

LATENT GENERATIVE MODELS WITH TUNABLE COMPLEXITY FOR COMPRESSED SENSING AND OTHER INVERSE PROBLEMS

Anonymous authors

Paper under double-blind review

ABSTRACT

Generative models have emerged as powerful priors for solving inverse problems. These models typically represent a class of natural signals using a single fixed complexity or dimensionality. This can be limiting: depending on the problem, a fixed complexity may result in high representation error if too small, or overfitting to noise if too large. We develop tunable-complexity priors for diffusion models, normalizing flows, and variational autoencoders, leveraging nested dropout. Across tasks including compressed sensing, inpainting, denoising, and phase retrieval, we show empirically that tunable priors consistently achieve lower reconstruction errors than fixed-complexity baselines. In the linear denoising setting, we provide a theoretical analysis that explicitly characterizes how the optimal tuning parameter depends on noise and model structure. This work demonstrates the potential of tunable-complexity generative priors and motivates both the development of supporting theory and their application across a wide range of inverse problems.

1 Introduction

Inverse problems aim to reconstruct an unknown signal, potentially corrupted by noise, from a set of measurements given by a forward model, which may or may not be known in advance. Such a formulation applies to various image-processing applications, including compressive sensing, denoising, and super-resolution. In practice, inverse problems are ill-posed and thus they require prior information about the signal to yield a successful recovery (Scarlett et al., 2022).

Deep generative models have been demonstrated to be powerful signal priors when used to solve inverse problems (Bora et al., 2017; Hand et al., 2018; Asim et al., 2020; Daras et al., 2021; 2022; Song et al., 2022). The process of using a deep generative prior typically has two phases: training and inversion. In the training phase, a generative neural network is trained on a dataset representative of the natural signal class intended for inversion. In the second phase, the model parameters are fixed from training, and an algorithm is deployed to estimate the signal of interest for a given forward operator. This approach has the benefit that the prior can be learned in isolation from the inverse problems being solved. Thus, the approach can apply to a variety of inverse problems, which is in contrast to other neural network-based approaches such as end-to-end training.

In recent years, there has been a dominant framework for using generative models as priors for inverse problems. Within this framework, generative priors have a fixed complexity that is set during training. For example, existing priors include Generative Adversarial Networks (GANs), which typically have a latent space of fixed low dimensionality (Bora et al. (2017); Hand et al. (2018); Menon et al. (2020); Cocola et al. (2020); Hand & Joshi (2019)); Normalizing Flow models, which have a latent space of fixed high-dimensionality equal to that of the images (Asim et al. (2020); Whang et al. (2021); Ardizzone et al. (2019); Liu et al. (2023)); and Score-Based Models, which maintain information about the probability density over the fixed high-dimensional space of all images (Song et al. (2022); Meng & Kabashima (2023); Jalal et al. (2021); Dou & Song (2024); Chung et al. (2023)).

We step outside the dominant framework and demonstrate the benefits of solving inverse problems using generative priors whose complexity can be selected by the user at inference time, after training. Such generative priors simultaneously maintain representations of varying complexities of the natural

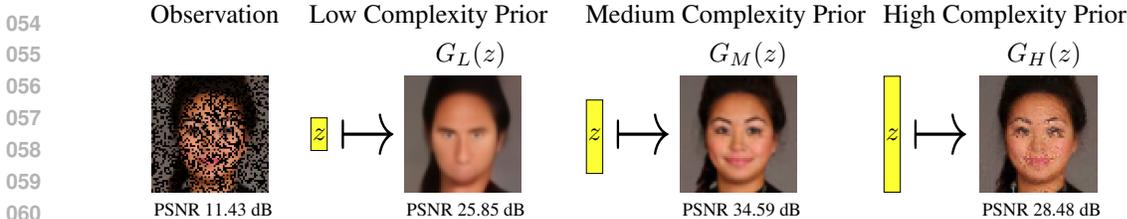


Figure 1: For visualization purposes, consider an inpainting problem with randomly missing pixels. We trained three separate generative models: one with low, medium, and high latent dimensionality as depicted by the size of the boxes representing z . The medium-complexity prior yields the reconstruction with the highest Peak Signal-to-Noise Ratio (PSNR).

signal class, and we refer to them as **generative priors with a tunable complexity**. In this paper, we demonstrate that tunable generative priors can lead to significantly improved reconstruction errors when model complexity is appropriately tuned for a specific inverse problem. We focus on latent generative models whose complexity is governed by the latent dimension k . By training the model to preserve meaningful representations across different values of k , we enable practitioners to tune its complexity to the inverse problem at hand. We illustrate this point in Figure 1 for a random pixel inpainting problem. The medium-complexity prior yields a higher quality image reconstruction than the low- and high-complexity prior, both qualitatively and quantitatively.

To motivate our approach, we first consider injective flows (Ross & Cresswell, 2021). We train a family of models, each with a distinct latent dimensionality k and no parameter sharing, and apply them to compressed sensing with random measurements. As shown in Figure 2, reconstruction error follows an upside-down U-shaped curve: with fewer measurements, models of intermediate complexity outperform both higher- and lower-complexity ones. Moreover, the optimal complexity depends on the number of available measurements. While this experiment requires training separate models for each k , a naive and computationally expensive strategy, it illustrates the importance of tunability. This motivates the more efficient algorithms we develop next, which scale to practical settings such as production-size images.

Building on this motivation, we will show how tunability can be achieved efficiently in three major classes of generative models: variational autoencoders (VAEs) (Kingma & Welling, 2014), normalizing flows (NFs) (Durkan et al., 2019), and latent diffusion models (LDMs) (Rombach et al., 2022). In each case, we design a single family of models with parameter sharing and provide an efficient training procedure. For LDMs, we introduce a new algorithm based on nested dropout (Rippel et al., 2014), described in Section 3.1. For NFs, we adopt an existing ordering method (Bekasov & Murray, 2020) Appendix B. For VAEs, we extend the adversarial objective of Esser et al. (2021) by adding a regularization term. Across all three settings, we empirically demonstrate that tunable generative priors consistently achieve lower reconstruction errors than fixed-complexity baselines over a range of inverse problems, undersampling ratios, and noise levels (Figure 4).

As an initial theoretical exploration into the effect of model tunability, we study denoising in the case of an invertible linear generative model and the best lower-dimensional linear models that approximate it. We rigorously establish a theory for how to select the tuning parameter in the simple case of a linear generative model. In this setting, we provide an explicit expression for the reconstruction error as a function of the modeled dimensionality in the cases of maximum likelihood estimation and maximum *a posteriori* estimation. This analytical expression permits a direct expression for optimal signal complexity in this setting, revealing the theoretical benefit for tunability.

The results in this paper demonstrate the benefits of using generative priors with a tunable complexity for inverse problems both empirically and theoretically. We empirically demonstrate the affects of tunable complexity on using families of generative models and different model architectures and inversion algorithms. Additionally, for linear models, we provide a theoretical analysis of denoising. This work motivates research on how to bring tunability into generative models.

108 **Our main contributions are as follows:**

- 109
- 110 • We observe a potentially surprising phenomenon in the use of latent generative priors for
- 111 solving inverse problems. We train a single generative model to simultaneously represent
- 112 the natural signal class across multiple latent dimensionalities k . Across a variety of
- 113 architectures, inverse problems, and inversion algorithms, we empirically find that some
- 114 intermediate latent dimensionality yields an improved reconstruction error.
- 115 • We propose a new training algorithm for latent diffusion that utilizes nested dropout and
- 116 a convex combination of the original and truncated latent objective. This yields a tunable
- 117 latent diffusion model that learns hierarchical representations across latent dimensionalities,
- 118 enabling a single model to be tuned as a prior for downstream inverse problems.
- 119 • We provide a theoretical analysis of tunability in the context of denoising with linear
- 120 invertible generative models. Under this model, we derive an explicit expression for the
- 121 reconstruction error of MLE and MAP estimators as a function of the model complexity.

122 2 Background and Related Work

123 The background will focus on diffusion models due to their relevance in recent research trends.

124 For additional details on normalizing flows and variational autoencoders, we refer the reader to the

125 appropriate references.

126 2.1 Diffusion Models

127 The standard diffusion framework gradually corrupts data $\mathbf{x}_0 \sim p_{\text{data}}$, $\mathbf{x} \in \mathbb{R}^n$ into noise $\mathbf{x}_T \sim$

128 $\mathcal{N}(0, \mathbf{I})$ via a forward noising process, and learns a parameterization of the reverse process to recover

129 the data distribution (Ho et al., 2020; Song et al., 2021b; Dhariwal & Nichol, 2021). We adopt latent

130 diffusion models (LDMs) (Rombach et al., 2022), which improve efficiency by running diffusion in

131 a learned latent space: a *variational autoencoder* (VAE) (Kingma & Welling, 2014) with encoder

132 $\mathcal{E} : \mathbb{R}^n \rightarrow \mathbb{R}^k$ and decoder $\mathcal{D} : \mathbb{R}^k \rightarrow \mathbb{R}^n$ (typically $k \ll n$) yields $\mathbf{z}_0 = \mathcal{E}(\mathbf{x}_0)$ and $\mathbf{x}_0 \approx \mathcal{D}(\mathbf{z}_0)$ for all

133 $\mathbf{x}_0 \sim p_{\text{data}}$. The encoder pushes p_{data} to $p_Z := \mathcal{E}_{\#} p_{\text{data}}$, and diffusion operates on \mathbf{z}_t .

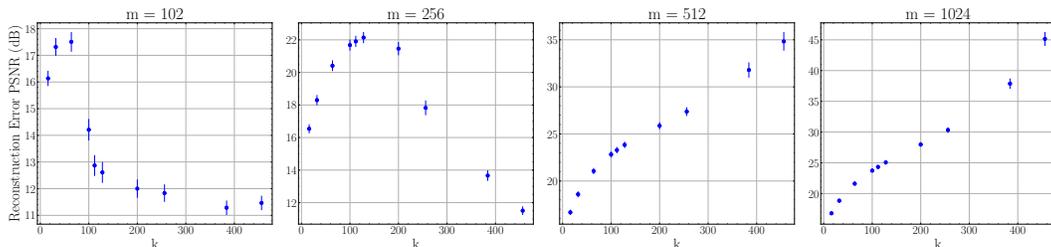
134 With schedule $\{\beta_t\}_{t=1}^T$, $\alpha_t := 1 - \beta_t$, and $\bar{\alpha}_t := \prod_{j=1}^t \alpha_j$, the forward marginal in latent space is

$$135 \mathbf{z}_t = \sqrt{\bar{\alpha}_t} \mathbf{z}_0 + \sqrt{1 - \bar{\alpha}_t} \boldsymbol{\epsilon}, \quad \boldsymbol{\epsilon} \sim \mathcal{N}(0, \mathbf{I}), \quad (\text{forward marginal})$$

136 which allows sampling \mathbf{z}_t at any timestep t directly from \mathbf{z}_0 . The reverse process is parameterized by

137 a neural network:

$$138 \mathbf{z}_{t-1} = \frac{1}{\sqrt{\alpha_t}} \left(\mathbf{z}_t - \frac{1 - \alpha_t}{\sqrt{1 - \bar{\alpha}_t}} \boldsymbol{\epsilon}_{\theta}(\mathbf{z}_t, t) \right) + \sigma_t \boldsymbol{\epsilon}'_t, \quad \boldsymbol{\epsilon}'_t \sim \mathcal{N}(0, \mathbf{I}), \quad (\text{reverse})$$



158 Figure 2: In this brute-force approach, we employ an injective flow prior trained separately on

159 MNIST, where each image has size $n = 32 \times 32 = 1024$ pixels and the latent dimensionality k varies

160 from 16 to 456. For compressed sensing with random measurements, intermediate latent dimensions

161 ($150 \leq k \leq 300$) yield the lowest reconstruction error at small measurement ratios m/n .

where $\sigma_t \in [0, \sigma_t^{\text{DDPM}}]$ controls sampling stochasticity and $\sigma_t^{\text{DDPM}} = \sqrt{\frac{1-\bar{\alpha}_t-1}{1-\bar{\alpha}_t}} (1-\alpha_t)$. Setting $\sigma_t = 0$ yields the deterministic denoising diffusion implicit model (DDIM) sampler (Song et al., 2021a).

A denoising network ϵ_θ (typically a U-Net) is trained to predict the noise in the latent space:

$$\mathcal{L}_{\text{LDM}} = \mathbb{E}_{t \sim \mathcal{U}(\{1, \dots, T\}), \mathbf{z}_0, \epsilon \sim \mathcal{N}(0, \mathbf{I})} \|\epsilon - \epsilon_\theta(\sqrt{\bar{\alpha}_t} \mathbf{z}_0 + \sqrt{1-\bar{\alpha}_t} \epsilon, t)\|_2^2, \quad (1)$$

where $t \sim \mathcal{U}(\{1, \dots, T\})$ denotes that the timestep is sampled uniformly from all diffusion steps.

To enable latent diffusion, the autoencoder must be pretrained; its objective combines reconstruction, regularization, and adversarial terms. The primary term is the reconstruction loss, $\mathcal{L}_{\text{recon}}$, typically a combination of ℓ_1 and perceptual losses, which measures the discrepancy between the original high-dimensional signal \mathbf{x} and its reconstruction $\mathcal{D}(\mathcal{E}(\mathbf{x}))$. A regularization penalty, \mathcal{L}_{reg} , is enforced via Kullback–Leibler (KL) divergence with respect to a reference distribution, typically the standard Gaussian. Finally, an adversarial loss, \mathcal{L}_{adv} , is introduced by training a discriminator \mathcal{C} to distinguish between real images \mathbf{x} and reconstructed samples $\mathcal{D}(\mathcal{E}(\mathbf{x}))$

$$\mathcal{L}_{\text{Autoencoder}} = \mathcal{L}_{\text{rec}}(\mathbf{x}, \mathcal{D}(\mathcal{E}(\mathbf{x}))) + \lambda_{\text{reg}} \mathcal{L}_{\text{reg}}(\mathcal{E}(\mathbf{x})) + \lambda_{\text{adv}} \mathcal{L}_{\text{adv}}(\mathcal{C}, \mathbf{x}, \mathcal{E}(\mathbf{x})). \quad (2)$$

After this stage, diffusion is performed entirely on \mathbf{z}_t .

2.2 Inverse Problems with Generative Priors

We consider the general linear inverse problem of recovering an unknown signal $\mathbf{x} \in \mathbb{R}^n$ from noisy measurements $\mathbf{y} \in \mathbb{R}^m$,

$$\mathbf{y} = \mathcal{A}(\mathbf{x}) + \boldsymbol{\eta}, \quad \boldsymbol{\eta} \sim \mathcal{N}(0, \boldsymbol{\sigma}^2 \mathbf{I}_m), \quad (3)$$

where $\mathcal{A} : \mathbb{R}^n \rightarrow \mathbb{R}^m$ is the forward operator (e.g., a linear projection, convolutional blur, or other transformation) (Scarlett et al., 2022), and $\boldsymbol{\eta}$ is additive noise drawn from a Gaussian distribution. Generative models can be employed as priors for inverse problems in two main ways: supervised, where the forward operator is known during training and models are trained on paired data $\{(x_i, y_i)\}$, or unsupervised, where the forward operator is unknown (Scarlett et al., 2022; Li et al., 2025).

For inverse problems, we modify the reverse dynamics in Equation (reverse) to account for measurements $\mathbf{y} = \mathcal{A}(\mathbf{x}) + \boldsymbol{\eta}$. Since we operate entirely in latent space, posterior sampling is guided by the conditional score, which decomposes as

$$\nabla_{\mathbf{z}_t} \log p_t(\mathbf{z}_t | \mathbf{y}) = \nabla_{\mathbf{z}_t} \log p_t(\mathbf{z}_t) + \nabla_{\mathbf{z}_t} \log p_t(\mathbf{y} | \mathbf{z}_t). \quad (4)$$

The first term is provided by the pretrained diffusion prior through the denoising network ϵ_θ , while the second term enforces data-consistency with the forward operator \mathcal{A} under the measurement model.

Following the taxonomy of interleaving methods highlighted by Wang et al. (2024), these solvers alternate unconditional reverse diffusion steps (e.g., DDIM) with data-consistency corrections that either explicitly approximate the measurement likelihood or enforce feasibility. One line of methods explicitly approximates the measurement-likelihood term via a projection or gradient update (Jalal et al., 2021; Kawar et al., 2021; Chung et al., 2023; Wang et al., 2023; Rout et al., 2024). A canonical formulation is latent diffusion posterior sampling (Rout et al., 2023):

$$\nabla_{\mathbf{z}_t} \log p(\mathbf{y} | \mathbf{z}_t) \approx \nabla_{\mathbf{z}_t} \log p(\mathbf{y} | \mathbf{x}_0 = \mathcal{D}(\mathbb{E}[z_0 | z_t])) = \frac{1}{\sigma^2} \nabla_{\mathbf{z}_t} \|\mathbf{y} - \mathcal{D}(\mathbf{z}_0)\|_2^2, \quad (5)$$

where the posterior mean is given by Tweedie’s formula (Efron, 2011).

Another class of methods does not explicitly compute the measurement likelihood, but instead approximates the feasible set of solutions $\{\mathbf{x} | \mathbf{y} = \mathcal{A}(\mathbf{x})\}$ (Kawar et al., 2021; 2022; Song et al., 2024). In particular Song et al. (2024) employs a hard data-consistency term that enforces $\mathcal{D}(\mathbf{z}) \in \{\mathbf{x} | \mathbf{y} = \mathcal{A}(\mathbf{x})\}$, which is approximated in practice by gradient-descent.

3 Methods

This section describes the training of a tunable latent diffusion model (Section 3.1) and its application as a prior for inverse problems (Section 3.1). We begin by training a variational autoencoder (VAE)

as the backbone of the latent diffusion model. The VAE consists of an encoder $\mathcal{E} : \mathbb{R}^n \rightarrow \mathbb{R}^k$ and a decoder $\mathcal{D} : \mathbb{R}^k \rightarrow \mathbb{R}^n$, such that for samples $\mathbf{x}_0 \sim p_{\text{data}}(\mathbf{x}_0)$ we have $\mathbf{x}_0 \approx \mathcal{D}(\mathcal{E}(\mathbf{x}_0))$. Once trained the autoencoder, a denoising network $\epsilon_\theta(\mathbf{z}_t, t)$ is trained in latent space to predict the Gaussian noise at each diffusion step t . Finally, diffusion is performed in latent space using the denoising network ϵ_θ .

3.1 Training a Tunable Latent Diffusion

We aim to train a latent generative model that can represent the natural signal class across multiple latent dimensionalities. To achieve this, we leverage nested dropout (Rippel et al., 2014), which imposes an ordered structure on the latent variables by always preserving a prefix of coordinates. Formally, let $k \sim p_k$ be drawn from $\{1, \dots, d\}$. In our experiments, we use a truncated geometric distribution with success parameter p , but other distribution over $\{1, \dots, d\}$ can be adopted depending on the application. The truncation operator is defined as $\mathbf{z}_{\downarrow k} = [z_1, z_2, \dots, z_k, 0, \dots, 0]$ with $k \leq d$. Given $\mathbf{z} = \mathcal{E}(\mathbf{x})$, reconstruction is performed from $\mathbf{z}_{\downarrow k}$, which encourages earlier coordinates to carry more information about the signal class.

Latent diffusion is trained in two stages. First, the autoencoder is trained to reconstruct $\mathbf{x} \in \mathbb{R}^n$ robustly across a range of latent dimensions k .

Building on Section 2.1, the VAE backbone is trained with reconstruction, regularization, and adversarial terms, which we extend here with a nested dropout objective. This follows the standard VAE training objective with an added nested dropout term:

$$\mathcal{L}_{\text{VAE}} = \min_{E, D} \max_C \left[\mathcal{L}_{\text{rec}}(\mathbf{x}, \mathcal{D}(\mathcal{E}(\mathbf{x}))) + \lambda_{\text{reg}} \mathcal{L}_{\text{reg}}(\mathcal{E}(\mathbf{x})) + \lambda_{\text{adv}} \mathcal{L}_{\text{adv}}(C, \mathbf{x}, \mathcal{E}(\mathbf{x})) + \lambda_{\text{drop}} \mathcal{L}_{\text{drop}}(\mathbf{x}, \mathcal{D}(\mathcal{E}(\mathbf{x})_{\downarrow k})) \right], \quad (6)$$

where $C : \mathbb{R}^n \rightarrow (0, 1)$ is a discriminator. We adopt a perceptual loss (Zhang et al., 2018) for $\mathcal{L}_{\text{drop}}$.

In the second stage, the diffusion model is trained in latent space with a loss that interpolates between the standard diffusion objective and its truncated-latent variant. For $\lambda \in [0, 1]$, we define

$$\mathcal{L}_{\text{LDM}} = \mathbb{E}_{\mathcal{E}(\mathbf{x}), \epsilon \sim \mathcal{N}(0,1), t \sim \mathcal{U}(\{1, \dots, T\})} \left[(1 - \lambda) \|\epsilon - \epsilon_\theta(\mathbf{z}_t, t)\|_2^2 + \lambda \|\epsilon - \epsilon_\theta((\mathbf{z}_t)_{\downarrow k}, t)\|_2^2 \right]. \quad (7)$$

The first term is the standard latent diffusion objective, while the second applies it to the truncated latent $(\mathbf{z}_t)_{\downarrow k}$, encouraging effective denoising even from a reduced representation. Here, \mathbf{z}_t is sampled from the forward DDPM process (Equation (forward marginal)), and k is drawn from the same distribution p_k used in the VAE objective, ensuring consistency. This encourages the latent space to organize hierarchically: early coordinates contain coarse information preserved under all truncation levels, while later coordinates refine the representation when higher dimensions are available. This effect is illustrated in Figure 3, which shows how FID varies with latent dimensionality under different λ values.

For details about training please go to Appendix A.2. For details on tunable normalizing flows and their use as priors for inverse problems, see Appendix B.

3.2 Inverse Problems with Tunable Priors

A broad family of methods applies a data-consistency step via projection, gradient, or small optimization after each reverse update to move the prior iterate toward the feasible set $\{x \mid \mathcal{A}(x) = y\}$. In

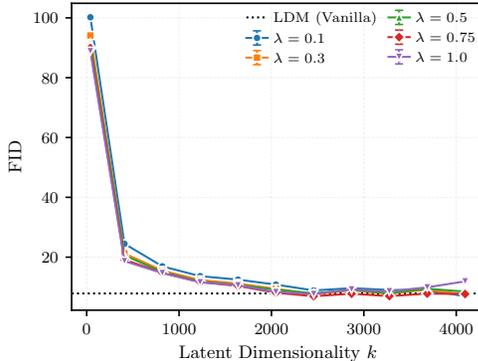


Figure 3: FID as a function of latent dimensionality, evaluated on 50k training images from CelebA. FID score computed with (Parmar et al., 2022).

Algorithm 1 General Template for Tunable Diffusion Priors

```

1: Input:  $\mathbf{y}, \mathcal{A}, \mathcal{E}, \mathcal{D}, \epsilon_\theta$ , steps  $T$ , tunable
   parameter  $k$ 
2: Output:  $\mathcal{D}(\hat{z}_0)$ 
3:  $z_T \sim \mathcal{N}(0, I)$ 
4: for  $t = T - 1$  to  $0$  do
5:    $\hat{s} \leftarrow \epsilon_\theta(z_t, t)$ 
6:    $\hat{z}_0 \leftarrow \frac{1}{\sqrt{\bar{\alpha}_t}}(z_t + \sqrt{1 - \bar{\alpha}_t}\hat{s})$ 
7:    $z'_{t-1} \leftarrow$  DDIM/DDPM reverse with
8:      $\hat{z}_0, \hat{s}$ 
9:    $z_{t-1} \leftarrow$  project/gradient update with
10:     $\hat{z}_0$  and  $z'_{t-1}$  to get closer to
11:     $\{z \mid \mathcal{A}(\mathcal{D}(z)) = \mathbf{y}\}$ 
12:    $z_{t-1} \leftarrow (z_{t-1})_{\downarrow k}$ 
13: end for
14: return  $\mathcal{D}(\hat{z}_0)$ 

```

Algorithm 2 Tunable Posterior Sampling (Concrete Instantiation)

```

1: Input:  $\mathbf{y}, \mathcal{A}, \mathcal{E}, \mathcal{D}, \epsilon_\theta$ , steps  $T$ , variances
    $\{\sigma_t\}$ , tunable parameter  $k$ 
2: Output:  $\mathcal{D}(\hat{z}_0)$ 
3:  $z_T \sim \mathcal{N}(0, I)$ 
4: for  $t = T - 1$  to  $0$  do
5:    $\hat{s} \leftarrow \epsilon_\theta(z_t, t)$ 
6:    $\hat{z}_0 \leftarrow \frac{1}{\sqrt{\bar{\alpha}_t}}(z_t + \sqrt{1 - \bar{\alpha}_t}\hat{s})$ 
7:    $\epsilon \sim \mathcal{N}(0, I)$ 
8:
9:    $z'_{t-1} \leftarrow \frac{\sqrt{\alpha_t(1-\bar{\alpha}_{t-1})}}{1-\bar{\alpha}_t}z_t + \frac{\sqrt{\bar{\alpha}_{t-1}\beta_t}}{1-\bar{\alpha}_t}\hat{z}_0$ 
10:   $+ \sigma_t\epsilon$ 
11:  Initialize  $z$  at  $\hat{z}_0$ 
12:   $z_{t-1} \leftarrow \arg \min_z \|\mathbf{y} - \mathcal{A}(\mathcal{D}(z))\|_2^2$ 
13:   $+ \frac{1}{2\sigma_t^2} \|z - z'_{t-1}\|_2^2$ 
14:   $z_{t-1} \leftarrow (z_{t-1})_{\downarrow k}$ 
15: end for
16: return  $\mathcal{D}(\hat{z}_0)$ 

```

latent space, representative examples include Latent Posterior Diffusion (LDPS) (Rout et al., 2023), Posterior Sampling with Latent Diffusion (PSLD) (Rout et al., 2023), ReSample (Song et al., 2024), and our formulation in Algorithm 2. The goal of Algorithm 2 is not to establish state-of-the-art performance, but rather to provide a simple and broadly effective algorithm across a variety of inverse problems. By contrast, contemporary inversion algorithms have certain drawbacks: PSLD relies on a correction term whose form is unclear for arbitrary forward operators, while ReSample enforces data consistency through an optimization that bridges latent and pixel spaces, making it unclear how to define a tunable parameter in pixel space without dedicated training. The experiments in Figure 4 employ Algorithm 2, whereas tunable versions of LDPS and PSLD would fall under the general template of Algorithm 1.

Our approach follows a generic template for tunable diffusion priors. Starting from Gaussian noise in the latent space, the algorithm iteratively denoises the latent variable using the learned noise prediction network. Each reverse step is then corrected by a data-consistency operation that is consistent with measurements $\mathbf{y} = \mathcal{A}(\mathcal{D}(z))$. To tune the latent dimensionality across different dimensionality truncation operator at every iteration. This template unifies a broad family of inversion methods:

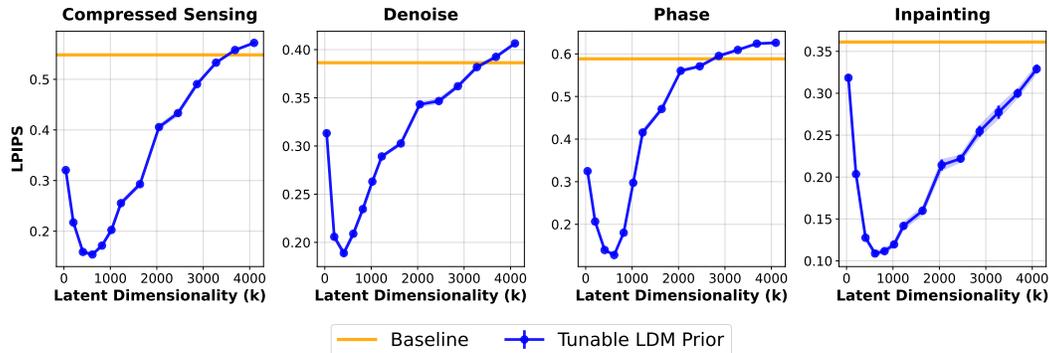


Figure 4: Performance of generative prior with a tunable complexity (Tunable LDM Prior) and its fixed complexity counterpart (BASELINE) for compressed sensing, inpainting, and denoising on CelebA Dataset. Tunable prior demonstrates a range of tunable parameters k that lead to a better estimate of the target signal measured in LPIPS than its baseline.

the prior supplies a generative update, the data-consistency step enforces measurements, and the tunable operator controls the representation capacity. The concrete instantiation in Algorithm 2 implements this template by combining a DDPM/DDIM reverse update with a quadratic data-consistency optimization.

4 Experiments

Our experiments underscore several key findings. First, we successfully demonstrate the construction of a tunable model (Figure 8). Second, tunable priors are broadly applicable they are not limited to any specific architecture, class of generative models, or inversion algorithm (Figures 2 and 4, Appendix B, and Tables 1 and 2). Finally, our approach is competitive with contemporary strong baselines (Dou & Song, 2024) and further improves upon already successful priors (Rout et al., 2023) demonstrated in Tables 1 and 2. The code will be released upon after the review period.

Naive Approach to Solving Tunable Prior We trained a family of injective flows following Ross et al. Ross & Cresswell (2021) with latent dimensionalities 16, 32, 64, 100, 112, 128, 200, 256, 384, 456, each on MNIST ($N = 1024$) for 1,000 epochs using their official GitHub Repository, without parameter sharing. All models had the same number of parameters and were used as priors for compressed sensing with Gaussian random measurements $A \in \mathbb{R}^{m \times n}$ and additive Gaussian noise η normalized to $\sqrt{\mathbb{E}|\eta|^2} = 0.1$.

Training Details To ensure reproducibility, we built on the Diffusers package (von Platen et al., 2022). For the VAE, we trained a standard continuous variant, while for the LDM we employed a U-Net in half-precision (float16) for memory efficiency, with approximately 200 million parameters operating on a $16 \times 16 \times 16$ latent space for CelebA (Coates et al., 2011) at an image size of $64 \times 64 \times 3$. All models were trained on a single RTX 2080 Ti GPU; training required about seven days in total—two days for the VAE and five days for the LDM. We used the AdamW optimizer (Loshchilov & Hutter, 2019) with a learning rate of 1×10^{-4} and a batch size of 128. Images were resized to 64×64 and normalized to $[-1, 1]$. Lastly, based on our objective, we found that setting $\alpha = 0.75$ yielded the lowest FID when averaged across latent dimensionalities. We therefore selected this model for solving inverse problems.

Inverse Problem Details For Figure 4, we evaluated four forward operators. All experiments were done on $64 \times 64 \times 3$ Compressed sensing used a random Gaussian operator $A \in \mathbb{R}^{m \times n}$ with i.i.d. $\mathcal{N}(0, 1/m)$ entries, where $m = 1228$ (approximately 10% of $n = 12288$). Denoising was modeled as additive Gaussian noise $\eta \sim \mathcal{N}(0, \sigma^2 I_n)$ with $\sigma = 0.25$. Inpainting employed a binary mask M with approximately 80% missing pixels. Phase retrieval used phaseless Gaussian measurements $|Ax|$, sharing the same A as compressed sensing, with a measurement ratio $m/n = 0.15$ (15%). All inverse problems were solved using Algorithm 2. In Table 1, we compare our approach to a contemporary state-of-the-art (SOTA) pixel-based diffusion prior (Chung et al., 2023) and show that our framework can also enhance other SOTA priors, such as PSLD (Rout et al., 2023), by leveraging

Method	CS		PHASE	
	PSNR \uparrow	LPIPS \downarrow	PSNR \uparrow	LPIPS \downarrow
DPS (Chung et al., 2023)	25.65 \pm 2.67	0.1593 \pm 0.0670	22.15 \pm 3.84	0.2665 \pm 0.0612
Tunable LDM Prior (ours)	25.49 \pm 1.70	0.1790 \pm 0.0451	25.24 \pm 1.71	0.1681 \pm 0.0464
LDPS (Rout et al., 2023)	23.30 \pm 1.24	0.2478 \pm 0.0634	24.48 \pm 1.35	0.1691 \pm 0.0381
Tunable LDPS (ours)	24.69 \pm 1.37	0.1635 \pm 0.0360	21.95 \pm 1.31	0.2564 \pm 0.0652
NF (Asim et al., 2020)	21.31 \pm 1.94	0.5080 \pm 0.0793	20.67 \pm 2.32	0.5544 \pm 0.0729
Tunable NF (ours)	27.16 \pm 1.56	0.2375 \pm 0.0582	26.11 \pm 2.07	0.2457 \pm 0.0758

Table 1: Quantitative Results on CelebA-HQ for Compressive Sensing and Phase Retrieval.

tunability to improve reconstruction quality. In addition, we introduce a tunable generative prior based on normalizing flows; details are provided in Appendix B. This model is trained following the objective of (Bekasov & Murray, 2020) and adopts the MAP estimator of Asim et al. (2020). Methods designated as tunable fall under our general formulation (Algorithm 1), and our goal is to enable a broad class of latent diffusion-based inversion algorithms to achieve improved results. For the measurement operators, we consider: compressed sensing with a measurement ratio $m/n = 0.075$; phase retrieval with 10% measurements; inpainting with 80% of pixels missing at random; and Gaussian deblurring with kernel size 5 and standard deviation 3. Lastly, for DPS we use a SOTA diffusion model trained on CelebA that achieves an FID of 1.27 on the training set (Ning et al., 2023).

Method	Inpaint (random)		Deblur (Gauss)	
	PSNR \uparrow	LPIPS \downarrow	PSNR \uparrow	LPIPS \downarrow
DPS (Chung et al., 2023)	23.18 \pm 2.11	0.1456 \pm 0.0420	27.90 \pm 1.53	0.0927 \pm 0.0187
Tunable LDM Prior (Ours)	25.30 \pm 1.76	0.1221 \pm 0.0280	29.52 \pm 1.11	0.0852 \pm 0.0239
PSLD (Rout et al., 2023)	25.85 \pm 2.18	0.1361 \pm 0.0360	27.33 \pm 1.23	0.1650 \pm 0.0371
Tunable PSLD (Ours)	26.75 \pm 1.71	0.0946 \pm 0.0280	28.50 \pm 1.05	0.1149 \pm 0.0305
NF (Asim et al., 2020)	20.07 \pm 2.42	0.3760 \pm 0.0912	21.43 \pm 2.11	0.4380 \pm 0.0341
Tunable NF (Ours)	23.24 \pm 2.43	0.1670 \pm 0.0569	24.85 \pm 2.21	0.2153 \pm 0.0433

Table 2: Quantitative Results on CelebA-HQ for Inpainting and Deblur.

5 Theory for Denoising with Linear Generative Model

The goal of this section is to provide preliminary theoretical conclusions that show tunability can lead to improved reconstruction errors relative to corresponding nontunable models. Further, this section aims to provide justification for how to select the tuning parameter for a generative model with tunable complexity. We present these insights for the problem of denoising under additive Gaussian noise. We show that the effect of tunability can be proven even in the case of linear generative models, and thus we restrict our attention to those.

We consider a family of linear generative models given as follows. Fix an invertible $\mathbf{G} \in \mathbb{R}^{n \times n}$, and let $\mathbf{G} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^T$ be a singular value decomposition with $\mathbf{\Sigma} = \text{diag}(s_1, \dots, s_n) \in \mathbb{R}^{n \times n}$. For any $k \leq n$, let $\mathbf{G}_k = \mathbf{U}\mathbf{\Sigma}_k\mathbf{V}^T \in \mathbb{R}^{n \times n}$, where $\mathbf{\Sigma}_k = \text{diag}(s_1, \dots, s_k, 0, \dots, 0) \in \mathbb{R}^{n \times n}$. Note that $\mathbf{G} = \mathbf{G}_n$. Each \mathbf{G}_k induces a probability distribution over \mathbb{R}^n by

$$\mathbf{x} = \mathbf{G}_k \mathbf{z}, \text{ where } \mathbf{z} \sim \mathcal{N}(0, \mathbf{I}_n).$$

We will refer to this distribution as $p_{\mathbf{G}_k}$ and observe that $p_{\mathbf{G}_k} = \mathcal{N}(0, \mathbf{G}_k \mathbf{G}_k^T)$. Note that k is a parameter that governs the complexity of the modeled signal class.

We consider the following denoising problem. Let $y = x_0 + \eta$, where $x_0 \sim p_{\mathbf{G}_n}$ and $\eta \sim \mathcal{N}(0, \sigma^2 \mathbf{I}_n)$. Our goal is to recover x_0 given y and \mathbf{G}_n . For a given k , we consider a maximum a posteriori (MAP) estimate of x_0 under the signal prior $p_{\mathbf{G}_k}$. This results in the following optimization problem over the latent space:

$$\hat{z}_\gamma(k) := \arg \min_{z^k \in \mathbb{R}^k} \frac{1}{2} \left\| \mathbf{y} - \mathbf{G}_k \begin{pmatrix} z^k \\ 0 \end{pmatrix} \right\|^2 + \frac{\gamma}{2} \|z^k\|^2 \quad (8)$$

where $\begin{pmatrix} z^k \\ 0 \end{pmatrix} \in \mathbb{R}^n$ is the vector z^k padded with zeroes, the estimated signal is $\hat{x}_\gamma(k) = \mathbf{G}_k \begin{pmatrix} \hat{z}_\gamma(k) \\ 0 \end{pmatrix}$, and γ is a parameter governing how strongly the prior is enforced. The case of $\gamma = 0$ corresponds to a Maximum Likelihood Estimate (MLE) formulation, and the case of $\gamma = \sigma^2$ corresponds to true MAP. It is common in the literature to consider γ as a hyperparameter, and thus we study the behavior Equation (8) for all $\gamma \geq 0$.

The following theorem provides an exact expression for the mean square error of the estimate above.

Theorem 5.1. Suppose we have a family $\{\mathbf{G}_k\}_{k=1\dots n}$ of generative models as given above, and let $p_{\mathbf{G}_k} = \mathcal{N}(0, \mathbf{G}_k \mathbf{G}_k^T)$, and let $\mathbf{G}_n \in \mathbb{R}^{n \times n}$ have singular values $s_1 \geq s_2 \geq \dots \geq s_n > 0$. Let $\mathbf{x}_0 \sim p_{\mathbf{G}_n}$ and $\boldsymbol{\eta} \sim \mathcal{N}(0, \sigma^2 \mathbf{I}_n)$. Then the estimator given by equation 8 yields

$$\mathbb{E}_{\mathbf{x}_0, \boldsymbol{\eta}} [\|\hat{\mathbf{x}}_\gamma(k) - \mathbf{x}_0\|^2] = \sum_{i=1}^k \frac{s_i^2 (s_i^2 \sigma^2 + \gamma^2)}{(s_i^2 + \gamma)^2} + \sum_{j=k+1}^n s_j^2.$$

The exact expression for reconstruction error in Theorem 5.1 makes it possible to immediately read of the optimal value of the tunable complexity parameter k , as established in the following corollary.

Corollary 5.2. Under the assumptions of Theorem 5.1, if $\gamma \leq \sigma^2/2$, then the parameter k that leads lowest reconstruction error of the target signal is given by:

$$\arg \min_{k \in [n]} \mathbb{E}_{\mathbf{x}_0, \boldsymbol{\eta}} \|\hat{\mathbf{x}}_\gamma(k) - \mathbf{x}_0\|^2 = \max \{k \mid s_k \geq \sqrt{\sigma^2 - 2\gamma}\}.$$

This corollary shows that for small enough hyperparameters γ , and in the presence of sufficient noise, then the optimal complexity parameter for minimizing reconstruction error can be less than the full signal dimensionality. In the case of MLE, an intermediate signal complexity is optimal if the variance of the measurement noise is larger than the smallest singular value of the linear generator \mathbf{G} . Additionally, we observe theoretically that if the noise level increases, then the optimal value of the tunable complexity decreases. The proofs to Theorem 5.1 and Corollary 5.2 are provided in Appendix.

6 Conclusion



Figure 5: Qualitative results based on Table 2. Blue is lowest LPIPS Score.

Prior work on solving inverse problems with generative priors has primarily focused on fixed models with fixed complexity, with most efforts directed at improving inversion algorithms. In this paper, we introduce tunability of complexity as a complementary and orthogonal axis of improvement. Rather than replacing algorithmic developments, tunability adds a new degree of freedom that practitioners can integrate into their preferred methods with minimal additional training overhead. We show that tunability consistently improves reconstruction across diverse model classes and inverse problems, with intermediate complexity often outperforming both lower and higher extremes. Finally, we present the first nested-dropout training algorithm for latent diffusion models, enabling adaptive complexity in SOTA generative priors. This constitutes a methodological advance for the diffusion community with impact beyond inverse problems.

This work opens several directions for future work. These include extending beyond latent models to pixel-space or other architectures, exploring softer and more flexible forms of complexity control, scaling to very large high-resolution datasets, and developing inversion algorithms specifically tailored to tunable models. Together, these opportunities point to tunability as a general principle for advancing generative priors in inverse problems.

7 REPRODUCIBILITY STATEMENT

We report all hyperparameter choices for our method in the Appendix and include detailed descriptions of the setup and configuration for every baseline. All models used in our experiments were trained by us, and to facilitate reproducibility, we will release both the code and trained model checkpoints to the public after the review period.

References

- Lynton Ardizzone, Jakob Kruse, Carsten Rother, and Ullrich Köthe. Analyzing inverse problems with invertible neural networks. In *International Conference on Learning Representations*, 2019. URL <https://openreview.net/forum?id=rJed6j0cKX>.
- Muhammad Asim, Max Daniels, Oscar Leong, Ali Ahmed, and Paul Hand. Invertible generative models for inverse problems: mitigating representation error and dataset bias. In *International Conference on Machine Learning*, pp. 399–409. PMLR, 2020.
- Artur Bekasov and Iain Murray. Ordering dimensions with nested dropout normalizing flows, 2020.
- Ashish Bora, Ajil Jalal, Eric Price, and Alexandros G Dimakis. Compressed sensing using generative models. In *International Conference on Machine Learning*, pp. 537–546. PMLR, 2017.
- Hyungjin Chung, Jeongsol Kim, Michael Thompson Mccann, Marc Louis Klasky, and Jong Chul Ye. Diffusion posterior sampling for general noisy inverse problems. In *The Eleventh International Conference on Learning Representations*, 2023. URL <https://openreview.net/forum?id=OnD9zGAGT0k>.
- Adam Coates, Andrew Ng, and Honglak Lee. An analysis of single-layer networks in unsupervised feature learning. In Geoffrey Gordon, David Dunson, and Miroslav Dudík (eds.), *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics*, volume 15 of *Proceedings of Machine Learning Research*, pp. 215–223, Fort Lauderdale, FL, USA, 11–13 Apr 2011. PMLR. URL <https://proceedings.mlr.press/v15/coates11a.html>.
- Jorio Cocola, Paul Hand, and Vlad Voroninski. Nonasymptotic guarantees for spiked matrix recovery with generative priors. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin (eds.), *Advances in Neural Information Processing Systems*, volume 33, pp. 15185–15197. Curran Associates, Inc., 2020. URL https://proceedings.neurips.cc/paper_files/paper/2020/file/ad62cfd33e3870262d6bf5331c1f13b0-Paper.pdf.
- Giannis Daras, Joseph Dean, Ajil Jalal, and Alex Dimakis. Intermediate layer optimization for inverse problems using deep generative models. In Marina Meila and Tong Zhang (eds.), *Proceedings of the 38th International Conference on Machine Learning, ICML 2021, 18-24 July 2021, Virtual Event*, volume 139 of *Proceedings of Machine Learning Research*, pp. 2421–2432. PMLR, 2021. URL <http://proceedings.mlr.press/v139/daras21a.html>.
- Giannis Daras, Yuval Dagan, Alex Dimakis, and Constantinos Daskalakis. Score-guided intermediate level optimization: Fast Langevin mixing for inverse problems. In Kamalika Chaudhuri, Stefanie Jegelka, Le Song, Csaba Szepesvari, Gang Niu, and Sivan Sabato (eds.), *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pp. 4722–4753. PMLR, 17–23 Jul 2022. URL <https://proceedings.mlr.press/v162/daras22a.html>.
- Prafulla Dhariwal and Alexander Nichol. Diffusion models beat gans on image synthesis. *Advances in neural information processing systems*, 34:8780–8794, 2021.
- Zehao Dou and Yang Song. Diffusion posterior sampling for linear inverse problem solving: A filtering perspective. In *The Twelfth International Conference on Learning Representations*, 2024. URL <https://openreview.net/forum?id=tplXNcHZs1>.
- Conor Durkan, Artur Bekasov, Iain Murray, and George Papamakarios. Neural spline flows. *Advances in neural information processing systems*, 32, 2019.

- 540 Bradley Efron. Tweedie’s formula and selection bias. *Journal of the American Statistical Association*,
541 106(496):1602–1614, 2011. doi: 10.1198/jasa.2011.tm11181. URL [https://doi.org/10.](https://doi.org/10.1198/jasa.2011.tm11181)
542 [1198/jasa.2011.tm11181](https://doi.org/10.1198/jasa.2011.tm11181). PMID: 22505788.
- 543
- 544 Patrick Esser, Robin Rombach, and Bjorn Ommer. Taming transformers for high-resolution im-
545 age synthesis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern*
546 *Recognition (CVPR)*, pp. 12873–12883, June 2021.
- 547 Paul Hand and Babhru Joshi. Global guarantees for blind demodulation with generative pri-
548 ors. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d’Alché-Buc, E. Fox, and R. Garn-
549 nett (eds.), *Advances in Neural Information Processing Systems*, volume 32. Curran Asso-
550 ciates, Inc., 2019. URL [https://proceedings.neurips.cc/paper_files/paper/](https://proceedings.neurips.cc/paper_files/paper/2019/file/598a90004bace6540f0e2230bdc47c09-Paper.pdf)
551 [2019/file/598a90004bace6540f0e2230bdc47c09-Paper.pdf](https://proceedings.neurips.cc/paper_files/paper/2019/file/598a90004bace6540f0e2230bdc47c09-Paper.pdf).
- 552
- 553 Paul Hand, Oscar Leong, and Vlad Voroninski. Phase retrieval under a generative prior.
554 In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garn-
555 nett (eds.), *Advances in Neural Information Processing Systems*, volume 31. Curran Asso-
556 ciates, Inc., 2018. URL [https://proceedings.neurips.cc/paper/2018/file/](https://proceedings.neurips.cc/paper/2018/file/1bc2029a8851ad344a8d503930dfd7f7-Paper.pdf)
557 [1bc2029a8851ad344a8d503930dfd7f7-Paper.pdf](https://proceedings.neurips.cc/paper/2018/file/1bc2029a8851ad344a8d503930dfd7f7-Paper.pdf).
- 558 Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In
559 H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin (eds.), *Advances in Neu-*
560 *ral Information Processing Systems*, volume 33, pp. 6840–6851. Curran Associates, Inc.,
561 2020. URL [https://proceedings.neurips.cc/paper_files/paper/2020/](https://proceedings.neurips.cc/paper_files/paper/2020/file/4c5bcfec8584af0d967f1ab10179ca4b-Paper.pdf)
562 [file/4c5bcfec8584af0d967f1ab10179ca4b-Paper.pdf](https://proceedings.neurips.cc/paper_files/paper/2020/file/4c5bcfec8584af0d967f1ab10179ca4b-Paper.pdf).
- 563
- 564 Ajil Jalal, Sushrut Karmalkar, Alex Dimakis, and Eric Price. Instance-optimal compressed sensing via
565 posterior sampling. In Marina Meila and Tong Zhang (eds.), *Proceedings of the 38th International*
566 *Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pp.
567 4709–4720. PMLR, 18–24 Jul 2021. URL [https://proceedings.mlr.press/v139/](https://proceedings.mlr.press/v139/jalal21a.html)
568 [jalal21a.html](https://proceedings.mlr.press/v139/jalal21a.html).
- 569 Bahjat Kawar, Gregory Vaksman, and Michael Elad. SNIPS: Solving noisy inverse problems
570 stochastically. In A. Beygelzimer, Y. Dauphin, P. Liang, and J. Wortman Vaughan (eds.), *Advances*
571 *in Neural Information Processing Systems*, 2021. URL [https://openreview.net/forum?](https://openreview.net/forum?id=9xPJ7cZ4ntc)
572 [id=9xPJ7cZ4ntc](https://openreview.net/forum?id=9xPJ7cZ4ntc).
- 573 Bahjat Kawar, Michael Elad, Stefano Ermon, and Jiaming Song. Denoising diffusion restoration
574 models. In *Advances in Neural Information Processing Systems*, 2022.
- 575
- 576 Diederik Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *International*
577 *Conference on Learning Representations (ICLR)*, San Diego, CA, USA, 2015.
- 578
- 579 Diederik P. Kingma and Max Welling. Auto-Encoding Variational Bayes. In *2nd International*
580 *Conference on Learning Representations, ICLR 2014, Banff, AB, Canada, April 14-16, 2014,*
581 *Conference Track Proceedings*, 2014.
- 582 Xin Li, Yulin Ren, Xin Jin, Cuiling Lan, Xingrui Wang, Wenjun Zeng, Xinchao Wang, and Zhibo
583 Chen. Diffusion models for image restoration and enhancement: A comprehensive survey, 2025.
584 URL <https://arxiv.org/abs/2308.09388>.
- 585
- 586 Tianci Liu, Tong Yang, Quan Zhang, and Qi Lei. Optimization for amortized inverse problems. In
587 *International Conference on Machine Learning*, pp. 22289–22319. PMLR, 2023.
- 588
- 589 Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *International Confer-*
590 *ence on Learning Representations (ICLR)*, 2019. URL [https://openreview.net/forum?](https://openreview.net/forum?id=Bkg6RiCqY7)
591 [id=Bkg6RiCqY7](https://openreview.net/forum?id=Bkg6RiCqY7).
- 592 Cheng Lu, Yuhao Zhou, Fan Bao, Jianfei Chen, Chongxuan Li, and Jun Zhu. Dpm-solver: A fast
593 ode solver for diffusion probabilistic model sampling in around 10 steps. In *Advances in Neural*
Information Processing Systems (NeurIPS), 2022.

- 594 Xiangming Meng and Yoshiyuki Kabashima. Quantized compressed sensing with score-based
595 generative models. In *The Eleventh International Conference on Learning Representations, 2023*.
596 URL https://openreview.net/forum?id=OOWLRfAI_V_.
597
- 598 Sachit Menon, Alexandru Damian, Shijia Hu, Nikhil Ravi, and Cynthia Rudin. PULSE: self-
599 supervised photo upsampling via latent space exploration of generative models. In *2020*
600 *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020, Seattle,*
601 *WA, USA, June 13-19, 2020*, pp. 2434–2442. Computer Vision Foundation / IEEE, 2020. doi:
602 10.1109/CVPR42600.2020.00251. URL [https://openaccess.thecvf.com/content_](https://openaccess.thecvf.com/content_CVPR_2020/html/Menon_PULSE_Self-Supervised_Photo_Upsampling_via_Latent_Space_Exploration_of_Generative_CVPR_2020_paper.html)
603 [CVPR_2020/html/Menon_PULSE_Self-Supervised_Photo_Upsampling_via_](https://openaccess.thecvf.com/content_CVPR_2020/html/Menon_PULSE_Self-Supervised_Photo_Upsampling_via_Latent_Space_Exploration_of_Generative_CVPR_2020_paper.html)
604 [Latent_Space_Exploration_of_Generative_CVPR_2020_paper.html](https://openaccess.thecvf.com/content_CVPR_2020/html/Menon_PULSE_Self-Supervised_Photo_Upsampling_via_Latent_Space_Exploration_of_Generative_CVPR_2020_paper.html).
- 605 Mang Ning, Enver Sangineto, Angelo Porrello, Simone Calderara, and Rita Cucchiara. Input
606 perturbation reduces exposure bias in diffusion models, 2023. URL [https://arxiv.org/](https://arxiv.org/abs/2301.11706)
607 [abs/2301.11706](https://arxiv.org/abs/2301.11706).
- 608 Gaurav Parmar, Richard Zhang, and Jun-Yan Zhu. On aliased resizing and surprising subtleties in
609 gan evaluation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern*
610 *Recognition (CVPR)*, pp. 11410–11420, June 2022.
- 611 Oren Rippel, Michael Gelbart, and Ryan Adams. Learning ordered representations with nested
612 dropout. In *International Conference on Machine Learning*, pp. 1746–1754. PMLR, 2014.
- 613 Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-
614 resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF Confer-*
615 *ence on Computer Vision and Pattern Recognition (CVPR)*, pp. 10684–10695, June 2022.
- 616 Brendan Leigh Ross and Jesse C Cresswell. Tractable density estimation on learned manifolds
617 with conformal embedding flows. In A. Beygelzimer, Y. Dauphin, P. Liang, and J. Wortman
618 Vaughan (eds.), *Advances in Neural Information Processing Systems*, 2021. URL [https://](https://openreview.net/forum?id=DqU-rIH4Eh)
619 openreview.net/forum?id=DqU-rIH4Eh.
- 620 Litu Rout, Negin Raouf, Giannis Daras, Constantine Caramanis, Alex Dimakis, and Sanjay Shakkottai.
621 Solving linear inverse problems provably via posterior sampling with latent diffusion models. In
622 *Thirty-seventh Conference on Neural Information Processing Systems*, 2023. URL [https:](https://openreview.net/forum?id=XKBFdYwfRo)
623 [/openreview.net/forum?id=XKBFdYwfRo](https://openreview.net/forum?id=XKBFdYwfRo).
- 624 Litu Rout, Yujia Chen, Abhishek Kumar, Constantine Caramanis, Sanjay Shakkottai, and Wen-Sheng
625 Chu. Beyond first-order tweedie: Solving inverse problems using latent diffusion. In *CVPR*, pp.
626 9472–9481, 2024. URL <https://doi.org/10.1109/CVPR52733.2024.00905>.
- 627 Jonathan Scarlett, Reinhard Heckel, Miguel R. D. Rodrigues, Paul Hand, and Yonina C. Eldar.
628 Theoretical perspectives on deep learning methods in inverse problems. *IEEE Journal on Selected*
629 *Areas in Information Theory*, 3(3):433–453, 2022. doi: 10.1109/JSAIT.2023.3241123.
- 630 Bowen Song, Soo Min Kwon, Zecheng Zhang, Xinyu Hu, Qing Qu, and Liyue Shen. Solving inverse
631 problems with latent diffusion models via hard data consistency. In *The Twelfth International*
632 *Conference on Learning Representations*, 2024. URL [https://openreview.net/forum?](https://openreview.net/forum?id=j8hdRqOUhN)
633 [id=j8hdRqOUhN](https://openreview.net/forum?id=j8hdRqOUhN).
- 634 Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. In *Interna-*
635 *tional Conference on Learning Representations*, 2021a. URL [https://openreview.net/](https://openreview.net/forum?id=StlgjarCHLP)
636 [forum?id=StlgjarCHLP](https://openreview.net/forum?id=StlgjarCHLP).
- 637 Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and Ben
638 Poole. Score-based generative modeling through stochastic differential equations. In *International*
639 *Conference on Learning Representations*, 2021b. URL [https://openreview.net/forum?](https://openreview.net/forum?id=PXTIG12RRHS)
640 [id=PXTIG12RRHS](https://openreview.net/forum?id=PXTIG12RRHS).
- 641 Yang Song, Liyue Shen, Lei Xing, and Stefano Ermon. Solving inverse problems in medical imaging
642 with score-based generative models. In *International Conference on Learning Representations*,
643 2022. URL <https://openreview.net/forum?id=vaRCHVj0uGI>.
- 644
- 645
- 646
- 647

Patrick von Platen, Suraj Xu, Pedro Cuenca Anaya, Thomas Wolf, and Hubert Schmid. Diffusers: State-of-the-art diffusion models. In *Proceedings of the 2022 Conference on Neural Information Processing Systems, Datasets and Benchmarks Track*, 2022. URL <https://github.com/huggingface/diffusers>.

Hengkang Wang, Xu Zhang, Taihui Li, Yuxiang Wan, Tiancong Chen, and Ju Sun. DMPlug: A plug-in method for solving inverse problems with diffusion models. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024. URL <https://openreview.net/forum?id=8lIFFsfQUj>.

Yinhuai Wang, Jiwen Yu, and Jian Zhang. Zero-shot image restoration using denoising diffusion null-space model. In *The Eleventh International Conference on Learning Representations*, 2023. URL <https://openreview.net/forum?id=mRieQgMtNTQ>.

Jay Whang, Qi Lei, and Alex Dimakis. Solving inverse problems with a flow-based noise model. In *International Conference on Machine Learning*, pp. 11146–11157. PMLR, 2021.

Richard Zhang, Phillip Isola, Alexei A. Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric, 2018. URL <https://arxiv.org/abs/1801.03924>.

A Appendix

We will release our code publicly after the review period. All fid scores are computed with (Parmar et al., 2022).

A.1 VAE Training/Inversion

Our VAE follows the `AutoencoderKL` architecture from von Platen et al. (2022), with a latent dimensionality of $16 \times 16 \times 16$, resulting in a model of approximately 44 million parameters. The model was trained as described in the main text. For the hyperparameter λ , we used a fixed value of 0.1. To incorporate perceptual similarity, we employed a VGG-based perceptual loss. Training was performed with a single λ setting and a geometric series success parameter of 10^{-3} . Further hyperparameter exploration remains possible. Furthermore, for the inversion algorithm refer to (Bora et al., 2017) Github Repository. Simple implementation of gradient descent on the latent vector.

A.2 LDM Training/Inversion

Following von Platen et al. (2022), we trained a `UNet2DModel` with approximately 200 million parameters using the AdamW optimizer with a learning rate of 1×10^{-4} and a batch size of 128. We employed the `DDPMScheduler` provided in the `diffusers` package. The FID scores reported below were generated using the `DPMSolverMultistepScheduler` (Lu et al., 2022) with 100 sampling steps. During training, we withheld a validation set and evaluated the FID score every 20,000 iterations, retaining the three checkpoints with the lowest FID scores. For inverse problems, we used the model corresponding to the best FID score. The FID score was computed using 50,000 training examples, while the test score was evaluated on 10,000 unseen images from the PyTorch implementation of CelebA.

For the inversion algorithms DPS (Chung et al., 2023) and PSLD/LPDS (Rout et al., 2023), we used the official codebases but adapted them to the Diffusers framework, which already provides existing pipelines. All hyperparameters were selected by grid search over the configuration files provided on GitHub, and we report results using the best-performing settings. These search procedures will be released alongside our official code. In general, for tunability we observed that DDIM sampling led to stronger improvements in intermediate representations compared to stochastic samplers, though additional experimental evidence is needed to fully understand this effect. For Algorithm 2, the best-performing setup used the Adam optimizer (Loshchilov & Hutter, 2019) with the DDIM reverse

702
703
704
705
706
707
708
709
710
711
712
713
714
715
716
717
718
719
720
721
722
723
724
725
726
727
728
729
730
731
732
733
734
735
736
737
738
739
740
741
742
743
744
745
746
747
748
749
750
751
752
753
754
755

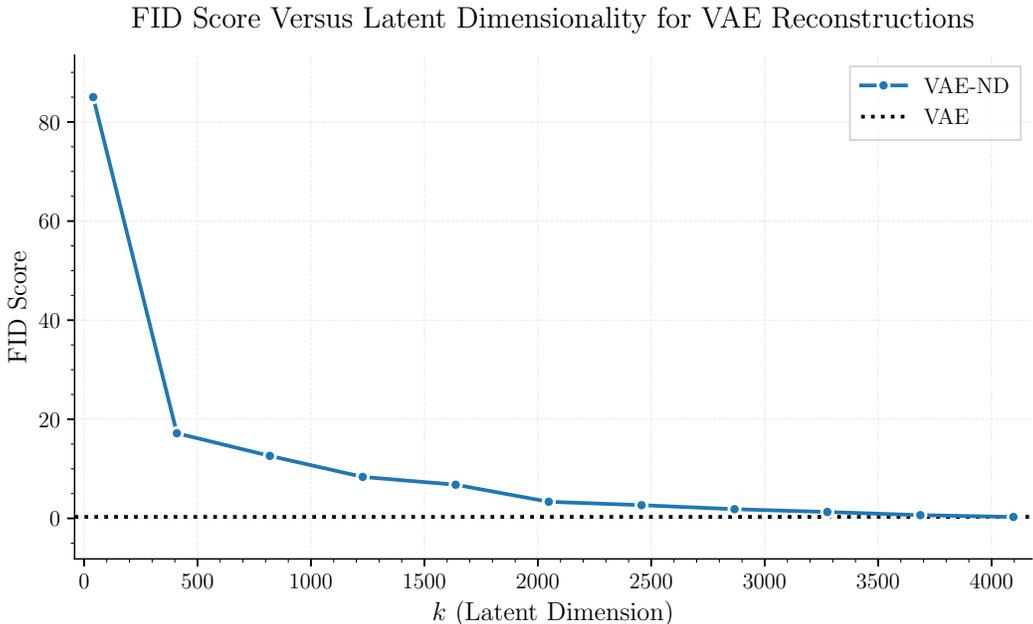


Figure 6: Reconstruction FID as function of latent dimensionality k .

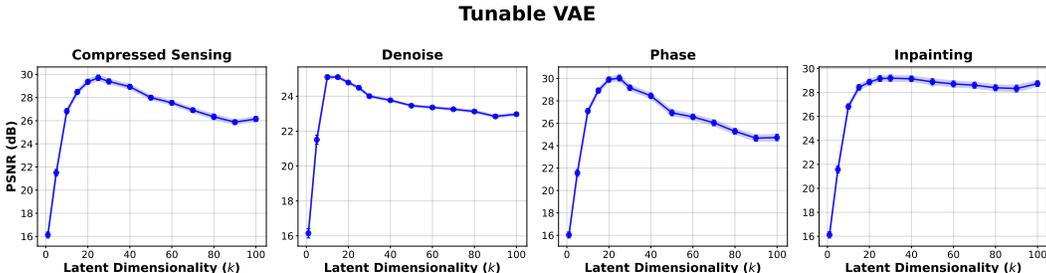
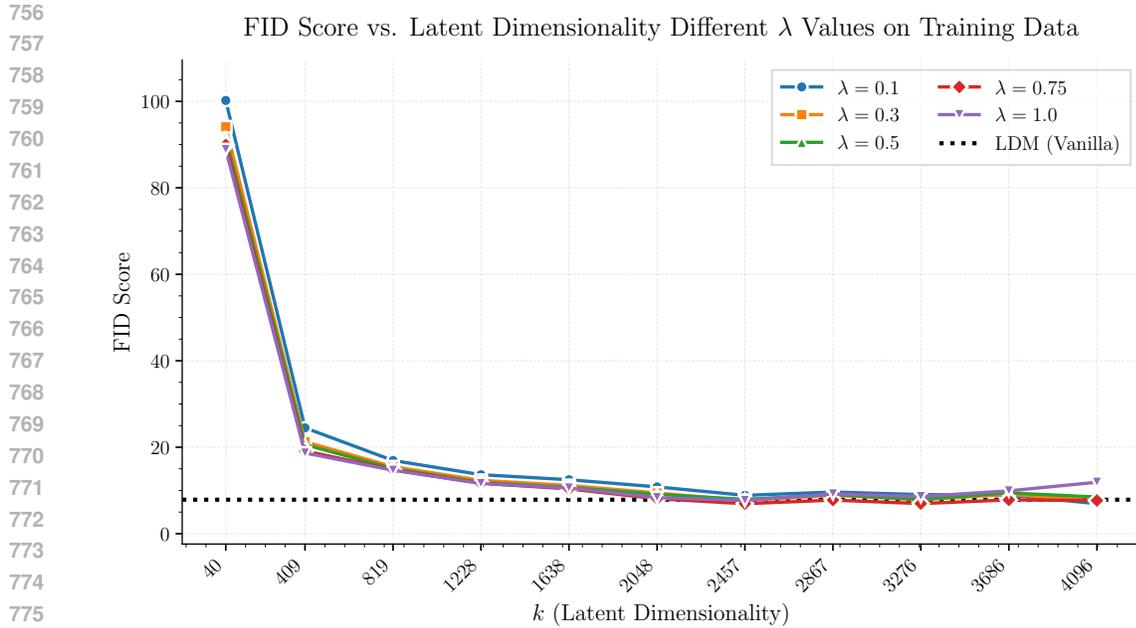
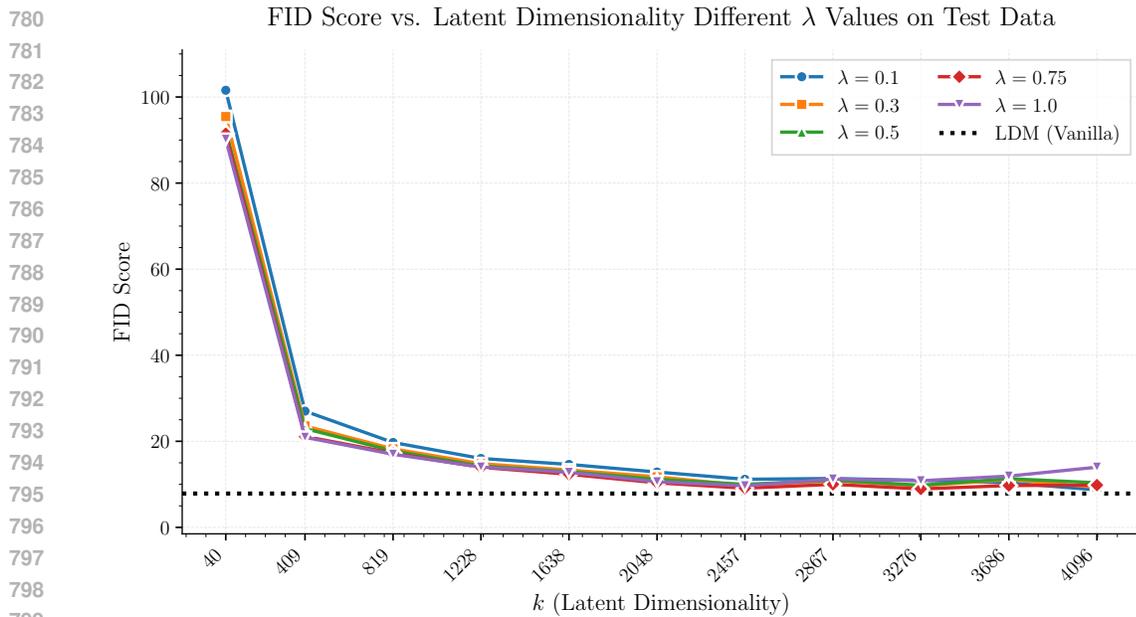


Figure 7: Performance of generative prior with a tunable complexity (Tunable VAE Prior) and its fixed complexity counterpart (VAE) for compressed sensing, inpainting, and denoising on CelebA Dataset. Tunable prior demonstrates a range of tunable parameters k that lead to a better estimate of the target signal measured in PSNR than its baseline.

equation, 5 optimization steps per update, 500 reverse steps, and variance set to 0.1. Importantly, we found that the algorithm does not work reliably without introducing a small degree of stochasticity. This approach is particularly effective in our setting due to the relatively large latent space compared to the image space. The overall objective was to implement a straightforward inversion algorithm that can be readily applied across a wide range of problems.

B Normalizing Flow Inverse Problems

We assume the practitioner has trained a tunable normalizing flow with the following objective provided by [Bekasov & Murray \(2020\)](#). Our formulation is for latent-variable generative priors. For simplicity assume the measurements have the following form: the forward operator $\mathcal{A} = \mathbf{A} \in \mathbb{R}^{m \times n}$ and the additive noise term is i.i.d $\eta \sim \mathcal{N}(0, \sigma^2 \mathbf{I}_m)$, as in the compressed sensing case. Thus the measurements are generated by $\mathbf{y} = \mathbf{A}\mathbf{x} + \eta$, where $\mathbf{x} \in \mathbb{R}^n$ is an unknown signal.

Figure 8: Samples Generated FID Score as a function of k (latent dimensionality)Figure 9: Samples Generated FID Score as a function of k (latent dimensionality)

800
801
802
803
804 Consider a fixed invertible neural network $G : \mathbb{R}^n \rightarrow \mathbb{R}^n$. Define for each $k \in [n] = \{1, \dots, n\}$ the
805 map

806
807
808
809

$$G_k : \mathbb{R}^k \rightarrow \mathbb{R}^n$$

$$z \mapsto G \left(\begin{pmatrix} z \\ 0 \end{pmatrix} \right).$$

810
811
812
813
814
815
816
817
818
819
820
821
822
823
824
825
826
827
828
829
830
831
832
833
834
835
836
837
838
839
840
841
842
843
844
845
846
847
848
849
850
851
852
853
854
855
856
857
858
859
860
861
862
863

Method: Gradient Based Sampling • Forward op: Phase retrieval ($|Ax|$) ($m/n=0.15$) • LPIPS (batch avg, best-over-steps): 0.132
Ground Truth Reconstruction



Figure 10: Qualitative Results of Figure 4

Method: Gradient Based Sampling • Forward op: Phase retrieval ($|Ax|$) ($m/n=0.15$) • LPIPS (batch avg, best-over-steps): 0.132
Ground Truth Reconstruction

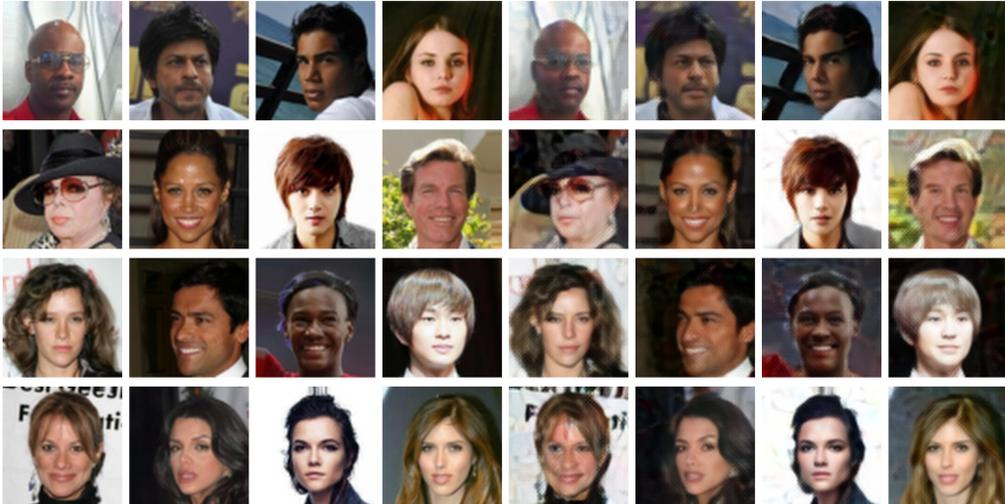


Figure 11: Qualitative Results of Figure 4

864
865
866
867
868
869
870
871
872
873
874
875
876
877
878
879
880
881
882
883
884
885
886
887
888
889
890
891
892
893
894
895
896
897
898
899
900
901
902
903
904
905
906
907
908
909
910
911
912
913
914
915
916
917



Figure 12: Generated samples from the latent diffusion model with different latent dimensionalities. Each row shows images produced when only the first k of the 4096 latent dimensions are kept during sampling ($k = 4096$: 100%, $k = 3600$: 88%, $k = 3000$: 73%, $k = 2500$: 61%, $k = 2000$: 49%, $k = 1500$: 37%).

We note that k is the parameter that governs the complexity of the prior, and $\begin{pmatrix} z \\ 0 \end{pmatrix} \in \mathbb{R}^n$ is obtained by latent representation $z \in \mathbb{R}^k$. Then the signal representation is given by $x = G((z^T, 0))$ where $x \in \mathbb{R}^n$ is as above.

We are interested in recovering a signal $\mathbf{x} \in \mathbb{R}^n$ given a set of noisy measurements $\mathbf{y} \in \mathbb{R}^m$. Our prior has a valid density over the entire signal space when $x \in \text{Range}(G_k)$, therefore a natural attempt to solve the given inverse problems is by a maximum *a posteriori* estimation:

$$\begin{aligned} \hat{x}_{\text{MAP}}(k) &:= \arg \min_{\mathbf{x} \in \text{Range}(G_k)} -\log p(\mathbf{x}|\mathbf{y}) \\ &= \arg \min_{\mathbf{x} \in \text{Range}(G_k)} -\log p(\mathbf{y}|\mathbf{x}) - \log p_{G_k}(x) \\ &= \arg \min_{\mathbf{x} \in \text{Range}(G_k)} \frac{1}{2} \|\mathbf{y} - \mathbf{A}\mathbf{x}\|^2 - \sigma^2 \log p_{G_k}(x) \end{aligned}$$

where p_{G_k} is the density function on x induced by G_k , $\|\cdot\|$ is the Euclidean norm, and σ is given by the model noise. Similar to previous work of [Asim et al. \(2020\)](#); [Whang et al. \(2021\)](#), we optimized over the latent space given by:

$$\hat{z}_{\text{MAP}}(k) := \arg \min_{z \in \mathbb{R}^k} \frac{1}{2} \left\| \mathbf{y} - \mathbf{A}G \left(\begin{pmatrix} z \\ 0 \end{pmatrix} \right) \right\|^2 - \sigma^2 \log p_{G_k} \left(G \left(\begin{pmatrix} z \\ 0 \end{pmatrix} \right) \right)$$

The proposed optimization problem above is solved via gradient descent with Adam optimizer [Kingma & Ba \(2015\)](#) and is initialized at $z = 0$. Depending on the specific inverse problem at hand the tunable parameter k can be selected to change the dimensionality of the model, so the first k elements of the vector z will be optimized over and the rest will be set to zero. Our formulation relies on a density function p_{G_k} , and works by [Asim et al. \(2020\)](#); [Whang et al. \(2021\)](#) have shown that a smoothing parameter on a density function improves performance when solving the optimization problem above. We empirically observe the same phenomena, and as a result we replace σ^2 with a hyperparameter γ . For more information about choosing hyperparameters and how to compute p_{G_k} please refer to the Appendix. Therefore instead we optimized over a modified MAP estimate given by

$$\hat{z}_{\text{MAP}}(k) := \arg \min_{z \in \mathbb{R}^k} \frac{1}{2} \left\| \mathbf{y} - \mathbf{A}G \left(\begin{pmatrix} z \\ 0 \end{pmatrix} \right) \right\|^2 - \gamma \log p_{G_k} \left(G \left(\begin{pmatrix} z \\ 0 \end{pmatrix} \right) \right).$$

For the VAE we follow the same formulation as [Bora et al. \(2017\)](#), which is similar to the one above, but instead of having a tractable density we approximate it with ℓ_2 penalty in the latent space.

CelebA NF 5-bit Results

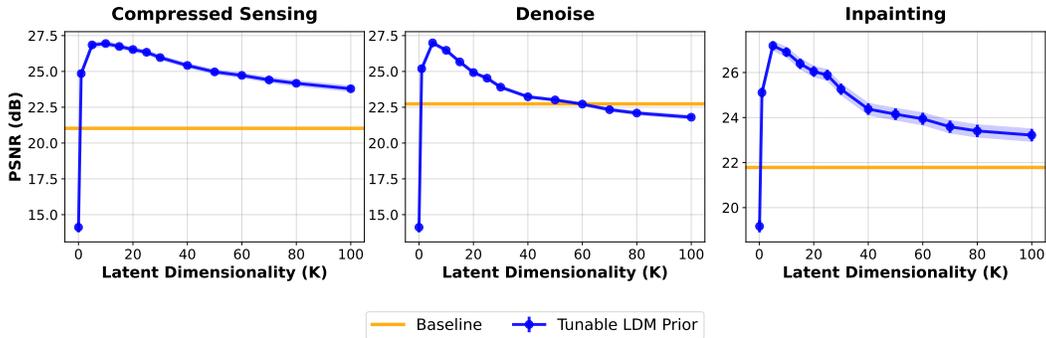
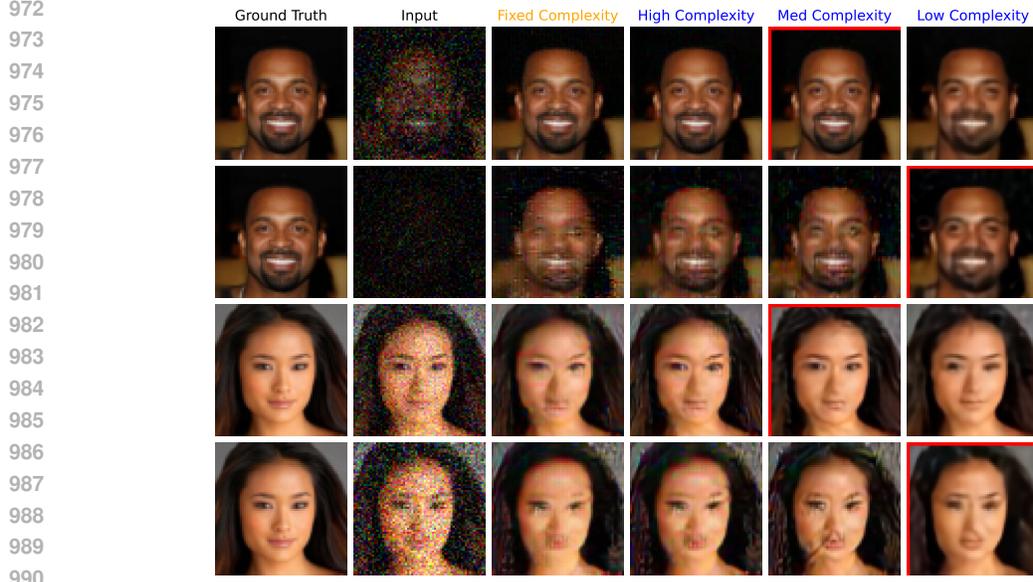


Figure 14: Performance of generative prior with a tunable complexity (Tunable NF Prior) and its fixed complexity counterpart (Normalizing Flow) for compressed sensing, inpainting, deblurring, and denoising on CelebA Dataset. Tunable prior demonstrates a range of tunable parameters k that lead to a better estimate of the target signal measured in PSNR than its baseline.



991
992
993
994
995
996

Figure 13: Results on inpainting and denoising across various measurement regimes for Normalizing Flow. The tuning parameter should be chosen based on the complexities of the given inverse problem. The red bounding box indicates which model achieved the lowest reconstruction error with respect to SSIM.

997 C Proofs

998
999
1000
1001 *Proof of Theorem 5.1.*

1002 Without loss of generality, we take G_k to be a diagonal matrix. This follows because of invariance of
1003 Euclidean norms with respect to orthogonal transformations and because of the rotational invariance
1004 of the random variable z_0 . Let \mathcal{P}_k and \mathcal{P}_k^\perp be the orthogonal projectors onto $\text{span}(\{e_1, \dots, e_k\})$ and
1005 its orthogonal complement, respectively. For a diagonal matrix $\Sigma = \text{diag}(s_1, s_2, \dots, s_n) \in \mathbb{R}^{n \times n}$,
1006 let $\mathbf{S}_k \in \mathbb{R}^{k \times k}$ and $\mathbf{S}_k^\perp \in \mathbb{R}^{n-k \times n-k}$ be such that

$$1007 \Sigma = \begin{pmatrix} \mathbf{S}_k & \mathbf{0} \\ \mathbf{0} & \mathbf{S}_k^\perp \end{pmatrix}.$$

1008
1009
1010
1011 We want to find the expectation

$$1012 \mathbb{E}_{x_0, \eta} \|\hat{x}_\gamma(k) - x_0\|^2 = \mathbb{E}_{x_0, \eta} \|\mathcal{P}_k(\hat{x}_\gamma(k) - x_0)\|^2 + \mathbb{E}_{x_0} \|\mathcal{P}_k^\perp(x_0)\|^2. \quad (9)$$

1013
1014
1015
1016 For the equation above we will compute the first term on the right hand side first, then do the same
1017 for the second term on the right hand side. The first term on the right hand side breaks into sum
1018 of two terms which can readily computed, while the second term on the right hand side follows
1019 from a lemma. Notice $\mathcal{P}_k(\hat{x}_\gamma(k)) = \mathbf{S}_k \hat{z}_\gamma(k) \in \mathbb{R}^k$ and the solution of the optimization problem
1020 equation 8 is

$$1021 \hat{z}_\gamma(k) := (\mathbf{S}_k \mathbf{S}_k^T + \gamma \mathbf{I}_k)^{-1} \mathbf{S}_k^T \mathcal{P}_k(y)$$

$$1022 = (\mathbf{S}_k \mathbf{S}_k^T + \gamma \mathbf{I}_k)^{-1} \mathbf{S}_k^T (\mathbf{S}_k \mathcal{P}_k(z_0) + \mathcal{P}_k(\eta)).$$

1023
1024
1025

1026 Thus,

$$\begin{aligned}
1027 & \mathbb{E}_{x_0, \eta} \|\mathcal{P}_k(\hat{x}_\gamma(k) - x_0)\|^2 = \mathbb{E}_{z_0, \eta} \|\mathbf{S}_k \hat{z}_\gamma(k) - \mathbf{S}_k \mathcal{P}_k(z_0)\|^2 \\
1028 & = \mathbb{E}_{z_0, \eta} \|\mathbf{S}_k (\mathbf{S}_k \mathbf{S}_k^T + \gamma \mathbf{I}_k)^{-1} \mathbf{S}_k^T \mathcal{P}_k(y) - \mathbf{S}_k \mathcal{P}_k(z_0)\|^2 \\
1029 & = \mathbb{E}_{z_0, \eta} \|\mathbf{S}_k (\mathbf{S}_k \mathbf{S}_k^T + \gamma \mathbf{I}_k)^{-1} \mathbf{S}_k^T (\mathbf{S}_k \mathcal{P}_k(z_0) + \mathcal{P}_k(\eta)) \\
1030 & \quad - \mathbf{S}_k \mathcal{P}_k(z_0)\|^2 \\
1031 & = \mathbb{E}_{z_0, \eta} \|(\mathbf{S}_k (\mathbf{S}_k \mathbf{S}_k^T + \gamma \mathbf{I}_k)^{-1} \mathbf{S}_k - \mathbf{S}_k) \mathcal{P}_k(z_0) \\
1032 & \quad + \mathbf{S}_k (\mathbf{S}_k \mathbf{S}_k^T + \gamma \mathbf{I}_k)^{-1} \mathbf{S}_k^T \mathcal{P}_k(\eta)\|^2 \tag{10}
\end{aligned}$$

1036 The last term above Equation (10) can be decomposed

$$1038 \mathbb{E}_{z_0} \|(\mathbf{S}_k (\mathbf{S}_k \mathbf{S}_k^T + \gamma \mathbf{I}_k)^{-1} \mathbf{S}_k - \mathbf{S}_k) \mathcal{P}_k(z_0)\|^2 \tag{11}$$

$$1039 + \mathbb{E}_{z_0, \eta} \langle (\mathbf{S}_k (\mathbf{S}_k \mathbf{S}_k^T + \gamma \mathbf{I}_k)^{-1} \mathbf{S}_k - \mathbf{S}_k) \mathcal{P}_k(z_0), \mathbf{S}_k (\mathbf{S}_k \mathbf{S}_k^T + \gamma \mathbf{I}_k)^{-1} \mathbf{S}_k^T \mathcal{P}_k(\eta) \rangle \tag{12}$$

$$1041 + \mathbb{E}_\eta \|\mathbf{S}_k (\mathbf{S}_k \mathbf{S}_k^T + \gamma \mathbf{I}_k)^{-1} \mathbf{S}_k^T \mathcal{P}_k(\eta)\|^2 \tag{13}$$

1043 For the term above, the expectation of inner product term (Equation (12)) is zero because both random
1044 variables have an expectation of zero. Furthermore, apply Lemma C.1 to Equation (11), yielding

$$\begin{aligned}
1047 & \mathbb{E}_{z_0} \|(\mathbf{S}_k (\mathbf{S}_k \mathbf{S}_k^T + \gamma \mathbf{I}_k)^{-1} \mathbf{S}_k^T \mathbf{S}_k - \mathbf{S}_k) \mathcal{P}_k(z_0)\|^2 \\
1048 & = \|(\mathbf{S}_k \mathbf{S}_k \mathbf{S}_k^T + \gamma \mathbf{I}_k)^{-1} \mathbf{S}_k^T \mathbf{S}_k - \mathbf{S}_k\|_F^2 \\
1049 & = \sum_{i=1}^k \left| \frac{s_i^3}{s_i^2 + \gamma} - s_i \right|^2 \\
1050 & = \sum_{i=1}^k \frac{s_i^2 \gamma^2}{(s_i^2 + \gamma)^2}. \tag{14}
\end{aligned}$$

1056 Then apply Lemma C.2 to Equation (13)

$$\begin{aligned}
1057 & \mathbb{E}_\eta \|\mathbf{S}_k (\mathbf{S}_k \mathbf{S}_k^T + \gamma \mathbf{I}_k)^{-1} \mathbf{S}_k^T \mathcal{P}_k(\eta)\|^2 \\
1058 & = \text{Tr}(\mathbf{S}_k (\mathbf{S}_k \mathbf{S}_k^T + \gamma \mathbf{I}_k)^{-1} \mathbf{S}_k^T \sigma^2 \mathbf{I}_k \mathbf{S}_k (\mathbf{S}_k \mathbf{S}_k^T + \gamma \mathbf{I}_k)^{-1} \mathbf{S}_k^T) \\
1059 & = \sum_{i=1}^k \frac{s_i^4 \sigma^2}{(s_i^2 + \gamma)^2}. \tag{15}
\end{aligned}$$

1063 This concludes the calculation of the first term on the right hand side of Equation (9). Now we apply
1064 Lemma C.1 to the second term on the right hand side

$$1066 \mathbb{E}_{x_0} \|\mathcal{P}_k^\perp(x_0)\|^2 = \mathbb{E}_{z_0} \|\mathbf{S}_k^\perp \mathcal{P}_k^\perp(z_0)\|^2 = \|\mathbf{S}_k^\perp\|_F^2 = \sum_{j=k+1}^n s_j^2. \tag{16}$$

1069 Lastly, combine Eq.16,14,15 and simplify

$$1071 \mathbb{E}_{x_0, \eta} \|\hat{x}_\gamma(k) - x_0\|^2 = \sum_{i=1}^k \frac{s_i^2 (s_i^2 \sigma^2 + \gamma^2)}{(s_i^2 + \gamma)^2} + \sum_{j=k+1}^n s_j^2.$$

1074 □

1076 *Proof of Corollary 5.2.*

1077 For notational convenience, let $E(k) = \mathbb{E}_{x_0, \eta} \|\hat{x}_\gamma(k) - x_0\|^2$. Since $[n]$ is a finite non-empty set,
1078 $\arg \min_k E(k)$ exists, and so we consider the set C of candidate minimizers,

$$1079 C = \{k \mid E(k-1) \geq E(k)\}.$$

1080 Notice $E(k-1) - E(k) \geq 0$ for each $k \in C$, and moreover that this difference is

$$\begin{aligned}
1082 & \sum_{i=1}^{k-1} \frac{s_i^2 (s_i^2 \sigma^2 + \gamma^2)}{(s_i^2 + \gamma)^2} + \sum_{j=k}^n s_j^2 - \sum_{i=1}^k \frac{s_i^2 (s_i^2 \sigma^2 + \gamma^2)}{(s_i^2 + \gamma)^2} \\
1084 & - \sum_{j=k+1}^n s_j^2 \geq 0 \\
1086 & \iff \\
1088 & s_k - \frac{s_k^2 (s_k^2 \sigma^2 + \gamma^2)}{(s_k^2 + \gamma)^2} \geq 0 \\
1090 & \iff \\
1092 & s_k \geq \sqrt{\sigma^2 - 2\gamma},
\end{aligned}$$

1095 which reveals

$$1096 \quad C = \{k \mid s_k \geq \sqrt{\sigma^2 - 2\gamma}\}.$$

1097 From our assumptions, $\sigma^2 - 2\gamma \geq 0$. Note now that for any $i, j \in C$ with $i < j$, we have that

$$1098 \quad s_j \geq s_k \geq s_i \geq \sqrt{\sigma^2 - 2\gamma} \quad \text{for all } i < k \leq j,$$

1100 hence $k \in C$.

1101 To prove the corollary is to prove that $\arg \min_k E(k) = \max C$, so it suffices to show E is decreasing

1103 on C . Since $s_i^2 \geq \sigma^2 - 2\gamma$ for all $i \in C$, it is clear that

$$1104 \quad s_i^4 + s_i^2(2\gamma - \sigma^2) \geq s_i^4 - s_i^2 \cdot s_i^2 = 0$$

1106 is non-negative. From this, we deduce that $s_i^2 \sigma^2 + \gamma^2 \leq (s_i^2 + \gamma)^2$, hence

$$1108 \quad \Delta(i) := 1 - \frac{s_i^2 \sigma^2 + \gamma^2}{(s_i^2 + \gamma)^2} \geq 0$$

1110 is also non-negative.

1111 Let $P, Q \in C$ be such that $P \leq Q$. We'll show $E(P) \geq E(Q)$ by showing the difference

1113 $E(P) - E(Q)$ is non-negative. It is easily computed by

$$1114 \quad E(P) - E(Q) = \sum_{i=P+1}^Q s_i^2 \Delta(i).$$

1117 Previous work shows that $\Delta(i)$ is non-negative for all i in the range $P < i \leq Q$ since all such i are

1119 elements of C , thus concluding the proof. □

1121 **Lemma C.1.** Let $x \sim \mathcal{N}(0, \mathbf{I}_n) \in \mathbb{R}^n$ then for any matrix $M \in \mathbb{R}^{m \times n}$, we have

$$1122 \quad \mathbb{E}_x \|Mx\|^2 = \|M\|_F^2.$$

1125 *Proof of Lemma C.1.*

1126 Compute that

$$\begin{aligned}
1127 & \mathbb{E}_x \|Mx\|^2 = \mathbb{E}_x \langle Mx, Mx \rangle \\
1128 & = \mathbb{E}_x \langle M^T M, xx^T \rangle \\
1129 & = \langle M^T M, \mathbf{I}_n \rangle \\
1130 & = \|M\|_F^2.
\end{aligned}$$

1133 □

1134 **Lemma C.2.** Let $x \sim \mathcal{N}(0, \Sigma) \in \mathbb{R}^n$ then for any matrix $M \in \mathbb{R}^{m \times n}$, we have

$$1135 \mathbb{E}_x \|Mx\|^2 = \text{Tr}(M\Sigma M^T).$$

1137

1138 *Proof of Lemma C.2.*

1139 Let $Y = Mx$, then $Y \sim \mathcal{N}(0, M\Sigma M^T)$. Thus,

1140

$$1141 \mathbb{E}_x \|Mx\|^2 = \mathbb{E}_Y \|Y\|^2 = \mathbb{E}[Y^T Y] = \sum_i^n Y_i^2 = \sum_{i=1}^n \text{Var}(Y_i)$$

1142

1143

$$1144 = \text{Tr}(M\Sigma M^T).$$

(17)

1145

1146

□

1147

1148 You may include other additional sections here.

1149

1150

1151

1152

1153

1154

1155

1156

1157

1158

1159

1160

1161

1162

1163

1164

1165

1166

1167

1168

1169

1170

1171

1172

1173

1174

1175

1176

1177

1178

1179

1180

1181

1182

1183

1184

1185

1186

1187