}_{IR}(x_0^k, H_{\eta^k}) \quad \text{and} \quad \eta^{k+1} = \eta^k - \alpha \nabla_\eta \mathcal{L}_{IR}(x_0^k, H_{\eta^k}). \quad (21)$$

To impose the normalization constraint on  $x_T$  we set its Euclidean norm to  $\|x_T\| = \sqrt{N_x \times M_x}$  via

$$x_T^{k+1} = \frac{x_T^{k+1}}{\|x_T^{k+1}\|} \sqrt{N_x \times M_x}. \quad (22)$$

After  $N$  iterations, or when  $\mathcal{L}_{IR}(x_0^N, H_{\theta^N})$  becomes small enough, we deem that BIRD has converged. The restored image  $\hat{x}_0$  is then generated by  $\hat{x}_0 = \text{DDIMReverse}(x_T^N, \delta t)$ , which always ensures that  $\hat{x}_0$  is within the manifold of realistic images, as already shown in Figure 3. Moreover, in Figure 4 we show two examples of the reconstruction with  $H_\eta(x) = x$  and with  $y$  un-corrupted noise-free images from the validation dataset of ImageNet [4] and CelebA [10].

## 4 Experiments

**Image Restoration Tasks** We showcase BIRD on four different image restoration tasks: Gaussian and motion deblurring, JPEG de-artifacting, and image super-resolution (SR). For Gaussian deblurring, we use an anisotropic Gaussian kernel. We follow the procedure described in [15] and use the same feed-forward network for the kernel estimation. For all tasks, we note that we consider the noisy case with a noise standard deviation of  $\sigma \approx 0.05$ . We introduce a combination of Gaussian (signal-independent) and speckle (signal-dependent) noises simulating common sources of noise in cameras, namely shot noise and dead (or hot) pixels respectively. For all our experiments, we use  $\delta t = 100$  and a maximum number of iterations  $N = 200$ . Adam is adopted as an optimizer with a 0.003 learning rate. Our code is provided in the supplementary material. We evaluate our method both on the validation datasets of ImageNet [4] and CelebA [10] at  $256 \times 256$  pixel resolution. We follow [2, 3] and report the obtained LPIPS [25] and PSNR for each experiment.

Table 1: Quantitative evaluation of several inverse problems on the CelebA validation dataset. The best and second best methods are indicated in bold and underlined respectively.

Method	Motion Deblur		Gaussian Deblur		$8 \times$ SR		De-artifacting	
	PSNR $\uparrow$	LPIPS $\downarrow$						
BlindDPS [1]	<u>23.15</u>	<u>0.281</u>	<u>23.56</u>	<u>0.257</u>	<u>21.82</u>	<u>0.345</u>	-	-
DIP [18]	-	-	-	-	18.64	0.415	24.51	0.284
DGPGAN [13]	22.23	0.304	20.65	0.378	19.83	0.372	-	-
GDP [5]	22.49	0.314	22.53	0.304	20.78	0.357	-	-
BIRD	<b>23.76</b>	<b>0.263</b>	<b>24.67</b>	<b>0.225</b>	<b>22.75</b>	<b>0.306</b>	<b>28.67</b>	<b>0.224</b>

Table 2: Runtime (in seconds per processed image) comparison of blind deblurring methods on CelebA.

Method	Time[s]	Memory[GB]
DGPGAN [13]	190	0.9
GDP [5]	118	1.1
<i>Naive Inversion</i>	5700	2.2
BlindDPS [1]	270	6.1
BIRD	234	1.2

Table 3: Effect of the step size  $\delta t$  on the visual quality and the runtime of De-artifacting on CelebA.

Step size ( $\delta t$ )	PSNR $\uparrow$	LPIPS $\downarrow$	Time[s] $\downarrow$
50	28.74	0.218	412
100	28.67	0.224	234
200	28.45	0.237	110

Table 4: Quantitative evaluation of several inverse problems on the ImageNet validation dataset. The best and second best methods are indicated in bold and underlined respectively.

Method	Gaussian Deblur		$4 \times$ SR	
	PSNR $\uparrow$	LPIPS $\downarrow$	PSNR $\uparrow$	LPIPS $\downarrow$
BlindDPS [1]	<b>24.28</b>	<b>0.296</b>	<u>21.36</u>	<u>0.374</u>
DIP [18]	-	-	18.67	0.456
DGPGAN [13]	21.67	0.384	19.58	0.461
GDP [5]	22.45	0.368	20.24	0.435
BIRD	<u>23.76</u>	<u>0.321</u>	<b>22.15</b>	<b>0.354</b>

**Qualitative Results** Figures 5, 6, 7 show non cherry-picked examples of ImageNet and CelebA images restored in the cases of  $4 \times$  super-resolution, Gaussian deblurring and JPEG de-artifacting. We also compare our work with GDP [5] and BlindDPS [1]. BIRD seems to recover always more realistic, consistent and accurate images than the other competing methods.

**Quantitative Results** Tables 1 and 4 report the performance of BIRD as well as DIP [18], DGPGAN [13], DPS [2], GDP [5] for CelebA and ImageNet, respectively. For both datasets, we observe that our method achieves state of the art results both in terms of PSNR and LPIPS metrics.

**Ablations** We also compare the runtime performance of BIRD to other state-of-the-art methods and evaluate the impact of the step size  $\delta t$ . All experiments are carried out on a GeForce GTX 1080 Ti.

**Effect of the step size  $\delta t$ :** The step size  $\delta t$  defines a trade-off between speed and accuracy. Table 3 shows the PSNR, LPIPS and runtime for image denoising on CelebA. We observe that the reconstruction accuracy of BIRD is not too sensitive to this parameter, while the runtime is linearly affected.  $\delta t = 100$  gives a good trade-off between the speed and the reconstruction accuracy.

**Memory usage and Computational Cost.** In Table 2, we report the memory usage and the runtime per image of different blind deblurring methods. We also show as a reference a *Naive Inversion*, which performs the inversion by unrolling all the steps of the pre-trained diffusion model. This case shows how BIRD results in a significantly faster execution. Despite being an iterative method, BIRD has a reasonable runtime making it a practical IR method. The reported speed uses  $\delta t = 100$ .

**Limitations and Broader Impacts** We have evaluated our method on degradation operators with a given explicit parametrized form (albeit with unknown parameters). It would be interesting to further investigate the use of BIRD on problems where an explicit form of the degradation operator is difficult to have, such as in JPEG compression, or with image deraining and dehazing. In terms of broader impacts, BIRD leverages pre-trained denoising diffusion models with both their advantages and pitfalls. It takes advantage of their strong prior, but may also be affected by their data-induced biases.

## 5 Conclusion

We have introduced BIRD, a novel robust, accurate and fast framework for solving general blind image restoration tasks by using pre-trained diffusion generative models as learned priors. Our method exploits the deterministic correspondence between noise and images in DDIM by casting the inverse restoration problem as a latent estimation problem. Our framework does not require training networks on specific task, but can instead be directly applied to new images and new degradation

models. We leverage the capability of DDIM to skip ahead in the forward diffusion process and provide an efficient diffusion inversion in the context of image restoration. We demonstrate that BIRD achieves state of the art performance on blind image restoration tasks including Gaussian deblurring, motion deblurring, super-resolution and JPEG de-artifacting.

## References

- [1] Chung, H., Kim, J., Kim, S., Ye, J.C.: Parallel diffusion models of operator and image for blind inverse problems. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 6059–6069 (2023) [1](#), [2](#), [3](#), [7](#), [8](#), [9](#)
- [2] Chung, H., Kim, J., Mccann, M.T., Klasky, M.L., Ye, J.C.: Diffusion posterior sampling for general noisy inverse problems. arXiv preprint arXiv:2209.14687 (2022) [1](#), [3](#), [8](#), [9](#)
- [3] Chung, H., Sim, B., Ryu, D., Ye, J.C.: Improving diffusion models for inverse problems using manifold constraints. In: Oh, A.H., Agarwal, A., Belgrave, D., Cho, K. (eds.) Advances in Neural Information Processing Systems (2022), <https://openreview.net/forum?id=nJJjv0JDJju> [1](#), [3](#), [8](#)
- [4] Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L.: Imagenet: A large-scale hierarchical image database. In: 2009 IEEE conference on computer vision and pattern recognition. pp. 248–255. Ieee (2009) [8](#)
- [5] Fei, B., Lyu, Z., Pan, L., Zhang, J., Yang, W., Luo, T., Zhang, B., Dai, B.: Generative diffusion prior for unified image restoration and enhancement. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 9935–9946 (2023) [3](#), [7](#), [8](#), [9](#)
- [6] Ho, J., Jain, A., Abbeel, P.: Denoising diffusion probabilistic models. Advances in Neural Information Processing Systems **33**, 6840–6851 (2020) [1](#), [6](#)
- [7] Karras, T., Aittala, M., Laine, S., Härkönen, E., Hellsten, J., Lehtinen, J., Aila, T.: Alias-free generative adversarial networks. In: Proc. NeurIPS (2021) [1](#)
- [8] Kawar, B., Elad, M., Ermon, S., Song, J.: Denoising diffusion restoration models. arXiv preprint arXiv:2201.11793 (2022) [1](#), [6](#)
- [9] Kobyzev, I., Prince, S.J., Brubaker, M.A.: Normalizing flows: An introduction and review of current methods. IEEE transactions on pattern analysis and machine intelligence **43**(11), 3964–3979 (2020) [5](#)
- [10] Liu, Z., Luo, P., Wang, X., Tang, X.: Large-scale celebfaces attributes (celeba) dataset. Retrieved August **15**(2018), 11 (2018) [8](#)
- [11] Lugmayr, A., Danelljan, M., Romero, A., Yu, F., Timofte, R., Van Gool, L.: Repaint: Inpainting using denoising diffusion probabilistic models. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 11461–11471 (2022) [1](#), [6](#)
- [12] Menon, S., Damian, A., Hu, S., Ravi, N., Rudin, C.: Pulse: Self-supervised photo upsampling via latent space exploration of generative models. In: Proceedings of the ieee/cvf conference on computer vision and pattern recognition. pp. 2437–2445 (2020) [5](#)
- [13] Pan, X., Zhan, X., Dai, B., Lin, D., Loy, C.C., Luo, P.: Exploiting deep generative prior for versatile image restoration and manipulation. IEEE Transactions on Pattern Analysis and Machine Intelligence **44**(11), 7474–7489 (2021) [3](#), [9](#)
- [14] Poirier-Ginter, Y., Lalonde, J.F.: Robust unsupervised stylegan image restoration. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 22292–22301 (2023) [3](#)
- [15] Ren, D., Zhang, K., Wang, Q., Hu, Q., Zuo, W.: Neural blind deconvolution using deep priors. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 3341–3350 (2020) [8](#)

- [16] Song, J., Meng, C., Ermon, S.: Denoising diffusion implicit models. arXiv preprint arXiv:2010.02502 (2020) [2](#), [5](#), [6](#), [7](#)
- [17] Song, Y., Ermon, S.: Generative modeling by estimating gradients of the data distribution. Advances in neural information processing systems **32** (2019) [1](#)
- [18] Ulyanov, D., Vedaldi, A., Lempitsky, V.: Deep image prior. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 9446–9454 (2018) [8](#), [9](#)
- [19] Vershynin, R.: Random vectors in high dimensions. Cambridge Series in Statistical and Probabilistic Mathematics. Cambridge University Press **3**, 38–69 (2018) [5](#)
- [20] Wang, Y., Yu, J., Zhang, J.: Zero-shot image restoration using denoising diffusion null-space model. The Eleventh International Conference on Learning Representations (2023) [1](#), [3](#), [6](#)
- [21] Yang, T., Ren, P., Xie, X., Zhang, L.: Gan prior embedded network for blind face restoration in the wild. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 672–681 (2021) [3](#)
- [22] Yu, Y., Zhang, L., Fan, H., Luo, T.: High-fidelity image inpainting with gan inversion. In: European Conference on Computer Vision. pp. 242–258. Springer (2022) [3](#)
- [23] Zhang, B., Gu, S., Zhang, B., Bao, J., Chen, D., Wen, F., Wang, Y., Guo, B.: Styleswin: Transformer-based gan for high-resolution image generation. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 11304–11314 (2022) [1](#)
- [24] Zhang, K., Li, Y., Zuo, W., Zhang, L., Gool, L.V., Timofte, R.: Plug-and-play image restoration with deep denoiser prior. IEEE Transactions on Pattern Analysis and Machine Intelligence **44**, 6360–6376 (2020) [4](#)
- [25] Zhang, R., Isola, P., Efros, A.A., Shechtman, E., Wang, O.: The unreasonable effectiveness of deep features as a perceptual metric. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 586–595 (2018) [8](#)