# VIPAINT: IMAGE INPAINTING WITH PRE-TRAINED DIFFUSION MODELS VIA VARIATIONAL INFERENCE

**Sakshi Agarwal**
Dept. of Computer Science
University of California, Irvine
sakshia1@uci.edu

**Gabriel Hope**
Dept. of Computer Science
Harvey Mudd College
ghope@hmc.edu

**Erik B. Sudderth**
Dept. of Computer Science
University of California, Irvine
sudderth@uci.edu

## ABSTRACT

Diffusion probabilistic models learn to remove noise added during training, generating novel data (e.g., images) from Gaussian noise through sequential denoising. However, conditioning the generative process on corrupted or masked images is challenging. While various methods have been proposed for inpainting masked images with diffusion priors, they often fail to produce samples from the true conditional distribution, especially for large masked regions. Additionally, many can't be applied to latent diffusion models which have been demonstrated to generate high-quality images, while offering efficiency in model training. We propose a hierarchical variational inference algorithm that optimizes a non-Gaussian Markov approximation of the true diffusion posterior. Our VIPaint method outperforms existing approaches in both plausibility and diversity of imputations, and is also effective for other inverse problems like deblurring and superresolution.

## 1 INTRODUCTION

Diffusion models (Ho et al., 2020b; Song et al., 2021b; Nichol & Dhariwal, 2021; Song & Ermon, 2019) learn to generate synthetic data by sequentially reducing Gaussian noise across hundreds or thousands of steps, producing deep generative models that have advanced the state-of-the-art in natural image generation (Dhariwal & Nichol, 2021; Kingma et al., 2021a; Karras et al., 2022). Diffusion models for high-dimensional data like images are computationally intensive. Efficiency may be improved by leveraging an autoencoder (Kingma & Welling, 2019; Rombach et al., 2022; Vahdat et al., 2021) to map data to a lower-dimensional encoding, and then training a diffusion model for the lower-dimensional codes. This dimensionality reduction enables tractable but expressive models for images with millions of pixels. The effectiveness of *latent diffusion models* (LDMs) has made them a new standard for natural image generation, and they are thus our focus here.

Motivated by the foundational information captured by diffusion models of images, numerous algorithms have incorporated a pre-trained diffusion model as a prior for image editing (Meng et al., 2021), inpainting (Song et al., 2021b; Wang et al., 2023b; Kawar et al., 2022; Chung et al., 2022a; Lugmayr et al., 2022; Cardoso et al., 2024; Feng et al., 2023; Trippe et al., 2023; Dou & Song, 2024), or other inverse problems (Kadkhodaie & Simoncelli, 2021; Song et al., 2023; Graikos et al., 2022; Mardani et al., 2023; Chung et al., 2023). Many of these prior methods are specialized to inpainting with pixel-based diffusion models, where every data dimension is either perfectly observed or completely missing, and are not easily adapted to state-of-the-art LDMs.

For image inpainting, popular methods like DPS Chung et al. (2023) and RedDiff (Mardani et al., 2023) simplified evaluations by only masking a small fraction of test images. More recent work like RePaint Lugmayr et al. (2022) and CoPaint Zhang et al. (2023) notes that these methods struggle with large masks. Liu et al. (2024b) successfully adapted probabilistic circuits (Choi et al., 2020) for inpainting, but their supervised approach must be trained to match a known image mask distribution. Wang et al. (2024) assume additional side-information, such as segmentations or depths or poses, is available to inpaint large mask regions.

Most widely used inpainting algorithms employ an iterative refinement procedure, like that used to generate unconditional samples, and guide their predictions towards the partially observed image via various approximations and heuristics. In Fig. 1, we see that by sequentially annealing from

independent Gaussian noise to noise-free images, these approaches produce myopic samples that do not adequately incorporate information from observed pixels and fail to correct errors introduced in earlier stages of the "reverse-time" diffusion. More recent work extends these approaches to image editing (Avrahami et al., 2022) or inpainting (Rout et al., 2023; Corneanu et al., 2024; Chung et al., 2023; Song et al., 2024) with LDM, but they continue to suffer similar inaccuracies (see Sec 5).

We propose *VIPaint*, a novel application of *variational inference* (VI) (Wainwright & Jordan, 2008; Blei et al., 2017) that employs both LDMs and pixel-based DMs as priors to handle large masks for image inpainting. VI has achieved excellent image restoration results with a wide range of priors, including mixtures (Fergus et al., 2006; Ji et al., 2017) and hierarchical VAEs (Agarwal et al., 2023), but there is little work exploring its integration with state-of-the-art LDMs. While *RedDiff* (Mardani et al., 2023) applies VI to approximate the posterior of pixel-based DMs, its local approximation of the noise-free image posterior is difficult to optimize, requiring annealing heuristics that we demonstrate are sensitive to local optima. Instead, VIPaint strategically defines a hierarchical, Markovian and non-Gaussian approximation to the true (L)DM posterior that accounts for a subset of latent noise levels, enabling the inference of both high-level semantics and low-level details from observed pixels *simultaneously* (see Fig. 1). Further, we efficiently infer variational parameters for each inpainting query, avoiding the need to collect a training set of corrupted images (Liu et al., 2024a; Corneanu et al., 2024), expensively fine-tune generative models (Avrahami et al., 2022) or variational posteriors (Feng et al., 2023) for each query, or retrain large-scale conditional diffusion models (Rombach et al., 2022; Saharia et al., 2022a; Nichol et al., 2022; Chung et al., 2022b).

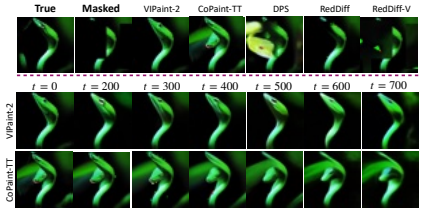


Figure 1: **Top:** Sampling methods like CoPaint-TT (Zhang et al., 2023) and DPS (Chung et al., 2023) produce incoherent images for large masks.Our VIPaint method leads to more robust inference. **Bottom:** We visualize intermediate samples in the diffusion latent space, comparing VIPaint with the best baseline, CoPaint-TT. CoPaint-TT's intermediate samples fail to correctly infer the underlying image, while VIPaint's samples better align with the observations and are more coherent.

We begin by reviewing properties of (latent) diffusion models in Sec. 2, and prior work on inferring images via pretrained diffusion models in Sec. 3. Sec. 4 then develops the VIPaint algorithm, which first fits a hierarchical posterior that best aligns with the observations, and then samples from this approximate posterior to produce diverse reconstruction hypotheses. Results in Sec. 5 on inpainting, and the Appendix on other inverse problems, then show substantial qualitative and quantitative improvements in capturing multimodal uncertainty for both pixel-based and latent DMs.

## 2 Background : Diffusion Models

The diffusion process begins with clean data $x$, and defines a sequence of increasingly noisy versions of $x$, which we call the *latent variables* $z_t$, where $t$ runs from $t = 0$ (low noise) to $t = T$ (substantial noise). The distribution of latent variable $z_t$ given $x$, for any integer time $t \in [0, T]$, is

$$q(z_t \mid x) = \mathcal{N}(z_t \mid \alpha_t x, \sigma_t^2 I), \tag{1}$$

where $\alpha_t$ and $\sigma_t$ are strictly increasing scalar functions of $t$. This noise implicitly defines a Markov chain for which the conditionals $q(z_t \mid z_{t-1})$, $q(z_{t-1} \mid z_t, x)$ are tractable Gaussians (see Appendix B.1). The signal-to-noise ratio (Kingma et al., 2021b) induced by this diffusion process at time $t$ equals $SNR(t) = \alpha_t^2/\sigma_t^2$. The SNR monotonically decrease with time, so that $SNR(t) < SNR(s)$ for $t > s$. This DM specification includes variance-preserving diffusions (Ho et al., 2020a; Sohl-Dickstein et al., 2015) as a special case, where $\alpha_t = \sqrt{1 - \sigma_t^2}$. Another special case, variance-exploding diffusions (Song & Ermon, 2019; Song et al., 2021b), takes $\alpha_t = 1$.

**Image Generation.** The generative model reverses the diffusion process outlined in Eq. (1), resulting in a hierarchical generative model that samples a sequence of latent variables $z_t$ before sampling $x$. Generation progresses backward in time from $t = T$ to $t = 0$ via a finite temporal discretization into $T \approx 1000$ steps, either uniformly spaced as in discrete diffusion models (Ho et al., 2020a), or via a possibly non-uniform discretization (Karras et al., 2022) of an underlying continuous-time stochastic differential equation (Song et al., 2021b). Denoting $t - 1$ as the timestep preceding $t$, for

$0 < t < T$, the hierarchical generative model for data $x$ is expressed as follows:

$$p_\theta(x) = \int_z p(z_T)p(x \mid z_0) \prod_{t=1}^{T} p_\theta(z_{t-1} \mid z_t)\, dz. \tag{2}$$

The marginal distribution of $z_T$ is typically a spherical Gaussian $p(z_T) = \mathcal{N}(z_T \mid 0, \sigma_T^2 I)$. Pixel-based diffusion models take $p(x \mid z_0)$ to be a simple factorized likelihood for each pixel in $x$, while LDMs define $p(x \mid z_0)$ using a decoder neural network. The conditional latent distribution $p_\theta(z_{t-1} \mid z_t)$ maintains the same form as the forward noise process $q(z_{t-1} \mid z_t, x)$, but with the data $x$ approximated by the output of a parameterized denoising model:

$$p_\theta(z_{t-1} \mid z_t) = q(z_{t-1} \mid z_t, z_0 = \hat{z}_\theta(z_t, t)), \text{where} \quad \hat{z}_\theta(z_t, t) = \frac{z_t - \sigma_t \hat{\epsilon}_\theta(z_t, t)}{\alpha_t}. \tag{3}$$

The denoising model $\hat{\epsilon}_\theta(z_t, t)$ typically uses variants of the UNet architecture (Ronneberger et al., 2015) and is trained to optimize a re-weighted variational lower bound of the marginal likelihood of data $x$, which after simplification (Ho et al., 2020b; Song et al., 2021b) can be written as

$$\mathcal{L}_{(0,T)}(z_0) = \frac{T}{2} \mathbb{E}_{\epsilon \sim \mathcal{N}(0,1), t \sim \mathcal{U}(1,T)} \left[ ||\epsilon - \hat{\epsilon}_\theta(z_t, t)||_2^2 \right]. \tag{4}$$

The expectation is taken over times $t$, $z_t \sim q(z_t|x)$, and $x \sim p_{\text{data}}(x)$. Latent diffusion models Rombach et al. (2022); Vahdat et al. (2021) use an encoder $q_\phi(z_0|x)$ to map $x$ to a lower-dimensional space and train a diffusion model in this reduced encoding space for efficiency.

## 3 Background : Inference with Diffusion Models

In many real-life scenarios, we encounter partial observations $y = x \odot m$, where $m$ is a binary mask indicating missing pixels. In cases where large portions of the image are masked, exactly recovering $x$ from $y$ is challenging, because many $x$ could produce the same observation $y$. To express the resulting posterior $p_\theta(x \mid y)$ given a DM prior, we can adapt the generative process in Eq. 2 as:

$$p_\theta(x \mid y) = \int_z p_\theta(z_T \mid y) p_\theta(x \mid z_0, y) \prod_{t=1}^{T} p_\theta(z_{t-1} \mid z_t, y)\, dz. \tag{5}$$

Exactly evaluating this predictive distribution is infeasible due to the non-linear noise prediction (and decoder) network, and the intractable posteriors of latent codes $p(z_{t-1} \mid z_t, y)$ for all $t$. Various methods have been proposed to conditionally sample latent codes $z_{t-1}$, detailed in Appendix B.5. Below, we elaborate RedDiff, a variational inference approach that is a special case of VIPaint.

### 3.1 RedDiff: Variational Inference of Missing Data

*RedDiff* (Mardani et al., 2023) uses pixel-based diffusion models as priors and defines a variational distribution to approximate $p_\theta(x \mid y)$. It defines a simple Gaussian variational distribution over the data space $x$ as $q_\lambda(x) = \mathcal{N}(\mu, \sigma^2)$, where $\lambda = \{\mu, \sigma\}$ and both $\mu, \sigma$ are defined per pixel. It further assumes a small constant variance $\sigma^2 \approx 0$, reducing the posterior approximation to a *single* image $\mu$ that minimizes the KL divergence:

$$D(q_\lambda(x)||p(x|y)) = -\log p(y|\mu) + D(q_\lambda(x)||p_\theta(x)) \tag{6}$$

RedDiff seeks an image $\mu$ that reconstructs the observation $y$ according to the given mask $m$, while having high probability under the diffusion prior (second term).

The second term acts as a regularizer and decomposes as an expectation that averages over many diffusion time steps. Mardani et al. (2023) find direct optimization of this loss to be very difficult, and observe that annealing time from $t = T$ to $t = 0$, as in standard backward diffusion samplers, yields better performance rather than directly optimizing the variational bound through random time sampling. Some visual examples are provided in Fig. 1 for a comparison between RedDiff-V, which uses random-time sampling as justified by the correct variational bound, and RedDiff which gradually anneals time from $T$ to 0. RedDiff does not propagate gradients through the denoising network $\epsilon_\theta(z_t, t)$, as optimization of the true variational bound would require, to prevent

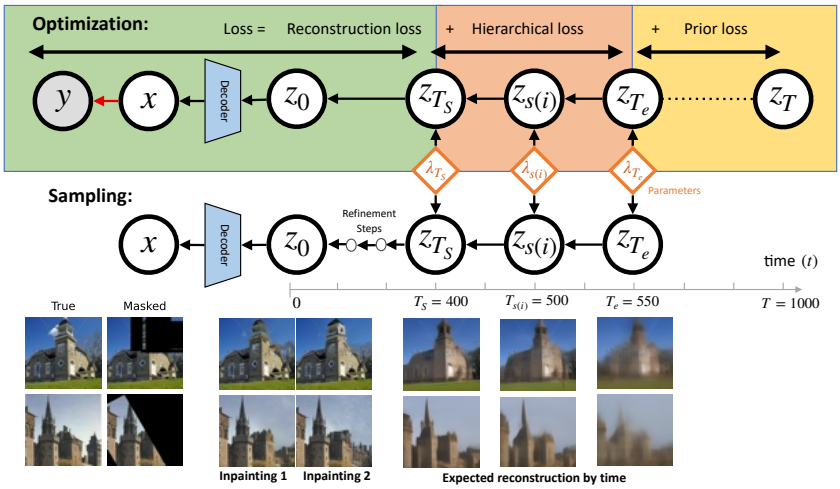


Figure 2: *Top:* The hierarchical approximate posterior of VIPaint is defined over a coarse sequence of intermediate latent steps, or keypoints, between $T_e$ and $T_s$. During optimization, the variational parameters $\lambda$ defining the posterior on a subset of latent times are fit via a prior loss on times above $T_e$, a hierarchical loss defined across $K$ intermediate times, and a reconstruction loss estimated using a one-step approximation $p_\theta(x|z_{T_s})$ from the posterior samples. *Bottom:* After variational inference, samples from the hierarchical posterior (now *aligned* with the observation) transition smoothly in the intermediate latent space $[0, T_s]$ via gradient updates.

optimization instability. We hypothesize that this instability arises due to the denoising function's lack of smoothness at low-noise levels (Yang et al., 2024).

Because RedDiff employs a simple variational posterior that directly optimizes an image at the noise-free ($t = 0$) level only, it is inherently incapable of capturing uncertainty in $x$, and instead seeks a single posterior mode. Additionally, its optimization process is *biased* because it relies on annealing time during the diffusion process rather than randomly sampling time points. We demonstrate that in contrast, our VIPaint framework better models posterior uncertainty, enables stable optimization of an unbiased variational bound, and can be applied to both pixel-based and latent DMs.

## 4 VIPaint: Variational Posterior over Latent Space $z$

Given a pre-trained diffusion model, VIPaint infers the *latent* distribution over $z_t$ induced by a test observation $y$. VIPaint constructs a hierarchical posterior over a subset of mid-ranged latents, $z_t \in [T_s, T_e]$, where $0 < T_s < T_e < T$; optimizes its parameters $\lambda$ using a variational bound; and employs iterative gradient-based refinements to the posterior samples in the low-range latent space $[0, T_s]$ to produce final inpaintings. An overview is provided in Fig. 2.

This technique offers several advantages over RedDiff, as the latent-space hierarchical posterior: 1) infers coarse-to-fine global semantics in the latent space, consistent with the corrupted image $y$; 2) accounts for uncertainty in missing pixels; 3) strategically avoids training instabilities (Yang et al., 2024) which arise in the low-noise latent space $[0, T_s)$; and 4) easily extends to latent DMs. Below, we detail VIPaint's Markov posterior, optimization, and sampling strategies for diverse inpaintings.

**Variational Posterior Formulation.** VIPaint defines the latent-space hierarchical posterior via a set of $K$ *keypoints* which capture the informative phases of the latent diffusion:

$$q_\lambda(z_{T_s:T_e}) = \left( \prod_{i=1}^{K-1} q_\lambda(z_{s(i)} \mid z_{s(i+1)}) \right) q_\lambda(z_{T_e}) \tag{7}$$

Here, $K \geq 2$ and $s(i)$ is the time of the keypoint preceding $s(i+1)$ for all $i \in [1, K-1]$, where $s(1) = T_s$ and $s(K) = T_e$. Experiments suggest tuning these keypoints to capture intermediate-noise timesteps with SNR $(\alpha_t^2/\sigma_t^2)$ in the range $[0.2, 0.5]$ across different (latent) diffusion models. For the highest timestep $T_e$, we let $q_\lambda(z_{T_e})$ be a factorized Gaussian $\mathcal{N}(\mu_{T_e}, \tau_{T_e})$, and the conditionals :

$$q_\lambda(z_{s(i)} \mid z_{s(i+1)}) = \mathcal{N}(z_{s(i)} \mid \gamma_{s(i)} \bar{z}_{s(i)} + (1 - \gamma_{s(i)}) \mu_{s(i)}, \tau^2_{s(i)}). \tag{8}$$

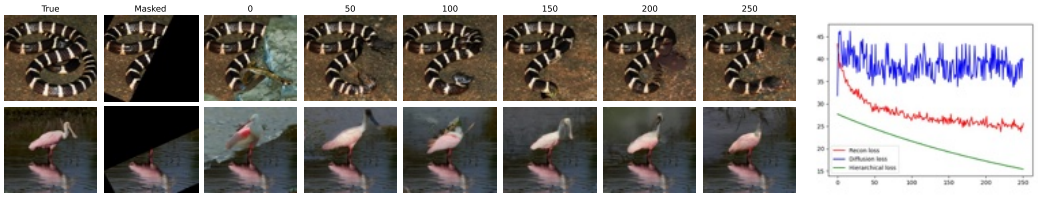


Figure 3: We show VIPaint's posterior fitting progress and sample generation every 50 iterations for two test cases. We see VIPaint quickly grasps the image semantics within 50 optimization iterations.


Figure 4: Image completion results using the LDM prior for Imagenet256 (left) and LSUN (right) with large-mask inpainting (Random Masking and Rotated Window schemes) are shown. DPS, PSLD, and ReSample produce blurry inpaintings of varying quality. Despite being conditioned on class labels, baseline methods' inpaintings for ImageNet are inconsistent with the observed image. In contrast, VIPaint captures global semantics, producing highly realistic inpaintings. See Appendix Fig. 18, 19 for details.

Here, $\tau_{s(i)}$ is the standard deviation (which varies across data dimensions), and the mean is a convex combination of the prior diffusion prediction $\bar{z}_{s(i)} = \hat{z}_\theta(\bar{z}_{s(i+1)}, s(i+1))$, and a contextual variational parameter, $\mu_{s(i)}$. Previous work (Song et al., 2021b; Lugmayr et al., 2022; Kawar et al., 2022; Song et al., 2024) used linear combinations between the observed $y$ and generated sample $z_t$, but employed either hard constraints or fixed weights that are manually tuned. Instead, we incorporate free parameters $\lambda = \{\mu_{T_e}, \tau_{T_e}, (\gamma_{s(i)}, \mu_{s(i)}, \tau_{s(i)})_{i=1}^{K-1}\}$ across $K$ latent levels, defined over each pixel in the image or its encoding. Such a flexible posterior is key to reuse the diffusion prior *and* align precisely with a particular observation $y$, without the need to re-train $\theta$. We use $y$ to initialize $\mu_{s(i)}$ by first encoding it using the encoder and then scaling it by the forward diffusion parameter $\alpha_{s(i)}$, and use the noise schedule to initialize our posterior variance, see Appendix E.1 for details.

**Fitting the Posterior** To fit our hierarchical posterior, we optimize the variational lower bound (VLB) of the marginal likelihood of the observation $y$. The derivation is provided in Appendix C, and the simplified three-term objective is expressed as follows:

$$L(\lambda) = \underbrace{-\mathbb{E}_q[\log p_\theta(y|z_{T_s})]}_{\text{reconstruction loss}} + \beta \underbrace{\mathcal{L}_{(T_e,T)}(z_{T_e})}_{\text{diffusion loss}} + \beta \underbrace{\sum_{i=1}^{K-1} D\Big[q_\lambda(z_{s(i)}|z_{s(i+1)})||p_\theta(z_{s(i)}|z_{s(i+1)}))\Big]}_{\text{hierarchical loss}}. \tag{9}$$

VIPaint seeks latent-posterior distributions that assign high likelihood to the observed features $y$ (by minimizing the reconstruction loss), while simultaneously aligning with the medium-to-high noise levels encoding image semantics (hierarchical and diffusion losses) via weight $\beta > 1$ (Higgins et al., 2017; Agarwal et al., 2023). We approximate $L(\lambda)$ with $M$ Monte Carlo samples from our hierarchical posterior $q_\lambda(z_{T_s:T_e})$; we follow ancestral sampling to draw $z_{T_e}^{(m)} \sim q_\lambda(z_{T_e}), \{z_{s(i)}^{(m)} \sim q_\lambda(z_{s(i)}|z_{s(i+1)}^{(m)})\}_{i=1}^{K-1}$. We evaluate $L(\lambda)$ as below and use automatic differentiation to compute gradients with respect to $\lambda$.

$$\frac{1}{M}\sum_{m=1}^{M}\Bigg[\underbrace{-\log p_\theta(y|z_{T_s}^{(m)})}_{\text{reconstruction loss}} + \beta \underbrace{\mathcal{L}_{(T_e,T)}(z_{T_e}^{(m)})}_{\text{diffusion loss}} + \beta \underbrace{\sum_{i=1}^{K-1} D\Big[q_\lambda(z_{s(i)}|z_{s(i+1)}^{(m)})||p_\theta(z_{s(i)}|z_{s(i+1)}^{(m)}))\Big]}_{\text{hierarchical loss}}\Bigg] \tag{10}$$


Figure 5: Image completion results using the pixel-based EDM model for ImageNet64 with for large-mask inpainting (Random Masking and Rotated Window schemes). We show inpaintings from each method in the following columns. Posterior sampling methods for pixel-based diffusion priors, like RePaint and CoPaint, are generally more accurate but still produces inconsistent samples. In contrast, VIPaint infers the underlying image correctly, and produces highly realistic inpaintings. Additional qualitative plots are in Appendix Fig. 20.

| Task | VIPaint-2 | CoPaint-TT | CoPaint | RePaint | DPS | Blended | RedDiff | RedDiff-V |
|---|---|---|---|---|---|---|---|---|
| Small Mask | 0.090 | **0.079** | 0.089 | <u>0.081</u> | 0.098 | 0.113 | 0.101 | 0.142 |
| Rotated Window | **0.300** | <u>0.316</u> | 0.347 | 0.3213 | 0.3203 | 0.3409 | 0.463 | 0.407 |
| Random Mask | **0.227** | <u>0.245</u> | 0.278 | 0.2575 | 0.2880 | 0.2763 | 0.409 | 0.671 |

| Task | Imagenet-256 | | | | LSUN-Church | | | |
|---|---|---|---|---|---|---|---|---|
| | VIPaint-2 | ReSample | PSLD | DPS | VIPaint-2 | ReSample | PSLD | DPS |
| Rotated Window | **0.392** | <u>0.537</u> | 0.576 | 0.606 | **0.455** | <u>0.510</u> | 0.541 | 0.502 |
| Random Mask | **0.409** | <u>0.559</u> | 0.583 | 0.607 | **0.439** | <u>0.485</u> | 0.523 | 0.490 |
| Small Mask | **0.197** | <u>0.381</u> | 0.534 | 0.564 | **0.299** | <u>0.374</u> | 0.413 | 0.421 |

Table 1: Quantitative results (LPIPS, lower is better) for ImageNet64 for the task of image inpainting using pixel-based EDM prior (*top*) and Imagenet-256 and LSUN-Church using LDM priors (*bottom*). LPIPS is estimated as the mean score of 1000 inpaintings with respect to the true image, averaged across the test set. VIPaint has superior performance (highlighted in **bold**) in nearly all cases. We <u>underline</u> the second best method. Fig. 10 in the appendix has further comparisons.

**Reconstruction Loss.** This term guides the samples from the posterior $z_{T_s}^{(m)}$ to be closer to the observations $y$. We employ a one-step expected mean prediction $\mathbb{E}[\hat{z}_0^{(m)}|z_{T_s}^{(m)}]$ as in Eq. (3) to approximate $z_0$. Because $T_s$ is closer to $t = 0$, this approximation is accurate enough to guide samples $z_{T_s}$ to be consistent with $y$. In case of latent diffusion models, we use decoder upsampling to produce image $\hat{x}$. We use the L1 reconstruction loss, and add a perceptual loss term (Zhang et al., 2018) in case of Latent Diffusion Models. This loss was originally used to train the decoder, and Fig. 8 in Appendix shows an ablation that adding such a term helps avoid blurry reconstructions.

**Diffusion Loss.** We derive the diffusion loss $\mathcal{L}_{(T_e,T)}(z_{T_e}^{(m)})$ in Appendix C, simplified to:

$$\frac{T - T_e}{2}\mathbb{E}_{t,z_t} D[q(z_{t-1}|z_t, z_{T_e}^{(m)})||p_\theta(z_{t-1}|z_t)]. \tag{11}$$

where the expectation is over t uniformly sampled in $[T_e, T], t \sim \mathcal{U}(T_e, T)$ (instead of the entire diffusion timesteps), $z_t \sim q(z_t \mid z_{T_e}^{(m)})$ and $z_{T_e} \sim q_\lambda(z_{T_e})$. In other words, this loss term regularizes the samples $z_{T_e}^{(m)}$ in high-level image semantics encoded in the latent range of $[T_e, T]$. Following prior work, instead of summing this loss over all $t > T_e$, we sample timesteps $t \sim \mathcal{U}(T_e, T)$ defined on a non-uniform discretization (Karras et al., 2022), yielding an unbiased estimate of the loss.

**Hierarchical Loss.** The KL terms across the $K - 1$ intermediate times in the hierarchy is computed in closed form (an analytic function of the means and variances) between the posterior and prior conditional Gaussian distributions. Intuitively, this term further regularizes posterior samples $\{z_{s(i)}^{(m)}\}_{i=1}^{K-1}$ to capture high-to-medium level image details in the mid-ranged $[T_s, T_e]$ diffusion space.

Hence, all the loss terms in Eq. (9) are stochastically and differentiably estimated based on samples from the hierarchical posterior, enabling joint optimization. Progress in fitting VIPaint's posterior is

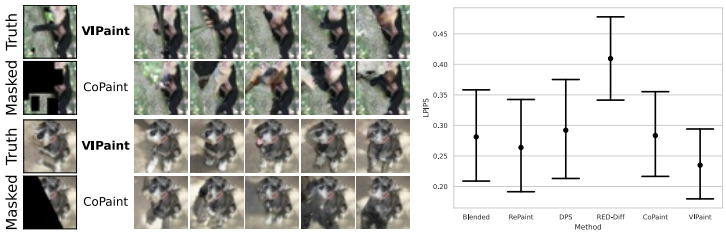


Figure 6: **Left:** Sample completions comparing VIPaint with the best performing baseline, CoPaint, for two test images. We show the true and masked images, and 5 in-painted samples for each method. For an extended comparison see Appendix Fig. 16. VIPaint yields coherent samples while capturing uncertainty in the missing pixels in images. In comparison, CoPaint has high variance in the quality of results. **Right:** We compute summary statistics (minimum, mean, maximum) of the LPIPS score across *100* sampled completions per test image. We show the average value of each of these statistics across the test set. We see that VIPaint improves on baselines, both in terms of the average quality of results *and* in the consistency in result quality.

shown in Fig. 3; the number of optimization steps may be reduced to more quickly give approximate posteriors. If the posterior is only defined on the noise-free level $z_0$ as in RedDiff (Mardani et al., 2023), the VIPaint objective of Eq. (9) degenerates to their (non-annealed) variational objective. However, VIPaint strategically avoids low noise levels in its posterior, avoiding the training instabilities observed by RedDiff and enabling generalization to latent DMs.

**Sampling.** After optimization, the hierarchical posterior $q_\lambda(z_{T_s:T_e})$ is now *semantically* aligned with the observation. We employ ancestral sampling on our $K$ level hierarchical posterior starting from $T_e$ to $T_s$, to yield samples $z_{T_s}$ as shown in Fig. 2. This step gradually adds semantic detials in samples. Further, VIPaint refines $z_{T_s}$ using the prior denoising model at every step $t < T_s$. Similar to DPS (Chung et al., 2023), we update the samples using gradient of the likelihood $\log p_\theta(y \mid z_t), t < T_s$. This ensures fine-grained consistency to our final inpaintings.

## 5 Experiments & Results

### 5.1 Experimental Setup

We conduct experiments across 3 image datasets: LSUN-Church (Yu et al., 2015), ImageNet-64 and ImageNet-256 (Deng et al., 2009). For ImageNet-64, we use the pre-trained class-conditioned pixel-space "EDM" diffusion model (Karras et al., 2022); for LSUN-Churches256 and ImageNet256 we use the pre-trained latent diffusion models from (Rombach et al., 2022). Then, we sample 100 or 1000 *non-cherry-picked* test images across the three datasets. We consider three masking patterns: 1) following RED-Diff: 1000 images using a small mask distribution adapted from Palette Saharia et al. (2022b) that masks up to 30% of each image 2) 100 images using a random mask distribution (Zhao et al., 2021) masking 40-80% of each image, and 3) 100 images using a randomly rotated masking window that masks at least half of the image. By masking large portions of each image we ensure a sufficiently challenging benchmark for inpainting. Acknowledging the pluralistic outcomes in this setting, we evaluate each method across 10 reconstructions per test image, totalling 1000 inpaintings. We use the notation VIPaint-$K$ to denote the number of steps in the hierarchical posterior in our experiments. We found empirically that discretizations and hyperparameters of VIPaint translate well between models using the same noise schedule (as shown for the LSUN and ImageNet-256 latent diffusion models). Please see Appendix E.1 for more details. We test VIPaint for other linear inverse problems like super resolution and Gaussian deblurring in Appendix H.3.

| Task | Imagenet-256, | |
|---|---|---|
| | VIPaint-4 | VIPaint-2 |
| Rotated Window | **0.358** | 0.392 |
| Random Mask | **0.373** | 0.409 |

Table 2: We present an additional setting with $K = 4$, where increasing the number of keypoints in the variational posterior enhances performance for both ImageNet-256 with the LDM prior and ImageNet-64 with the pixel-based EDM prior.

**Comparison.** We compare VIPaint with several recent methods that directly apply the diffusion models trained in the pixel space: *i)* blending methods: *blended* (Song et al., 2021b) and *RePaint*

Lugmayr et al. (2022) ; *ii)* Sampling methods: *DPS* (Chung et al., 2023), and *CoPaint* (Zhang et al., 2023) and *iii)* variational approximations: *RED-Diff* Mardani et al. (2023). Although not exhaustive, this set of methods summarizes recent developments in the state-of-the-art for image inpainting. For latent diffusion models, we compare VIPaint with *DPS*, *PSLD* Rout et al. (2023) and ReSample Song et al. (2024). Please see Appendix E.2 for additional details on their implementation. We report the Peak-Signal-To-Noise-Ratio (PSNR), Kernel Inception Distance (Bińkowski et al., 2018), and the Learned Perceptual Image Patch Similarity (LPIPS) (Zhang et al., 2018) metric in Table 6. We show qualitative images across methods for ImageNet256 in Fig. 18, LSUN-Church in Fig. 19 and ImageNet64 in Fig 20. For tasks like super-resolution and Gaussian Deblurring, we show qualitative results in Fig. 11, 12 and 13 (Appendix). Additionally, we visualize multiple inpaintings in Fig. 6.

### 5.2 Results

**VIPaint enforces consistency with large masking ratios.** Table 6 reports LPIPS scores for the task of image inpainting with small and large masking ratios using pixel and latent-based diffusion models, respectively. We see that all methods perform similarly for small masking ratios. However, for large masks we see a clear improvement with VIPaint. For pixel-based diffusion models, we see that RED-Diff and DPS perform poorly. RePaint, CoPaint and CoPaint-TT show relatively better scores, but do not match VIPaint across any dataset or masking pattern. We show imputations for multiple test examples in Fig. 18, 19 and 20 to highlight differences in inference methods. We see that VIPaint consistently produces plausible inpaintings while other methods fail to complete images for larger masking ratios meaningfully.

| PSLD | CoPaint-TT | RedDiff | VIPaint-2 | VIPaint-4 |
|---|---|---|---|---|
| – | 5.4 | 1.13 | 3.3 | 11.8 |
| 7.0 | – | – | 6.4 | 10 |

Table 3: Table comparing time (in mins) for three inference methods using EDM prior (top) and LDM prior (bottom): Sampling (CoPaint-TT for pixel-based EDM model and PSLD for LDM prior), RedDiff and VIPaint.

**VIPaint yields multiple plausible reconstructions in the case of high uncertainty.** We compare VIPaint with the best performing baseline, CoPaint across multiple sample inpaintings in Fig. 6, a more comprehensive comparison is in Appendix (Fig 15-16). We observe that VIPaint produces multiple visually-plausible imputations while not violating the consistency across observations. We show diversity in possible imputations using different class conditioning using VIPaint in Fig. 17.

**Computational Efficiency.** We report the time taken to produce 10 inpaintings for one test image using: the best performing sampling method: CoPaint-TT (pixel-based EDM prior) and PSLD (LDM prior), RedDiff (pixel-space variational posterior) and VIPaint (latent space variational posterior) in Table 3. RedDiff is fast but inconsistent and unsuitable for latent diffusion priors. Sampling methods are slower, produces better inpaintings than RedDiff but still shows inconsistencies. VIPaint-2 is faster than sampling-based methods for both pixel and latent DMs, *and* achieves better results (see Table 6). VIPaint smoothly trades off time and sample quality, with VIPaint-4 converging slowly but ultimately yielding the best solutions as shown in Table 2. For further details, see Appendix. F.

**VIPaint extends to general linear inverse problems.** We report a quantitative analysis in Table 7 and qualitative results in Fig. 11, 12, and 13. In addition to the LPIPS scores, we also compute the PSNR metrics, averaged over 10 random samples for each of 100 test images. We see that VIPaint shows strong advantages over ReSample and DPS for complex image datasets like ImageNet.

## 6 Conclusion

We present VIPaint, a simple and a general approach to adapt diffusion models for image inpainting and other inverse problems. We take widely used (latent) diffusion generative models, allocate variational parameters for the latent codes of each partial observation, and fit the parameters stochastically to optimize the induced variational bound. The simple but flexible structure of our bounds allows VIPaint to outperform previous sampling and variational methods when uncertainty is high.

### Acknowledgments

This research supported in part by NSF Robust Intelligence Award No. IIS-1816365 and ONR Award No. N00014-23-1-2712, and by the Hasso Plattner Institute (HPI) Research Center in Machine Learning and Data Science at UC Irvine.

## REFERENCES


Sakshi Agarwal, Gabriel Hope, Ali Younis, and Erik B. Sudderth. A decoder suffices for query-adaptive variational inference. In Robin J. Evans and Ilya Shpitser (eds.), *Proceedings of the Thirty-Ninth Conference on Uncertainty in Artificial Intelligence*, volume 216 of *Proceedings of Machine Learning Research*, pp. 33–44. PMLR, 31 Jul–04 Aug 2023. URL https://proceedings.mlr.press/v216/agarwal23a.html.

Omri Avrahami, Dani Lischinski, and Ohad Fried. Blended diffusion for text-driven editing of natural images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 18208–18218, June 2022.

Mikołaj Bińkowski, Dougal J. Sutherland, Michael Arbel, and Arthur Gretton. Demystifying MMD GANs. In *International Conference on Learning Representations*, 2018. URL https://openreview.net/forum?id=r1lUOzWCW.

David M Blei, Alp Kucukelbir, and Jon D McAuliffe. Variational inference: A review for statisticians. *Journal of the American statistical Association*, 112(518):859–877, 2017.

Gabriel Cardoso, Yazid Janati el idrissi, Sylvain Le Corff, and Eric Moulines. Monte carlo guided denoising diffusion models for bayesian linear inverse problems. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=nHESwXvxWK.

YooJung Choi, Antonio Vergari, and Guy Van den Broeck. Probabilistic circuits: A unifying framework for tractable probabilistic models. oct 2020. URL http://starai.cs.ucla.edu/papers/ProbCirc20.pdf.

Hyungjin Chung, Byeongsu Sim, Dohoon Ryu, and Jong Chul Ye. Improving diffusion models for inverse problems using manifold constraints. In Alice H. Oh, Alekh Agarwal, Danielle Belgrave, and Kyunghyun Cho (eds.), *Advances in Neural Information Processing Systems*, 2022a. URL https://openreview.net/forum?id=nJJjv0JDJju.

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

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.

Hyungjin Chung, Jong Chul Ye, Peyman Milanfar, and Mauricio Delbracio. Prompt-tuning latent diffusion models for inverse problems, 2024. URL https://openreview.net/forum?id=ckzglrAMsh.

Ciprian Corneanu, Raghudeep Gadde, and Aleix M. Martinez. Latentpaint: Image inpainting in latent space with diffusion models. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pp. 4334–4343, January 2024.

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pp. 248–255. Ieee, 2009.

Prafulla Dhariwal and Alexander Quinn Nichol. Diffusion models beat GANs on image synthesis. In A. Beygelzimer, Y. Dauphin, P. Liang, and J. Wortman Vaughan (eds.), *Advances in Neural Information Processing Systems*, 2021. URL https://openreview.net/forum?id=AAWuCvzaVt.

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.

Berthy T. Feng, Jamie Smith, Michael Rubinstein, Huiwen Chang, Katherine L. Bouman, and William T. Freeman. Score-based diffusion models as principled priors for inverse imaging. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 10520–10531, October 2023.

Rob Fergus, Barun Singh, Aaron Hertzmann, Sam T. Roweis, and William T. Freeman. Removing camera shake from a single photograph. *ACM Trans. Graph.*, 25(3):787–794, July 2006. ISSN 0730-0301. doi: 10.1145/1141911.1141956. URL https://doi.org/10.1145/1141911.1141956.

Alexandros Graikos, Nikolay Malkin, Nebojsa Jojic, and Dimitris Samaras. Diffusion models as plug-and-play priors. In Alice H. Oh, Alekh Agarwal, Danielle Belgrave, and Kyunghyun Cho (eds.), *Advances in Neural Information Processing Systems*, 2022. URL https://openreview.net/forum?id=yhlMZ3iR7Pu.

Irina Higgins, Loic Matthey, Arka Pal, Christopher Burgess, Xavier Glorot, Matthew Botvinick, Shakir Mohamed, and Alexander Lerchner. beta-VAE: Learning basic visual concepts with a constrained variational framework. In *International Conference on Learning Representations*, 2017. URL https://openreview.net/forum?id=Sy2fzU9gl.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin (eds.), *Advances in Neural Information Processing Systems*, volume 33, pp. 6840–6851. Curran Associates, Inc., 2020a. URL https://proceedings.neurips.cc/paper_files/paper/2020/file/4c5bcfec8584af0d967f1ab10179ca4b-Paper.pdf.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin (eds.), *Advances in Neural Information Processing Systems*, volume 33, pp. 6840–6851. Curran Associates, Inc., 2020b. URL https://proceedings.neurips.cc/paper_files/paper/2020/file/4c5bcfec8584af0d967f1ab10179ca4b-Paper.pdf.

Geng Ji, Michael C Hughes, and Erik B Sudderth. From patches to images: A nonparametric generative model. In *International Conference on Machine Learning*, pp. 1675–1683. PMLR, 2017.

Zahra Kadkhodaie and Eero Simoncelli. Stochastic solutions for linear inverse problems using the prior implicit in a denoiser. In M. Ranzato, A. Beygelzimer, Y. Dauphin, P.S. Liang, and J. Wortman Vaughan (eds.), *Advances in Neural Information Processing Systems*, volume 34, pp. 13242–13254. Curran Associates, Inc., 2021. URL https://proceedings.neurips.cc/paper_files/paper/2021/file/6e28943943dbed3c7f82fc05f269947a-Paper.pdf.

Tero Karras, Miika Aittala, Timo Aila, and Samuli Laine. Elucidating the design space of diffusion-based generative models. In *Proc. NeurIPS*, 2022.

Bahjat Kawar, Michael Elad, Stefano Ermon, and Jiaming Song. Denoising diffusion restoration models. In *Advances in Neural Information Processing Systems*, 2022.

Diederik Kingma, Tim Salimans, Ben Poole, and Jonathan Ho. Variational diffusion models. In *Advances in Neural Information Processing Systems*, 2021a.

Diederik P Kingma and Max Welling. An introduction to variational autoencoders. *Foundations and Trends in Machine Learning*, 12(4):307–392, 2019.

Diederik P Kingma, Tim Salimans, Ben Poole, and Jonathan Ho. On density estimation with diffusion models. In A. Beygelzimer, Y. Dauphin, P. Liang, and J. Wortman Vaughan (eds.), *Advances in Neural Information Processing Systems*, 2021b. URL https://openreview.net/forum?id=2LdBqxc1Yv.

Anji Liu, Mathias Niepert, and Guy Van den Broeck. Image inpainting via tractable steering of diffusion models. In *The Twelfth International Conference on Learning Representations*, 2024a. URL https://openreview.net/forum?id=NSIVHTbZBR.

Anji Liu, Mathias Niepert, and Guy Van den Broeck. Image inpainting via tractable steering of diffusion models. In *The Twelfth International Conference on Learning Representations*, 2024b. URL https://openreview.net/forum?id=NSIVHTbZBR.

Andreas Lugmayr, Martin Danelljan, Andrés Romero, Fisher Yu, Radu Timofte, and Luc Van Gool. Repaint: Inpainting using denoising diffusion probabilistic models. *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 11451–11461, 2022. URL https://api.semanticscholar.org/CorpusID:246240274.

Morteza Mardani, Jiaming Song, Jan Kautz, and Arash Vahdat. A variational perspective on solving inverse problems with diffusion models, 2023.

Morteza Mardani, Jiaming Song, Jan Kautz, and Arash Vahdat. A variational perspective on solving inverse problems with diffusion models. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=1YO4EE3SPB.

Chenlin Meng, Yutong He, Yang Song, Jiaming Song, Jiajun Wu, Jun-Yan Zhu, and Stefano Ermon. Sdedit: Guided image synthesis and editing with stochastic differential equations. *arXiv preprint arXiv:2108.01073*, 2021.

Alex Nichol, Prafulla Dhariwal, Aditya Ramesh, Pranav Shyam, Pamela Mishkin, Bob McGrew, Ilya Sutskever, and Mark Chen. Glide: Towards photorealistic image generation and editing with text-guided diffusion models, 2022.

Alexander Quinn Nichol and Prafulla Dhariwal. Improved denoising diffusion probabilistic models. In Marina Meila and Tong Zhang (eds.), *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pp. 8162–8171. PMLR, 18–24 Jul 2021. URL https://proceedings.mlr.press/v139/nichol21a.html.

Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and BjÃ¶rn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022. URL https://github.com/CompVis/latent-diffusionhttps://arxiv.org/abs/2112.10752.

Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In Nassir Navab, Joachim Hornegger, William M. Wells, and Alejandro F. Frangi (eds.), *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015*, pp. 234–241, Cham, 2015. Springer International Publishing. ISBN 978-3-319-24574-4.

Litu Rout, Negin Raoof, Giannis Daras, Constantine Caramanis, Alex Dimakis, and Sanjay Shakkottai. Solving linear inverse problems provably via posterior sampling with latent diffusion models. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023. URL https://openreview.net/forum?id=XKBFdYwfRo.

Chitwan Saharia, William Chan, Huiwen Chang, Chris Lee, Jonathan Ho, Tim Salimans, David Fleet, and Mohammad Norouzi. Palette: Image-to-image diffusion models. In *ACM SIGGRAPH 2022 Conference Proceedings*, SIGGRAPH '22, New York, NY, USA, 2022a. Association for Computing Machinery. ISBN 9781450393379. doi: 10.1145/3528233.3530757. URL https://doi.org/10.1145/3528233.3530757.

Chitwan Saharia, William Chan, Huiwen Chang, Chris Lee, Jonathan Ho, Tim Salimans, David Fleet, and Mohammad Norouzi. Palette: Image-to-image diffusion models. In *ACM SIGGRAPH 2022 conference proceedings*, pp. 1–10, 2022b.

Jascha Sohl-Dickstein, Eric Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. In Francis Bach and David Blei (eds.), *Proceedings of the 32nd International Conference on Machine Learning*, volume 37 of *Proceedings of Machine Learning Research*, pp. 2256–2265, Lille, France, 07–09 Jul 2015. PMLR. URL https://proceedings.mlr.press/v37/sohl-dickstein15.html.

Bowen Song, Soo Min Kwon, Zecheng Zhang, Xinyu Hu, Qing Qu, and Liyue Shen. Solving inverse problems with latent diffusion models via hard data consistency. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=j8hdRqOUhN.

Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. In *International Conference on Learning Representations*, 2021a. URL https://openreview.net/forum?id=St1giarCHLP.

Jiaming Song, Arash Vahdat, Morteza Mardani, and Jan Kautz. Pseudoinverse-guided diffusion models for inverse problems. In *International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=9_gsMA8MRKQ.

Yang Song and Stefano Ermon. Generative modeling by estimating gradients of the data distribution. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett (eds.), *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019. URL https://proceedings.neurips.cc/paper_files/paper/2019/file/3001ef257407d5a371a96dcd947c7d93-Paper.pdf.

Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. In *International Conference on Learning Representations*, 2021b. URL https://openreview.net/forum?id=PxTIG12RRHS.

Yang Song, Liyue Shen, Lei Xing, and Stefano Ermon. Solving inverse problems in medical imaging with score-based generative models. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=vaRCHVj0uGI.

Brian L. Trippe, Jason Yim, Doug Tischer, David Baker, Tamara Broderick, Regina Barzilay, and Tommi S. Jaakkola. Diffusion probabilistic modeling of protein backbones in 3d for the motif-scaffolding problem. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=6TxBxqNME1Y.

Arash Vahdat, Karsten Kreis, and Jan Kautz. Score-based generative modeling in latent space. In *Neural Information Processing Systems (NeurIPS)*, 2021.

M. J. Wainwright and M. I. Jordan. Graphical models, exponential families, and variational inference. *Foundations and Trends in Machine Learning*, 1:1–305, 2008.

Hao Wang, Yongsheng Yu, Tiejian Luo, Heng Fan, and Libo Zhang. MaGIC: Multi-modality guided image completion. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=o7x0XVlCpX.

Yinhuai Wang, Jiwen Yu, and Jian Zhang. Zero-shot image restoration using denoising diffusion null-space model. *The Eleventh International Conference on Learning Representations*, 2023a.

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*, 2023b. URL https://openreview.net/forum?id=mRieQgMtNTQ.

Zhantao Yang, Ruili Feng, Han Zhang, Yujun Shen, Kai Zhu, Lianghua Huang, Yifei Zhang, Yu Liu, Deli Zhao, Jingren Zhou, and Fan Cheng. Lipschitz singularities in diffusion models. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=WNkW0cOwiz.

Fisher Yu, Yinda Zhang, Shuran Song, Ari Seff, and Jianxiong Xiao. Lsun: Construction of a large-scale image dataset using deep learning with humans in the loop. *arXiv preprint arXiv:1506.03365*, 2015.

Jiwen Yu, Yinhuai Wang, Chen Zhao, Bernard Ghanem, and Jian Zhang. Freedom: Training-free energy-guided conditional diffusion model. *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2023.

Guanhua Zhang, Jiabao Ji, Yang Zhang, Mo Yu, Tommi S. Jaakkola, and Shiyu Chang. Towards coherent image inpainting using denoising diffusion implicit models. *CoRR*, abs/2304.03322, 2023. URL https://doi.org/10.48550/arXiv.2304.03322.

Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *CVPR*, 2018.

Shengyu Zhao, Jonathan Cui, Yilun Sheng, Yue Dong, Xiao Liang, Eric I Chang, and Yan Xu. Large scale image completion via co-modulated generative adversarial networks. In *International Conference on Learning Representations (ICLR)*, 2021.

# A Appendix

# B Diffusion Models: Definition & Training Procedure Recap

## B.1 Forward time Diffusion Process

The background and expressions on forward diffusion process is taken from Kingma et al. (2021b) and included here for completeness. Re-iterating Eq. 1, we have the forward diffusion as:

$$q(z_t \mid x) = \mathcal{N}(\alpha_t x, \sigma_t^2 I). \tag{12}$$

**Forward Conditional $q(z_t|z_s)$:** The distribution $q(z_t|z_s)$ for any $t > s$ are also Gaussian, and from Kingma et al. (2021b), we can re-write as

$$\mathcal{N}(\alpha_{t|s} z_s, \sigma_{t|s}^2 I) \tag{13}$$

$$\text{where,} \quad \alpha_{t|s} = \alpha_t / \alpha_s, \tag{14}$$

$$\text{and,} \quad \sigma_{t|s}^2 = \sigma_t^2 - \alpha_{t|s}^2 \sigma_s^2 \tag{15}$$

**Reverse Conditional, $q(z_s|z_t, x)$:** The posterior $q(z_s|z_t, x)$ from Kingma et al. (2021b) can be written as:

$$q(z_s|z_t, x) = \mathcal{N}(\mu_Q(z_t, x; s, t), \sigma_Q^2(s, t) I) \tag{16}$$

$$\text{where,} \quad \sigma_Q^2(s, t) = \sigma_{t|s}^2 \sigma_s^2 / \sigma_t^2 \tag{17}$$

$$\text{and,} \quad \mu_Q(z_t, x; s, t) = \frac{\alpha_{t|s} \sigma_s^2}{\sigma_t^2} z_t + \frac{\alpha_s \sigma_{t|s}^2}{\sigma_t^2} x \tag{18}$$

## B.2 Reverse Diffusion: Defining $p_\theta(z_s|z_t)$

Here, we describe in detail the conditional reverse model distributions $p_\theta(z_s|z_t)$ for the two cases of variance-exploding and variance preserving diffusion process. Given these formulations, it is straightforward to compute the KL distance between our posterior $q_\lambda(z_s|z_t, y)$ and the prior $p_\theta(z_s|z_t)$ in our loss objective (Eq. 9) since both are conditionally Gausian distributions and computing the KL between two Gaussians can be done in closed form.

**Variance Exploding Diffusion Process.** In this case, $\alpha_t = 1$ and $\sigma_t$ is usually in the range $[0.002, 50]$ Song et al. (2021b). We follow the ancestral sampling rule from the same work to define our prior conditional Gaussian distributions $p_\theta(z_s|z_t)$ :

$$p_\theta(z_s|z_t) = \mathcal{N}(\mu_\theta(z_t; s, t), \sigma_Q^2(s, t) I) \tag{19}$$

$$\text{where,} \quad \sigma_Q^2(s, t) = (\sigma_t^2 - \sigma_s^2) \frac{\sigma_s^2}{\sigma_t^2} \tag{20}$$

$$\text{and,} \quad \mu_\theta(z_t; s, t) = \frac{\sigma_s^2}{\sigma_t^2} z_t + \frac{\sigma_t^2 - \sigma_s^2}{\sigma_t^2} \hat{x}_\theta(z_t, t) \tag{21}$$

where $\hat{x}_\theta(z_t, t) = z_t - \sqrt{(\sigma_t^2 - \sigma_s^2)} * \epsilon_\theta(z_t, t)$

**Variance Preserving Diffusion Process.** In this case, $\alpha_t = \sqrt{1-\sigma_t^2}$ and $\sigma_t^2$ is usually in the range $[0.001, 1]$ Ho et al. (2020b). We follow the DDIM sampling rule Song et al. (2021a) to define our prior conditional Gaussian distributions $p_\theta(z_s|z_t)$. This sampling rule is widely used to generate unconditional samples in small number of steps, and naturally becomes a key design choice of our prior. Here,

$$p_\theta(z_s|z_t) = \mathcal{N}(\mu_\theta(z_t; s, t), \sigma_Q^2(s,t)I) \tag{22}$$

$$\text{where,} \quad \sigma_Q^2(s,t) = \eta(\frac{1-\alpha_{t-1}}{1-\alpha_t})(1-\frac{\alpha_t}{\alpha_{t-1}}) \tag{23}$$

$$\text{and,} \quad \mu_\theta(z_t; s, t) = \sqrt{\alpha_{t-1}}\hat{x}_\theta(z_t, t) + \sqrt{1-\alpha_t-\sigma_t^2}\epsilon_\theta(z_t, t) \tag{24}$$

where $\hat{x}_\theta(z_t, t) = \frac{z_t - \sqrt{1-\alpha_t}\epsilon_\theta(z_t,t)}{\sqrt{\alpha_t}}$. This schedule is adopted by the Latent Diffusion models.

### B.3 Derivation of Objective for training Diffusion Models: $\mathcal{L}_{(0,T)}(z_0)$

The usual variational bound on the negative loglikehood on data $x$:

$\mathbb{E}[-\log p_\theta(x)] \leq \mathbb{E}_q[-\log \frac{p_\theta(z_{0:T})}{q(z_{1:T}|z_0,x)}] = \mathbb{E}_q[-\log p(z_T) - \sum_{t=1}^T \log \frac{p_\theta(z_{t-1}|z_t)}{q(z_t|z_{t-1})}]$.

Let $0 < s < t < T$, we expand this derivation from Ho et al. (2020b) as follows:

$$\mathcal{L} = \mathbb{E}_q\left[\log \frac{q(z_{1:T}|z_0)}{p_\theta(z_0)}\right] \tag{25}$$

$$= \mathbb{E}_q\left[-\log p(z_T) + \sum_{t\geq 1} \log \frac{q(z_t|z_s)}{p_\theta(z_s|z_t)}\right] \tag{26}$$

$$= \mathbb{E}_q\left[-\log p(z_T) + \sum_{t>1} \log \frac{q(z_t|z_s)}{p_\theta(z_s|z_t)} + \log \frac{q(z_1|z_0)}{p_\theta(z_0|z_1)}\right] \tag{27}$$

$$= \mathbb{E}_q\left[-\log p(z_T) + \sum_{t>1} \log \frac{q(z_s|z_t,z_0)}{p_\theta(z_s|z_t)} \cdot \frac{q(z_t|z_0)}{q(z_s|z_0)} + \log \frac{q(z_1|z_0)}{p_\theta(z_0|z_1)}\right] \tag{28}$$

$$= \mathbb{E}_q\Bigg[-\log \frac{p(z_T)}{q(z_T|z_0)} + \underbrace{\sum_{t>1} \log \frac{q(z_s|z_t,z_0)}{p_\theta(z_s|z_t)}}_{\text{diffusion loss } \mathcal{L}_{(0,T)}(z_0)} - \log p_\theta(z_0|z_1)\Bigg] \tag{29}$$

### B.4 Inverse Problems

### B.5 Sampling Methods for Inverse Problems

**Blending Methods** methods (Song et al., 2022; Wang et al., 2023a) define a procedural, heuristic approximation to the posterior and is tailored for image inpainting. They first generate unconditional samples $z_{t-1}$ from the prior using the learned noise prediction network, and then incorporate $y$ by replacing the corresponding dimensions with the observed measurements. RePaint (Lugmayr et al., 2022) attempts to reduce visual inconsistencies caused by blending via a resampling strategy. A "time travel" operation is introduced, where images from the current time step $z_{t-1}$ are first blended with the noisy version of the observed image $y_{t-1}$, and then used to generate images in the $(t-1)+r, (r \geq 1)$ time step by applying a one-step forward process and following the Blended denoising process.

**Gradient-Based Methods.** Motivated by the goal of addressing more general inverse problems, Diffusion Posterior Sampling (*DPS*) (Chung et al., 2023) uses Bayes' Rule to sample from $p_\theta(z_{t-1}|z_t, y) \propto p_\theta(z_{t-1}|z_t)p_\theta(y|z_{t-1})$. Instead of directly blending or replacing images with noisy

versions of the observation, DPS uses the gradient of the likelihood $\log p_\theta(y|z_t)$ to guide the generative process at every denoising step $t$. Since computing $\nabla_{z_t} \log p(y|z_{t-1})$ is intractable due to the integral over all possible configurations of $z_{t'}$ for $t' < t-1$, DPS approximates $p(y|z_{t-1})$ using a one-step denoised prediction $\hat{x}$ using Eq. (3). The likelihood $p(y|x) = \mathcal{N}(f(x), \sigma_v^2)$ can then be evaluated using these approximate predictions. To obtain the gradient of the likelihood term, DPS require backpropagating gradients through the denoising network used to predict $\hat{x}$.

Specializing to image inpainting, *CoPaint* (Zhang et al., 2023) augments the likelihood with another regularization term to generate samples $z_{t-1}$ that prevent taking large update steps away from the previous sample $z_t$, in an attempt to produce more coherent images. Further, it proposes CoPaint-TT, which additionally uses the time-travel trick to reduce discontinuities in sampled images.

Originally designed for pixel-space diffusion models, it is difficult to adopt these works directly to latent diffusion models. Posterior Sampling with Latent Diffusion (*PSLD*) (Rout et al., 2023) first showed that employing *DPS* directly on latent space diffusion models produces blurry images. It proposes to add another "gluing" term to the measurement likelihood which penalizes samples $z_t$ that do not lie in the encoder-decoder shared embedding space. However, this may produce artifacts in the presence of measurement noise (see Song et al. (2024)). To address this issue, recent concurrent work on the *ReSample* (Song et al., 2024) method divides the timesteps in the latent space into 3 subspaces, and optimizes samples $z_t$ in the mid-subspace to encourage samples that are more consistent with observations. Other work (Yu et al., 2023) highlights a 3-stage approach where data consistency can be enforced in the latter 2 stages which are closer to $t = 0$.

# C VIPAINT: VI METHOD USING DIFFUSION MODELS AS PRIORS

## C.1 DERIVATION OF VIPAINT'S TRAINING OBJECTIVE

As specified in the main paper, we define a variational distribution over the latent space variable $z$ as $q_\lambda(z)$ and re-use the diffusion prior to generate $x \sim p_\theta(x \mid z)$. We derive the variational objective here:

$$
\begin{aligned}
&\mathcal{L}_N(\lambda; y) \\
&= \mathbb{E}_{q_\lambda(z,x)}[\log p_\theta(y, x, z) - \log q_\lambda(z, x \mid y)] \\
&= \mathbb{E}_{q_\lambda(z,x)}[\log p_\theta(z) + \log p_\theta(x \mid z_{T_s}) + \log p_\theta(y \mid z_{T_s}) - \log q_\lambda(z) - \log q_\lambda(x \mid z_{T_s})] \\
&= \mathbb{E}_{q_\lambda(z)}[\log p_\theta(y \mid z_{T_s}) + \log p_\theta(z) - \log q_\lambda(z)] \\
&= \mathbb{E}_{q_\lambda(z)}[\log p_\theta(y \mid z_{T_s})] - \mathbb{E}_{q_\lambda(z)}[\log q_\lambda(z) - \log p_\theta(z)] \\
&= \mathbb{E}_{q_\lambda(z)}[\log p_\theta(y \mid z_{T_s})] - \underbrace{\mathbb{E}_{q_\lambda(z)}[\log q_\lambda(z) - \log p_\theta(z)]}_{\text{second term}}
\end{aligned} \tag{30}
$$

The second term can be further decomposed as:

$$
\begin{aligned}
&= \mathbb{E}_{q_\lambda(z)}[\sum_{i=0}^{K-1} \log q_\lambda(z_{s(i)} \mid z_{s(i+1)}) + \log q_\lambda(z_{T_e}) - \sum_{i=0}^{K-1} \log p_\theta(z_{s(i)} \mid z_{s(i+1)}) - \log p_\theta(z_{T_e})] \\
&= \sum_{i=0}^{K-1} D[q_\lambda(z_{s(i)} \mid z_{s(i+1)}) || p_\theta(z_{s(i)} \mid z_{s(i+1)})] - \underbrace{D(q(z_{T_e}) || p(z_{z_{T_e}}))}_{\mathcal{L}_{(T_e,T)}(z_{T_e})}
\end{aligned} \tag{31}
$$

Finally, $\mathcal{L}_N(\lambda; y)$

$$
= \mathbb{E}_{q_\lambda(z)}[\log p_\theta(y \mid z_{T_s})] - \sum_{i=0}^{K-1} D[q_\lambda(z_{s(i)} \mid z_{s(i+1)}) || p_\theta(z_{s(i)} \mid z_{s(i+1)})] - \underbrace{D(q(z_{T_e}) || p(z_{z_{T_e}}))}_{\mathcal{L}_{(T_e,T)}(z_{T_e})} \tag{32}
$$

Negating the above objective, we get Eq. 9 in the main paper. Now, let's derive the third term $L_{(T_e,T)}(z_{T_e})$ following section B.3

## C.2 DERIVATION OF $L_{(T_e,T)}(z_{T_e})$

For any $T_e < s < t < T$, we have :

$$
\mathbb{E}_{z_{T_e} \sim q_\lambda(z_{T_e})} \left[ \log \frac{q(z_{T_e+1:T} | z_{T_e})}{p_\theta(z_{T_e:T})} \right] \tag{33}
$$

$$
= \mathbb{E}_{z_{T_e} \sim q_\lambda(z_{T_e})} \left[ -\log p(z_T) + \sum_{t \geq T_e} \log \frac{q(z_t|z_s)}{p_\theta(z_s|z_t)} \right] \tag{34}
$$

$$
= \mathbb{E}_{z_{T_e} \sim q_\lambda(z_{T_e})} \left[ -\log p(z_T) + \sum_{t > T_e} \log \frac{q(z_t|z_s)}{p_\theta(z_s|z_t)} + \log \frac{q(z_{T_e+1}|z_{T_e})}{p_\theta(z_{T_e}|z_{T_e+1})} \right] \tag{35}
$$

$$
= \mathbb{E}_{z_{T_e} \sim q_\lambda(z_{T_e})} \left[ -\log p(z_T) + \sum_{t > T_e} \log \frac{q(z_s|z_t, z_{T_e})}{p_\theta(z_s|z_t)} \cdot \frac{q(z_t|z_{T_e})}{q(z_s|z_{T_e})} + \log \frac{q(z_{T_e+1}|z_{T_e})}{p_\theta(z_{T_e}|z_{T_e+1})} \right] \tag{36}
$$

$$
= \mathbb{E}_{z_{T_e} \sim q_\lambda(z_{T_e})} \left[ -\log \frac{p(z_T)}{q(z_T|z_{T_e})} + \underbrace{\sum_{t > T_e} \log \frac{q(z_s|z_t, z_{T_e})}{p_\theta(z_s|z_t)}}_{\text{diffusion loss } \mathcal{L}_{(T_e,T)}(z_{T_e})} - \log p_\theta(z_{T_e}|z_{T_e+1}) \right] \tag{37}
$$

The first and third term can be stochastically and differentially estimated using standard techniques. Following Kingma et al. (2021b), we derive an estimator for the diffusion loss $\mathcal{L}_{(T_e,T)}(z_{T_e})$. In the case of finite timesteps $t > T_e$, this loss is:

$$\mathcal{L}_{(T_e,T)}(z_{T_e}) = \sum_{t>T_e} \mathbb{E}_{q(z_t|z_{T_e})} D[q(z_s|z_t, z_{T_e})||p_\theta(z_s|z_t)] \tag{38}$$

**Estimator of $\mathcal{L}_{(T_e,T)}(z_{T_e})$** Reparametering $z_t \sim q(z_t|z_{T_e})$ as $z_t = \alpha_{t|T_e} z_{T_e} + \sigma_{t|T_e}\epsilon$, where $\epsilon \sim \mathcal{N}(0,1)$, and to avoid having to compute all $T - T_e$ terms when calculating the diffusion loss, we construct an unbiased estimator of $\mathcal{L}_{(T_e,T)}(z_{T_e})$ using

$$\mathcal{L}_{(T_e,T)}(z_{T_e}) = \frac{T - T_e}{2} \mathbb{E}_{\epsilon,t\sim\mathcal{U}(T_e,T)}[D(q(z_s|z_t, z_{T_e})||p_\theta(z_s|z_t))] \tag{39}$$

where $\mathcal{U}(T_e, T)$ is a uniform distribution to sample $T_e < t \leq T$ from a non-uniform descretization of timesteps using Karras et al. (2022).

Now, we elaborate on the expression $q(z_s|z_t, z_{T_e})$ and $p(z_s|z_t)$ for any $T_e < s < t < T$.

### C.2.1 $q(z_s|z_t, z_{T_e})$

Our posterior at $T_e$ is $q(z_{T_e}) = \mathcal{N}(\mu_{T_e}, \tau^2_{T_e})$. For any $T_e < s < t < T$, we have $q(z_s|z_{T_e}) = \mathcal{N}(\alpha_{s|T_e} z_{T_e}, \tau^2_{s|T_e})$ and $q(z_t|z_s) = \mathcal{N}(\alpha_{t|s} z_s, \sigma^2_{t|s})$, yielding the posterior :

$$q(z_s|z_t, z_{T_e}) = \mathcal{N}(\mu_Q(z_t, z_{T_e}; s, t, T_e), \sigma^2_Q(s, t, T_e)I) \tag{40}$$

$$\text{where,} \quad \sigma^2_Q(s, t, T_e) = \sigma^2_{t|s} \frac{\tau^2_{s|T_e}}{\sigma^2_{t|s} + \alpha^2_{s|T_e}\tau^2_{s|T_e}} \tag{41}$$

$$\text{and,} \quad \mu_Q(z_t, z_{T_e}; s, t, T_e) = \sigma^2_Q \left( \frac{\alpha_{s|T_e}}{\tau^2_{s|T_e}} z_{T_e} + \frac{\alpha_{t|s}}{\sigma^2_{t|s}} z_t \right) \tag{42}$$

$$= \frac{\alpha_{s|T_e}\sigma^2_{t|s}}{(\sigma^2_{t|s} + \alpha^2_{s|T_e}\tau^2_{s|T_e})} z_{T_e} + \frac{\alpha_{t|s}\tau^2_{s|T_e}}{(\sigma^2_{t|s} + \alpha^2_{s|T_e}\tau^2_{s|T_e})} z_t \tag{43}$$

### C.2.2 $p(z_s|z_t)$

The conditional model distributions can be chosen as:

$$p_\theta(z_s|z_t) = q(z_s|z_t, z_{T_e} = \hat{z}_{\theta,T_e}(z_t, t)) = \mathcal{N}(z_s; \mu_\theta(z_t, z_{T_e}; s, t, T_e), \sigma^2_Q(s, t, T_e)) \tag{44}$$

$$\text{where,} \quad \mu_\theta(z_t, z_{T_e}; s, t, T_e) = \frac{\alpha_{s|T_e}\sigma^2_{t|s}}{(\sigma^2_{t|s} + \alpha^2_{s|T_e}\tau^2_{s|T_e})} \hat{z}_{\theta,T_e}(z_t, t) + \frac{\alpha_{t|s}\tau^2_{s|T_e}}{(\sigma^2_{t|s} + \alpha^2_{s|T_e}\tau^2_{s|T_e})} z_t \tag{45}$$

$$\text{and,} \quad \sigma^2_Q(s, t, T_e) = \sigma^2_{t|s} \frac{\sigma^2_{s|T_e}}{\sigma^2_{t|s} + \alpha^2_{s|T_e}\sigma^2_{s|T_e}} \tag{46}$$

where $\hat{z}_{\theta,T_e}(z_t, t) = \frac{z_t - \sigma_{t|T_e} * \epsilon_\theta(z_t,t)}{\alpha_{t|T_e}}$

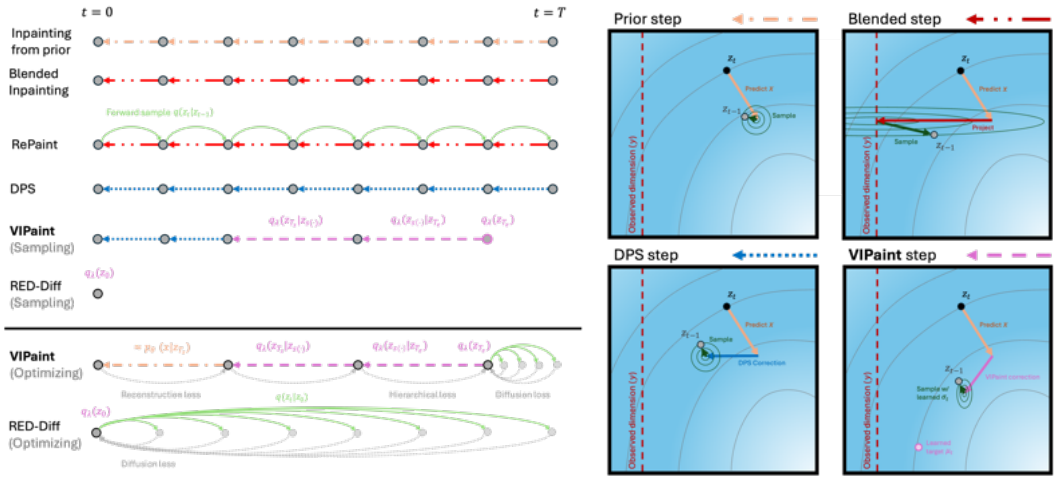


Figure 7: Expanded comparison of methods for diffusion model-based inpainting. **Left:** Timeline illustration of sampling steps with time flowing rightward from $t = 0$ (clean images) to $t = T$ (pure noise). Orange arrows indicate a single step of ancestral sampling under the generative prior $p_\theta(z_{t-1}|z_t)$. Red arrows indicate a single step of the *Blended* approximation of $p_\theta(z_{t-1}|z_t, y)$, while blue arrows indicate a single step of the *DPS* approximation. Green arrows indicate steps forward in time according to the diffusion process $q(z_t|z_{<t})$. Methods such as *RePaint* and *CoPaint* alternate between forward and reverse steps. Purple arrows indicate sampling from a step in the hierarchical **VIPaint** posterior $q_\lambda(z_{s(i-1)}|z_{s(i)})$. Both **VIPaint** and *RED-Diff* (without annealing) involve an initial optimization stage to fit variational parameters per-image. Gray arrows indicate the flow of gradient information during this optimization stage. Gray points are steps only used during optimization. **Right:** Illustration of each reverse-time sampling step in 2 dimensions. The horizontal dimension is assumed to be observed at the value marked by the red line. Each approach begins by computing $p_\theta(z_{t-1}|z_t)$ via a prediction of $x$ using the pre-trained denoising network $\hat{x}_\theta(z_t, t)$. *Blended* replaces observed dimensions with $q(z_{t-1}|y)$. *DPS* updates $p_\theta(z_{t-1}|z_t)$ according to a single-step approximation to the likelihood $p_\theta(y|z_{t-1})$. Finally, **VIPaint**, uses a learned variational distribution $q_\lambda(z_{t-1}|z_t)$, which can be seen as interpolating between the prediction of $x$ and a variational parameter $\mu_t$, coupled with a learned variance.

# D EXPANDED FIGURE 7

# E EXPERIMENTAL DETAILS

## E.1 VIPAINT

**Choosing $(T_s, T_e)$ for VIPaint** Extensive prior work Song et al. (2021b); Nichol & Dhariwal (2021); Dhariwal & Nichol (2021); Karras et al. (2022) explores different noise schedules for training diffusion models, and how it affects the generated image quality. Since we use these diffusion models incorporating different noise schedules, our latent hierarchical posterior needs to account for this shift, and we show that it is flexible to do so. To concentrate posterior inference on the noise levels which are most crucial to perceptual image quality, we define our posterior at intermediate time steps that induce a signal-to-noise-ratio $(\alpha_t^2/\sigma_t^2) \in [0.2, 0.5]$, Kingma et al. (2021b)) approximately across our experiments. This corresponds to choosing $(T_e = 5, T_s = 2)]$ for the pixel-based EDM prior with a variance-exploding noise schedule and $(T_e = 550, T_s = 400)$ ((DDIM sampling coefficient, $\eta = 0.2$) ) for the LDM prior using the VP noise schedule for both LSUN and ImageNet256 datasets. VIPaint is not sensitive to any subset of $K$ timesteps in between this signal-to-noise range. For instance, for VIPaint-4, we take $[T_e = 5, 4, 3.5, 2.5, T_s = 2]$ for the EDM noise schedule and $[T_e = 550, 500, 450, T_s = 400]$ for the LDM prior.

**Choosing $K$** We discuss in section F that for a $K$ step hierarchical posterior, the optimization run-time of VIPaint is O($KI$), when $I$ is the total number of optimization steps. From the experiments conducted, we see that $K$ can be easily selected to trade-off time and sample quality.

**Initialization** We follow the forward and reverse diffusion process defined by each VE and VP noise schedules to initialize VIPaint's variational parameters. For LDM prior, we use the lower dimensional encoding of $y$. We provide a comprehensive summary in Table 4.

Table 4: Initialization of Variational Parameters for VE and VP Schedules

| VI Parameters | VP Schedule | VE Schedule |
|---|---|---|
| $\mu_{T_e} = \alpha_{T_e} y + a_1 \sigma_{T_e} \epsilon$ (Scale factor to retain information from $y$.) | $a_1 = 0.8$ | $a_1 = 0.01$ |
| $\mu_{s(i)} = \alpha_{s(i)} y + a_2 \sigma_{s(i)} \epsilon$ (Noise adding process is still quite high for VE schedules.) | $a_2 = 1$ | $a_2 = 0.01$ |
| $\tau_{T_e} = \sigma_{T_e}$ (From the forward diffusion process. ) | – | – |
| $\tau_{s(i)\|s(i+1)}$ (From the reverse diffusion process.) | Eq. 23 with scaling factor $a_3/\eta$ $a_3 = 0.7$ | Eq. 20 |
| $\gamma_{s(i)} \forall i \in [1, K]$ (Weights samples from prior to construct plausible and close to real looking samples.) | 0.98 (ImageNet256), 0.88 (LSUN) | 0.5 |

**Optimization** We fit three sets of variational parameters at every $i$-th critical time in our hierarchy: means, $\mu_{s(i)}$, variances $\tau^2_{s(i)}$ and weights $\gamma_{s(i)}$. Instead of optimizing $\tau$ and $\gamma$ directly, we optimize the real valued $\tilde{\tau} = \log \tau^2$, and $\tilde{\gamma} = \log(\frac{\gamma}{1-\gamma})$. We optimize this set of variational parameters $\lambda = \{\mu, \tilde{\gamma}, \tilde{\tau}\}$ using Adam with an initial learning rate of $\{0.1, 0.1, 0.01\}$ respectively and decreasing the learning rate by a factor of $0.99$ every $10$ iterations. We find this setting to be robust across all prior diffusion models and datasets in our work.

During pre-training, most diffusion models parameterize the mean prediction at every diffusion time step $t$ and fix variances, however some previous work Nichol & Dhariwal (2021); Dhariwal & Nichol (2021) has found that (with appropriate training "tricks") learning variances improves performance. Some previous works like ReSample tunes this as a hyperparameter. We instead learn this in our work, and we adjust learning rates to avoid local optima in this process. We optimize the parameters in VIPaint with $K = 2$ for $50$ iterations; VIPaint with $K = 4$ is optimized for $100$ steps in the case of LSUN Churches, $150$ steps for the ImageNet64 dataset and $250$ steps for the ImageNet256 dataset.

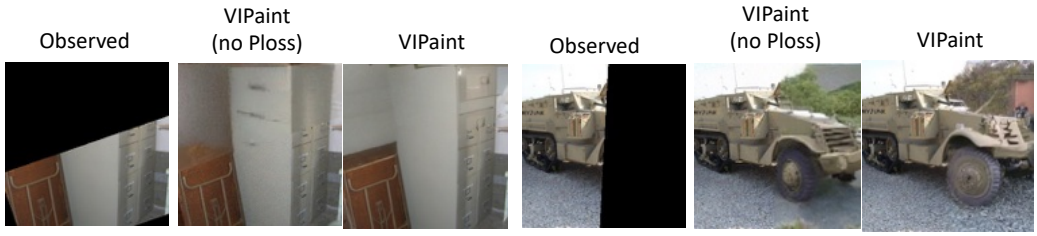


Figure 8: An ablation showing the effect of addition the perceptual loss (PLoss) in the reconstruction term for the task of image inpainting using latent diffusion priors. We see that even though VIPaint can inpaint the image semantically without the Perceptual loss, this loss becomes important to produce sharper reconstructions.


Figure 9: (**Left**) We show the effect of the hyperparameter $\beta$ with VIPaint with respect to optimization iterations. (**Right**) we show the respective loss curves. With $\beta = 10$, VIPaint captures more variations under the diffusion prior instead of "setting" to one kind of completion with $\beta = 1$.

**Sampling** Post training, we take $400$ iterative refinement steps from $T_s = 400$ in the LDM variance preserving schedule to sample inpaintings using a scale factor of 2, similar to the DPS algorithm using perceptual loss. On the other hand, for the EDM prior, we take $700$ refinement steps to produce inpaintings after $T_s = 2$, with scale $5$ (similar to DPS tuned for EDM in our work). This scale hyperparameter is tuned over the values [0.1, 0.5, 1, 2, 5, 10] on a validation set of 20 images. During the sampling phase, we use the classifier-free guidance rule with scale $= 3$ for the ImageNet256 latent diffusion prior.

**Reconstruction loss** We assume $p(y|z_{T_s})$ as a Laplacian distribution, where the mean is given by $y$ and a scale parameter, which is computed over $100$ images per dataset as a standard deviation over all pixel dimensions. For the $256$ pixel datasets, this is $0.56$, and for ImageNet64 it is $0.05$. In addition to this, we add the perceptual loss for LDM priors, computing them via feeding the pre-trained Inception network with masked images and masked reconstructions. See Fig. 8 for the benefits of using the perceptual loss with VIPaint. Additionally, we use $\beta = 1$ for VIPaint with $K = 1$, which is optimized for $50$ iterations for faster convergence. For VIPaint with $K = 4$, we use $\beta = 50$ for pixel-based EDM prior and $\beta = 10$ for LDM prior. We show the effect of the different $\beta$ values in Fig. 9. Generally speaking, higher values of $\beta$ explores the diffusion latent space more and lower values weighs the likelihood term relatively more and converges faster to a solution.

**Descretization of timesteps for prior diffusion loss $L_{(T_e,T)}(z_{T_e})$** Lastly, we directly adapt the descretization technique from EDM Karras et al. (2022) to compute the diffusion loss. We use $\rho = 7$ across all models and datasets as used by Karras et al. (2022).

### E.2 Baseline Details

Across all the baselines applicable to the latent diffusion models for the ImageNet256 dataset, we use the classifier-free guidance with a scale 3 Rombach et al. (2022).

**Blended** We run blended for $1000$ discretization steps using the EDM and LDM prior.

**RePaint** RePaint uses a descritization of 256 steps along with the standard jump length = 10, and number of times to perform this jump operation also set to 10, following standard practice Lugmayr et al. (2022).

**DPS** Similar to blended, we take 1000 denoising steps for DPS and set scale = 5 for the edm-based diffusion model, while take 500 steps and keep scale as 0.5 for the Latent Diffusion prior (similar to the original work in Chung et al. (2023)). When using the perceptual loss for the latent diffusion prior, we increase the scale to 2.

**PSLD** This is an inference technique only for the Latent Diffusion prior. Similar to DPS, we take 500 steps and keep scaling hyperparameters set to 0.2 as opposed to choosing 0.1 in the original paper Rout et al. (2023). We observe artifacts in the innpainted image if we increase the scale further as also observed by Chung et al. (2024).

**CoPaint** We directly adapt the author-provided implementation of CoPaint and CoPaint-TT Zhang et al. (2023) to use the EDM prior. Apart from the diffusion schedule and network architecture (taken from EDM) all other hyperparameters are preserved from the base CoPaint implementation.

**RED-Diff** As with CoPaint, We directly adapt the author-provided implementation of RED-Diff Mardani et al. (2024) and Red-Diff (Var) to use the EDM prior. In this case we increased the prior regularization weight from 0.25 to 50, which we found gave improved performance and more closely matches our VIPaint settings.

**ReSample** As with other baselines, we directly adapt the author-provided implementation of ReSample Song et al. (2024) for the LDM prior. Because the original code takes larger optimization steps, resulting in high sampling time, we decrease the number of optimization steps to 50, such that the wall-clock run-time of this method matches the other baselines.

## F Inference Time.

**Time Complexity.** The time taken by VIPaint-$K$ to optimize a Markov posterior with $K$ keypoints scales primarily with the number of denoising network calls. Each optimization step (assuming $K \ll T$) involves $O(K)$ calls to sample $z_{T_s} \sim q_\lambda(z_{T_s:T_e})$, and one to compute the diffusion prior loss, resulting in $O(K)$ function calls per step. Thus, for $I$ optimization steps, the overall complexity is $O(KI)$. For example, our VIPaint-2 is optimized over 50 iterations, it requires only $50*(2+1) = 150$ denoising network calls to infer global image semantics. Once fit, sampling requires an additional $T_s$ denoising network calls per sample instead of $T$ network calls as in traditional sampling methods.

**Time Complexity** The time taken by VIPaint scales primarily with the number of denoising network calls. Each optimization step for a $K-$step posterior ($K \ll T$) involves $O(K)$ calls to sample $z_{T_s} \sim q_\lambda(z)$, and one to compute the diffusion prior loss, resulting in $O(K)$ function calls per step. Thus, for $I$ optimization steps, the overall complexity is $O(KI)$. For example, when VIPaint-2 is optimized over 50 iterations, it requires only $50 * (2 + 1) = 150$ denoising network calls to infer global image semantics. Progress in fitting VIPaint's posterior is shown in Fig. 3; the number of optimization steps may be reduced to more quickly give approximate posteriors. Sampling from this posterior requires iterative refinement with the denoising diffusion network, for an additional $T_s$ calls per sample.

We report the time taken for each inference method to produce 10 inpaintings for 1 test image. VIPaint with $K = 2$ is comparable to the baseline methods in terms of wall clock time and the number of functional evaluations ($E$) of the denoising network. Red-Diff, Blended and RePaint baseline methods do not take gradient of the noise prediction network, whereas all other methods require gradients. In terms of time and number of function calls, we can see that VIPaint-2 takes comparable time and number of function calls as other baselines, but performs far better (Table 6).

Overall, gradient based methods like DPS take longer with an LDM prior because of the use of a decoder per gradient step. PSLD additionally utilizes the encoder and hence, takes longer than DPS.

Table 5: Table comparing (Time (in mins), $E$) pair per Inference method using EDM Prior (top) and LDM Prior (bottom)

| **Red-Diff** | **Blended** | **DPS** | **RePaint** | **CoPaint** | **CoPaint-TT** | **VIPaint-2** | **VIPaint** |
|---|---|---|---|---|---|---|---|
| (1.13, 1000) | (1.13, 1000) | (2.55, 1000) | (2.8, 4700) | (2.6, 500) | (5.4, 1000) | 3.3 = (1.5, 150) (opt.) (1.8, 700) (sampling) | 11.8 = (10, 900) (opt.) (1.8, 700) (sampling) |

| Dataset | **Blended** | **DPS** | **PSLD** | **VIPaint-2** | **VIPaint** |
|---|---|---|---|---|---|
| ImageNet256 | (4, 1000) | (10, 500) | (12.4, 500) | 10 = (2, 150) (opt.) (8, 400) (sampling) | 18 = (10, 1250) (optimization) (8, 400) (sampling) |
| LSUN | (1.3, 1000) | (5.1, 500) | (7.0, 500) | 6.4 = (2.1, 150) (optimization) (4.3, 400) (sampling) | 10 = (5.53, 750) (opt.) (4.3, 400) (sampling) |

## G Computational Resources

All experiments were conducted on a system with 4 Nvidia A6000 GPUs.

## H Additional Experiments

### H.1 Full table on large mask image inpainting

| Task | VIPaint-4 | VIPaint-2 | CoPaint-TT | CoPaint | RePaint | DPS | Blended | RedDiff | RedDiff-V |
|---|---|---|---|---|---|---|---|---|---|
| Rotated Window | **0.289** | <u>0.300</u> | 0.316 | 0.347 | 0.3213 | 0.3203 | 0.3409 | 0.463 | 0.407 |
| Random Mask | <u>0.231</u> | **0.227** | 0.245 | 0.278 | 0.2575 | 0.2880 | 0.2763 | 0.409 | 0.671 |

| Task | Imagenet-256 | | | | | LSUN-Church | | | |
|---|---|---|---|---|---|---|---|---|---|
| | VIPaint-4 | VIPaint-2 | ReSample | PSLD | DPS | VIPaint-2 | ReSample | PSLD | DPS |
| Rotated Window | **0.358** | <u>0.392</u> | 0.537 | 0.576 | 0.606 | **0.455** | <u>0.510</u> | 0.541 | 0.502 |
| Random Mask | **0.373** | <u>0.409</u> | 0.559 | 0.583 | 0.607 | **0.439** | <u>0.485</u> | 0.523 | 0.490 |
| Small Mask | <u>0.292</u> | **0.197** | 0.381 | 0.534 | 0.564 | **0.299** | <u>0.374</u> | 0.413 | 0.421 |

Table 6: Quantitative results (LPIPS, lower is better) for ImageNet64 for the task of image inpainting using pixel-based EDM prior (*top*) and Imagenet-256 and LSUN-Church using LDM priors (*bottom*). LPIPS is estimated as the mean score of 10 inpaintings with respect to the true image, averaged across the test set. VIPaint has superior performance (highlighted in **bold**) in nearly all cases. We <u>underline</u> the second best method. Fig. 10 in the appendix has further comparisons.

### H.2 Analysis of Imagenet Results

Fig. 10 shows details of the comparison between VIPaint and CoPaint with time-travel over 100 randomly selected images from the Imagenet-64 task.

### H.3 Linear Inverse Problems

For linear inverse problems other than inpainting, we consider the following tasks: (1) Gaussian deblurring and (2) super resolution. For Gaussian deblurring, we use a kernel with size $61 \times 61$ with standard deviation 3.0. For super resolution, we use bicubic downsampling, similar setup as Chung et al. (2023). Even though the focus of VIPaint is to remedy inconsistencies in image completion tasks, it can also be extended to linear inverse problems like Super Resolution and Gaussian Deblurring.

We compare the performance of VIPaint with ReSample, PSLD & DPS for ImageNet256 dataset using the LDM prior and for the pixel-based model, we include results for Gaussian Deblurrring comparing VIPaint with DPS. Since the Peak-Signal-to-Noise-Ratio (PSNR) is well defined for such tasks, we report it along with LPIPS in Table 7. Some qualitative plots are in Fig. 11 and 12. We see that VIPaint shows strong advantages over ReSample, DPS and Red-Diff for complex image datasets like ImageNet.

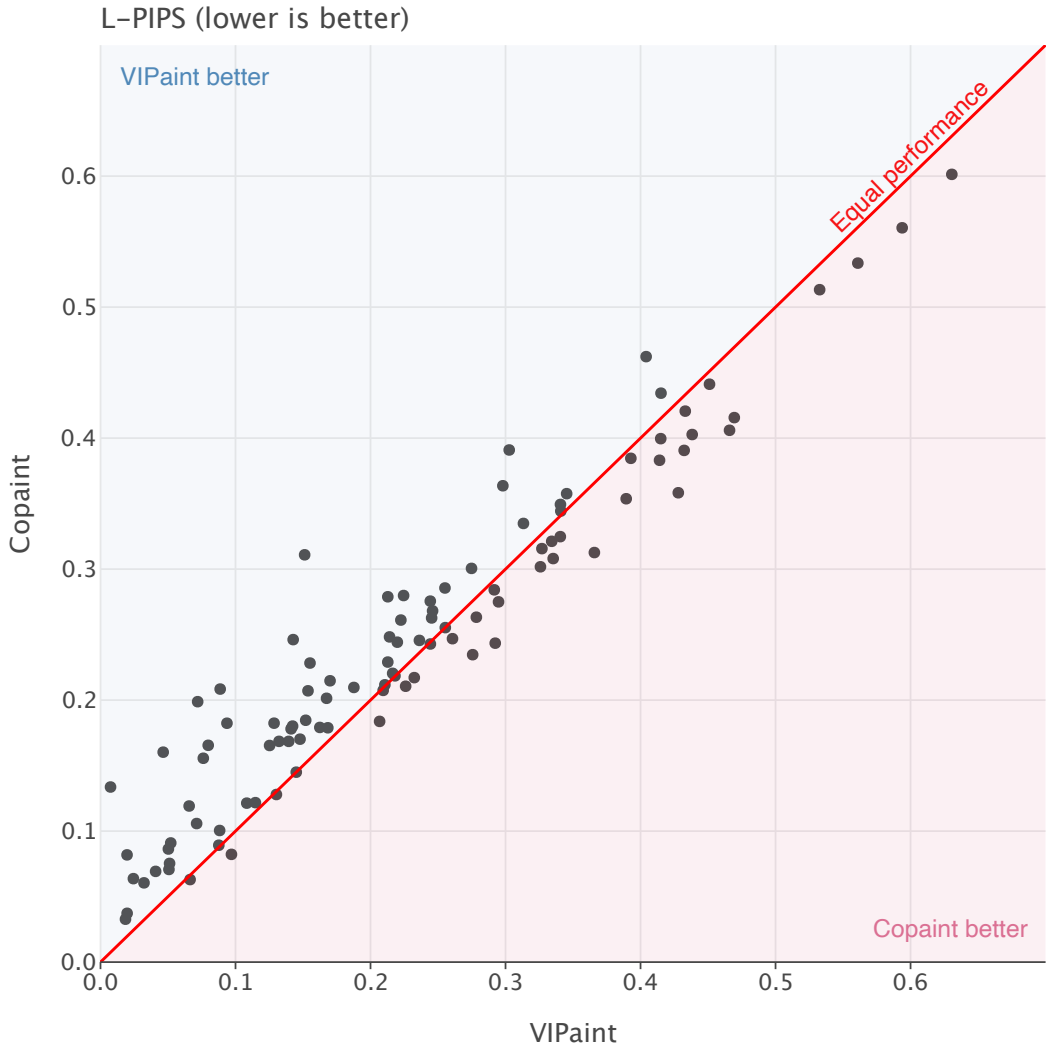


Figure 10: Paired comparison of LPIPS scores for VIPaint-2 and CoPaint with time-travel (CoPaint-TT) on the Imagenet64 “Random Mask" inpainting task (expanding on the experiment shown in table 1. Each point shows the mean LPIPS score across 10 sampled completions of the masked image, with the x and y coordinates showing the VIPaint and CoPaint-TT scores respectively. Additionally, we validated that VIPaint improves on CoPaint-TT using a one-sided paired t-test on the mean LPIPS scores of each method. We found that the improvement was statistically significant with a p-value of **4.133e-05**. As the normality assumption of the t-test may not hold, we also verfied the results using a nonparametric Wilcoxon signed ranked test, which indicated a statistically significant improvement with a p-value of **0.000114**

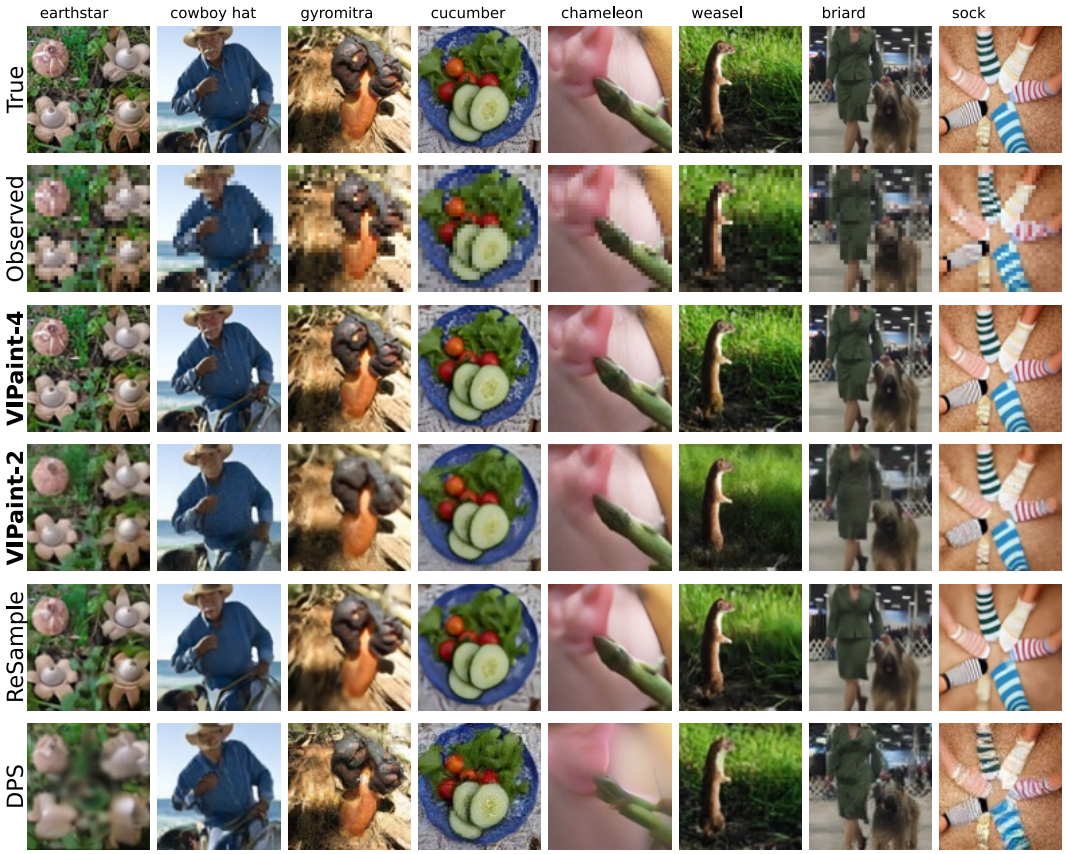


Figure 11: Qualitative results on Imagenet256 for Super Resolution. We see that DPS produces completely blurry images. We see improvements with ReSample. In contrast, VIPaint-4 leads to samples closer to the true image and produces *very* realistic images.

| Task | ImageNet256 Super-resolution 4x | | ImageNet256 Gaussian Deblur | | ImageNet64 Gaussian Deblur | |
|---|---|---|---|---|---|---|
| Metric | LPIPS ↓ | PSNR ↑ | LPIPS ↓ | PSNR ↑ | LPIPS ↓ | PSNR ↑ |
| VIPaint-4 | **0.33** | **19.31** | <u>0.44</u> | <u>17.90</u> | **0.306** | **13.47** |
| VIPaint-2 | 0.46 | 16.36 | 0.48 | 16.35 | **0.305** | **13.60** |
| ReSample | <u>0.395</u> | <u>18.410</u> | **0.435** | **18.03** | – | – |
| PSLD | 0.67 | 7.77 | 0.583 | 0.022 | – | – |
| DPS | 0.579 | 12.99 | 0.595 | 12.608 | 0.319 | **13.43** |

Table 7: Quantitative results (LPIPS, PSNR) for solving linear inverse problems on ImageNet256 using LDM priors and ImageNet64 using EDM priors. Best results are in bold and second best results are underlined. For nonlinear deblurring.

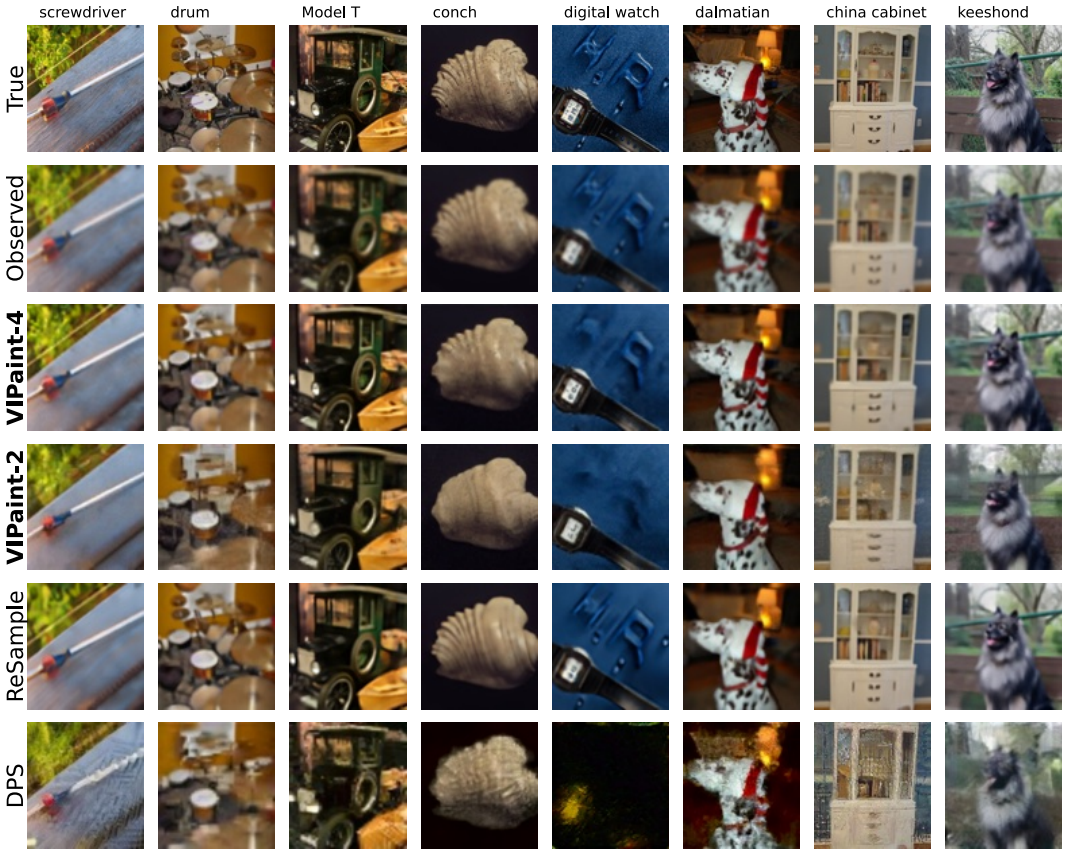


Figure 12: Qualitative results on Imagenet256 Gaussian DeBlurring using LDM prior.

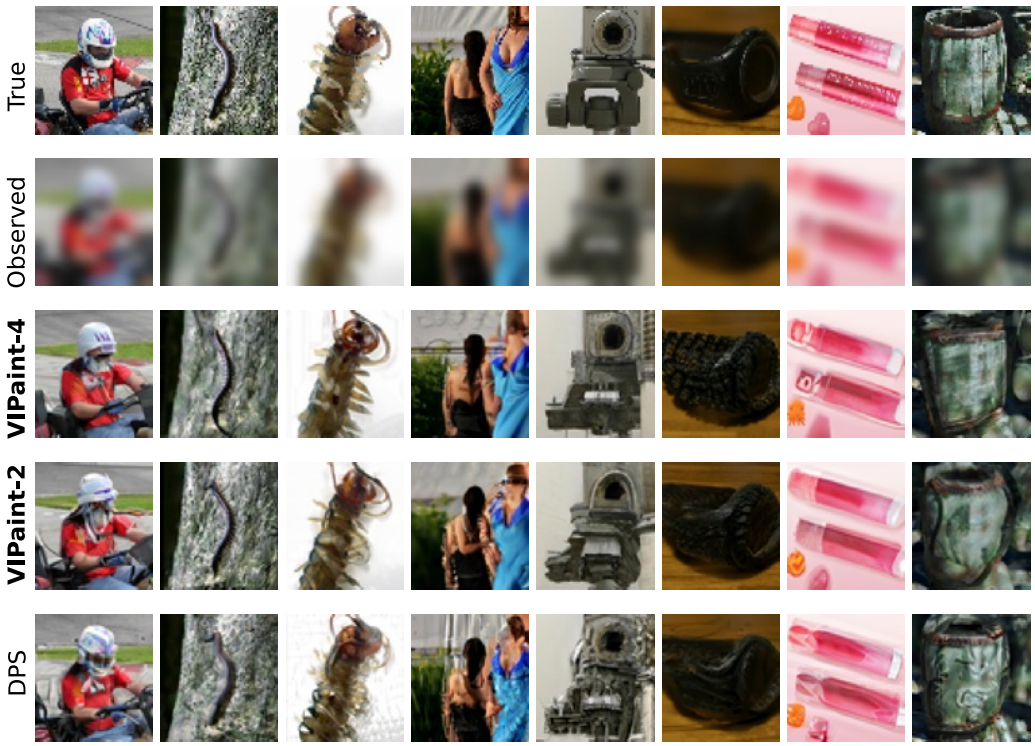


Figure 13: Qualitative results for Gaussian DeBlurring using EDM prior for ImageNet64. We see VIPaint leads to samples closer to the true image and producess *more* realistic images.

### H.4 Small-Mask Image Inpainting for LSUN, ImageNet256

We show some qualitative figures for small masking ratios (upto 20% of the image is corrupted) in Fig. 14 for ImageNet-256 dataset.

### H.5 VIPaint captures multi-modal posterior

In addition to producing valid inpaintings, we show multiple samples per test image for all datasets we consider in Fig. 15 -16.

### H.6 More qualitative results


Figure 14: Qualitative results on the performance across methods for small masking ratios for ImageNet256 dataset using LDM prior. All methods seem to perform reasonably well in this regime.

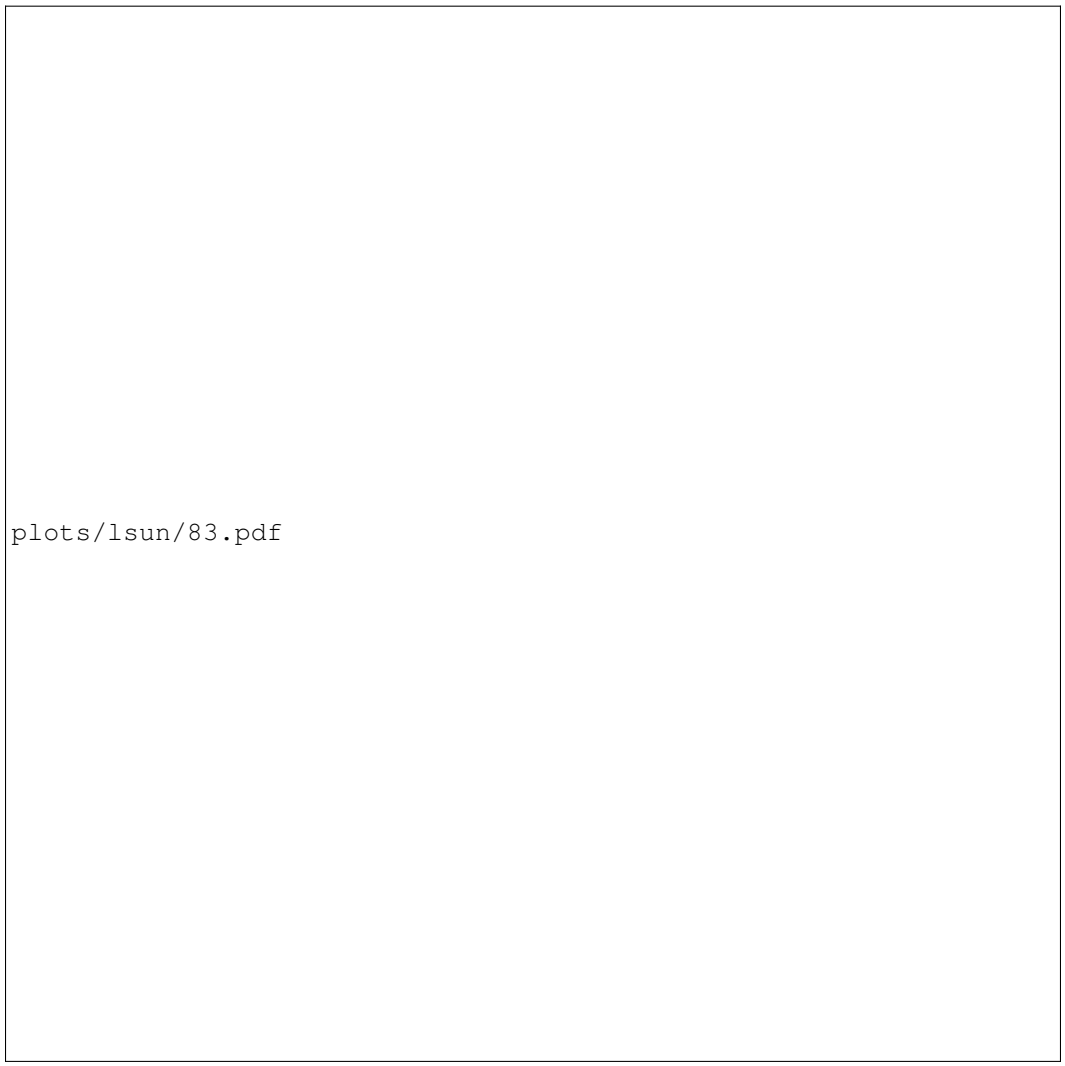


Figure 15: LSUN diversity results. Examples of diverse generation using VIPaint and baseline methods on LSUN using the same input and different initial noise.

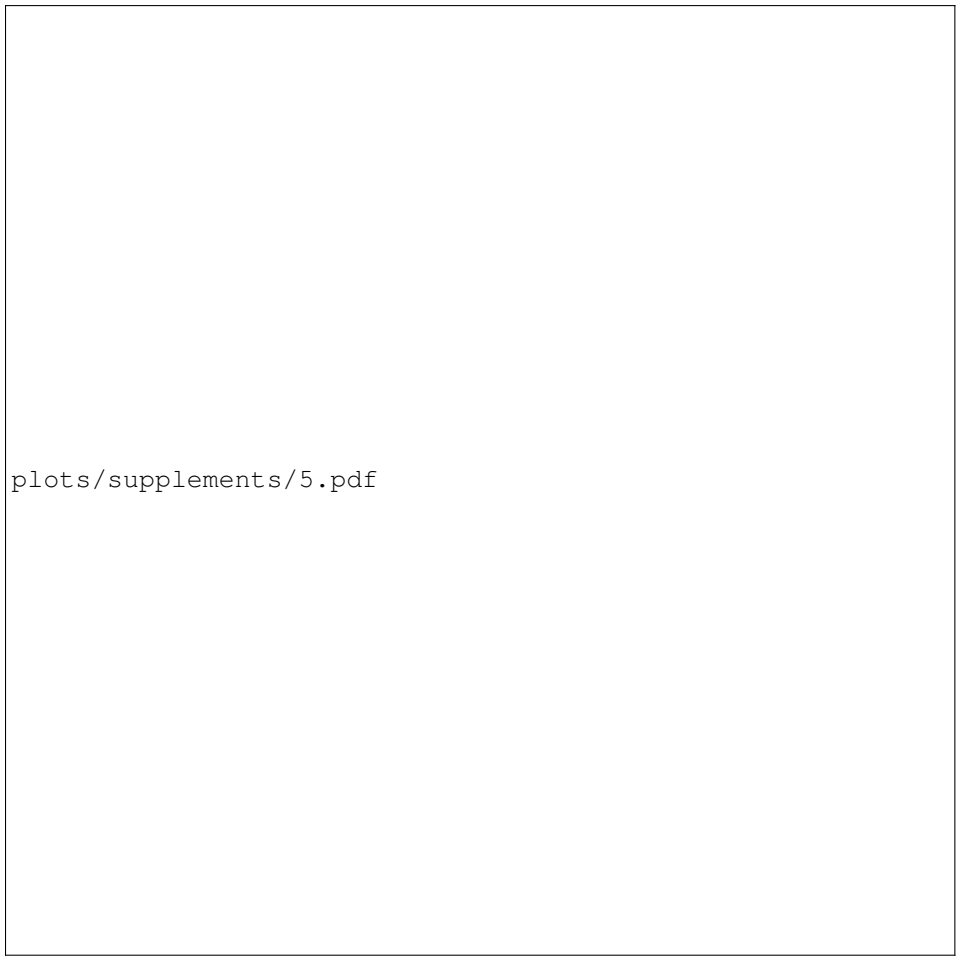


Figure 16: ImageNet64 diversity results with the same class condition but different initial noise.

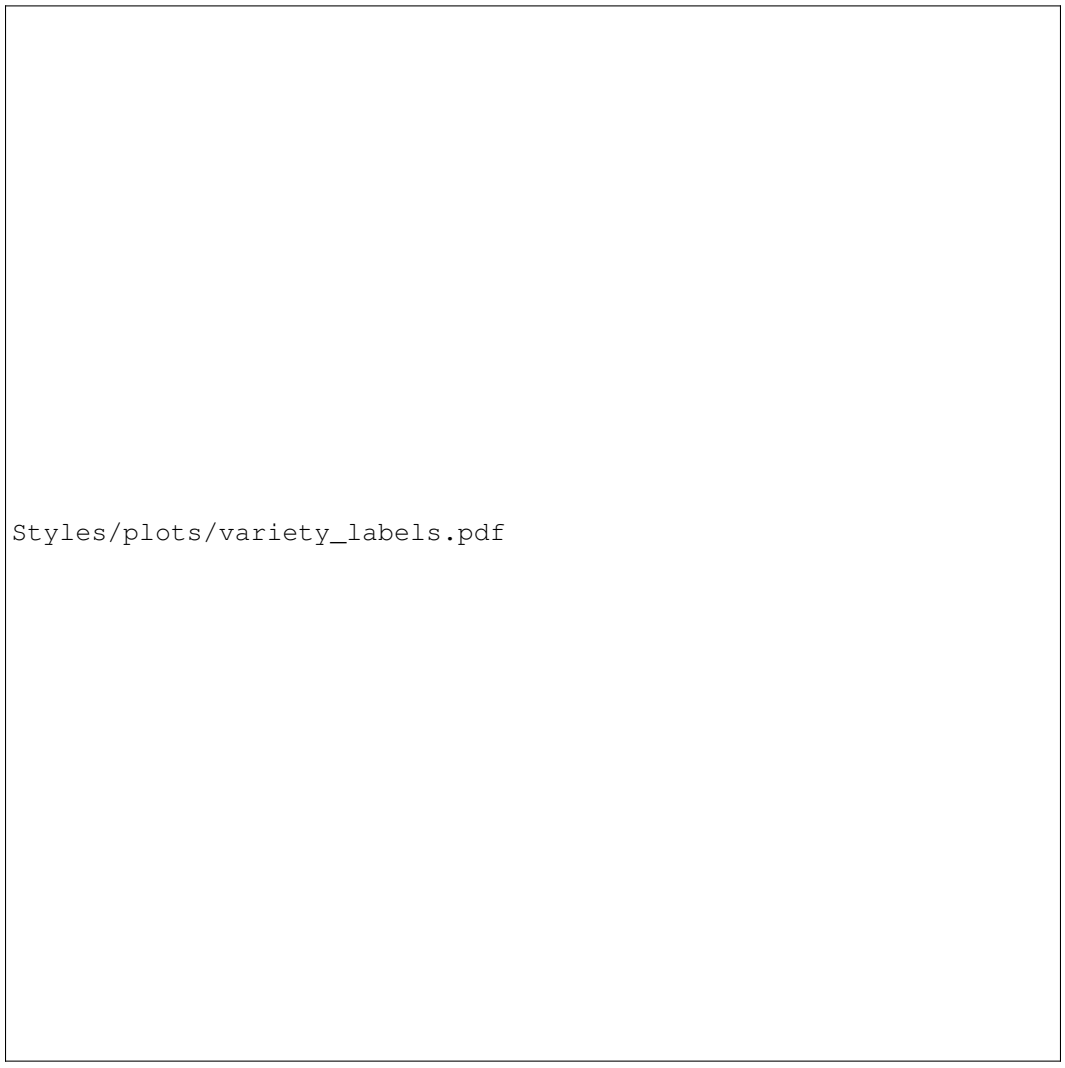


Figure 17: Qualitative results for VIPaint diversity for ImageNet256 with LDM prior using different class conditioning. We see that VIPaint follows the input label and ensures consistency with the observed set of pixels.


Figure 18: Image completion results on Imagenet256 using the LDM prior for Rotated Window and Random Masking schemes shown in the second row. We show an inpainting from each method in the following four rows. DPS, PSLD, and ReSample show blurry inpaintings of widely varying quality. In contrast, VIPaint interprets the global semantics in the observed image and produces *very* realistic images. Please find more qualitative plots for LSUN-church in the Appendix Fig. 19.

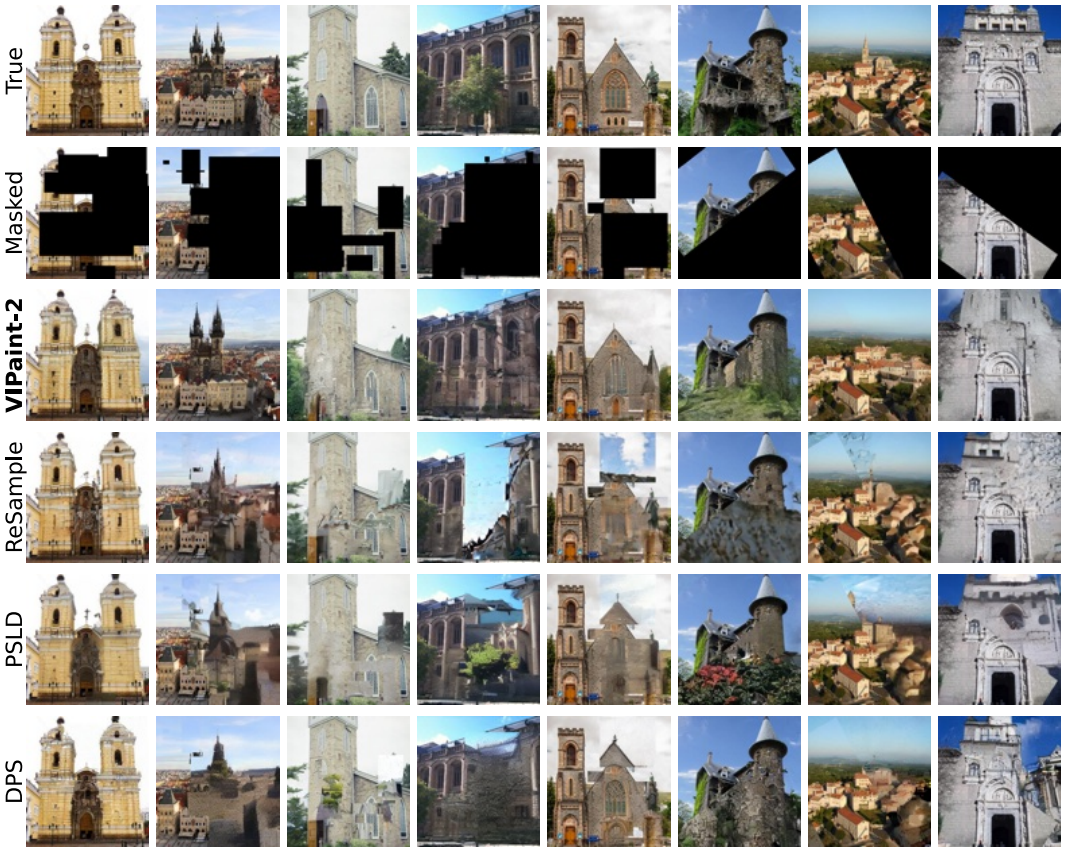


Figure 19: Qualitative results for LSUN-church dataset using LDM prior for the tasks of image inpainting with large masks. We see that VIPaint-2 can inpaint the images consistently and without any artifacts at the mask borders.

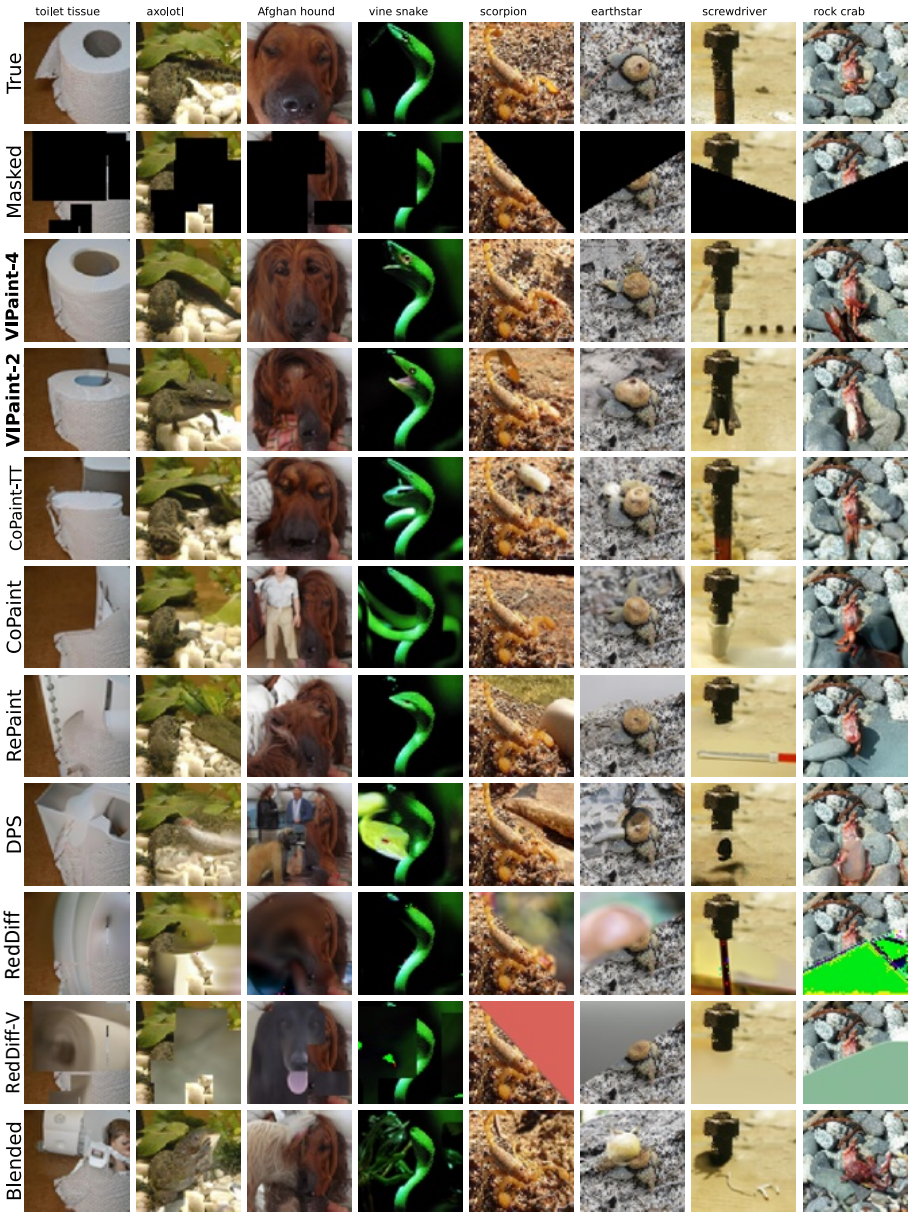


Figure 20: Image completion results on ImageNet64 using a conditional pixel-based EDM prior for image inpainting (Random Masking and Rotated Window schemes) shown in the second row. We show an inpainting from each method in the following rows. Even though the prior diffusion model for ImageNet is conditioned on class labels, inpaintings for baseline methods are inconsistent with the observed image. RePaint and CoPaint is typically more accurate than other baselines, but still produce inconsistent samples unless masks are small. In contrast, VIPaint interprets the global semantics in the observed image while enforcing consistency with the few observed pixels.