Going beyond Compositions, DDPMs Can Produce Zero-Shot Interpolations

Justin Deschenaux [*1] **Igor Krawczuk** [*2] **Grigorios G. Chrysos** [3 2] **Volkan Cevher** [2]

Abstract

Denoising Diffusion Probabilistic Models (DDPMs) exhibit remarkable capabilities in image generation, with studies suggesting that they can generalize by composing latent factors learned from the training data. In this work, we go further and study DDPMs trained on strictly separate subsets of the data distribution with large gaps on the support of the latent factors. We show that such a model can effectively generate images in the unexplored, intermediate regions of the distribution. For instance, when trained on clearly smiling and non-smiling faces, we demonstrate a sampling procedure which can generate slightly smiling faces without reference images (zero-shot interpolation). We replicate these findings for other attributes as well as other datasets. Our code is available on GitHub.

1. Introduction

In recent years Diffusion models (DM) (Sohl-Dickstein et al., 2015; Ho et al., 2020; Song et al., 2021) have emerged as highly successful generative methods for content synthesis (Nichol & Dhariwal, 2021; Rombach et al., 2022; Du et al., 2020). As observed by Okawa et al. (2023) and others, DMs are capable of leveraging *compositionality*, an important concept in psychology (Frege & Gabriel, 1980), philosophy (Pelletier, 1994) and linguistics (Williams, 1983) which refers to the human capability to combine attributes into novel configurations. For instance, after observing smiling people with both long hair and frowning people with short hair, humans can imagine smiling short-haired individuals. Furthermore, humans are able to *interpolate* attributes as well, e.g. they will be able to imagine short-haired individuals with mild smiles or neutral expressions as well. Crucially, interpolating implicitly assumes there exists a latent factor governing a set of attributes, inducing a natural ranking of samples according to the factor of interest. In the sequel, we will refer to attributes as the *expressions* corresponding to a latent variable which induces a ranking. **Remarkably, we observe that diffusion models are capable of interpolation without being trained on instances with intermediate expressions of the latent factor**. In the example, a generative model capable of interpolation would not be exposed to neutral individuals or mild smiles and frowns, yet could generate samples exhibiting these facial expressions.

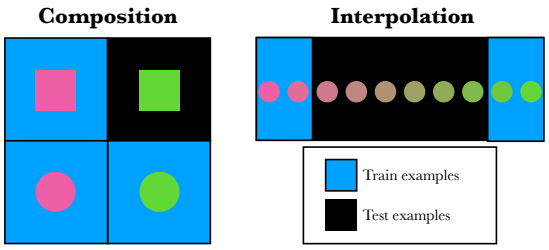


Figure 1: Assume that the characteristic of real samples is influenced by some latent variables z_i (e.g. color and shape). If the support of the latent space is a cartesian product of the individual latent variables (e.g. pairs (color, shape) typically finite and discrete), we say that the samples depend on **combinations of the individual latent variables**. Conversely, if the latent variables z_i are defined on a closed interval (e.g. $z_i \in [0, 1]$), such that any value of z_i induces a meaningful sample (e.g. any color in between green and pink), then we say that there exists an **interpolation between extreme samples**, where the extreme examples are samples whose latent are close to the extrema of their support (e.g. pure pink or pure green colors).

Interpolation abilities are crucial to address questions related to fairness and bias mitigation in machine learning through generative models (Sattigeri et al., 2019; Rajabi & Garibay, 2021; Tan et al., 2021). Indeed, a model able to interpolate would allow generating nuanced samples even when the training dataset contains little diversity. This is especially important because the data collection process might filter out instances that do not meet ad-hoc criteria. Such procedure

[*]Equal contribution [1]Department of Computer Science, École Polytechnique Fédérale de Lausanne (EPFL), Lausanne, Switzerland [2]LIONS, École Polytechnique Fédérale de Lausanne (EPFL), Lausanne, Switzerland [3]Department of Electrical and Computer Engineering, University of Wisconsin-Madison, USA. Correspondence to: Justin Deschenaux <justin.deschenaux@gmail.com>, Igor Krawczuk <igor@krawczuk.eu>.

Proceedings of the 41st International Conference on Machine Learning, Vienna, Austria. PMLR 235, 2024. Copyright 2024 by the author(s).

would result in unexpectedly non-representative training sets for specific underrepresented attributes. In addition, a model with strong interpolation would prove invaluable in scenarios where specific expressions of the latent factor of interest can only be sparsely observed. For instance, in medical imaging, where DMs are being explored as priors for MRI reconstruction(Güngör et al., 2023), certain rare pathologies may not be sufficiently represented. Thus the capacity of the model to seamlessly interpolate is crucial for generating (and accurately reconstructing) diverse data.

While the ability to compose fixed attributes was identified in diffusion models in controlled[1] settings (Okawa et al., 2023), this work presents, for the first time, a clear demonstration that Denoising Diffusion Probabilistic Models (DDPM) can reliably interpolate between training classes, going far beyond the support of the training data, as we illustrate in Figure 1.

In particular, we demonstrate zero-shot interpolation of DDPMs, wherein all models are exclusively trained on data clearly excluding intermediate values of the latent factor. We now present our main contributions. For reasons of space, we defer certain results to the appendix:

- We demonstrate the interpolation abilities of DDPMs trained on examples with one factor of interest in Section 5. We demonstrate this ability on real-world datasets, filtered to retain examples with clear attributes only, as depicted in Figure 1 (right). Importantly, the training samples are highly separated in our experiments.
- We demonstrate that interpolation can happen in low data regimes in Section 5.2, is robust to hyperparameter choices in Section 5.3, model architecture, and occurs with various sampling strategies (Appendix J). We further explore regularization to improve interpolation capabilities in Section 5.5
- We demonstrate that DMs trained on clear attributes only can perform attribute editing, and generate images beyond the training distribution (Appendix N).
- In Appendix M, we demonstrate that decoding interpolated latents generates samples with intermediate expression of the latent factors, despite being not trained on such examples.
- Finally, we present interpolation on two attributes in Section 6.

[1]Production text-to-image models deployed in applications use vast amounts of web-scraped training data. Hence, Occam's razor might suggest that novelties emerge as compositions of latent factors from the training data.

Figure 2: Images from CelebA dataset. **Left**: clearly non-smiling face. **Two center**: mild smiles. **Right**: clearly smiling.

2. Background

Consider statistical models that approximate a distribution $p(X, y)$, where $X \in \mathbb{R}^D$ represents high-dimensional signals such as images, and $y \in \mathbb{R}^L$ encodes conceptual information about the image. In recent years, diffusion models have garnered significant attention in the research community for their effectiveness in modeling and generating approximate samples from $p(X, y)$.

2.1. Diffusion and Energy-Based Models

Introduced in Sohl-Dickstein et al. (2015), diffusion models (DMs) serve as the foundation of many state-of-the-art text-to-image models (Ramesh et al., 2022; Rombach et al., 2022; Saharia et al., 2022). Ho et al. (2020) introduced Denoising Diffusion Probabilistic Models (DDPMs), and demonstrated that diffusion models could match the FID of GANs for certain datasets.

DDPMs materialize the diffusion process as a hierarchy of T increasingly noisy samples, mapping the original distribution to white noise, where sampling is achieved through variants of the stochastic gradient Langevin dynamics algorithm (Welling & Teh, 2011). The noise addition process is governed by a schedule $(\alpha_t)_{t=1}^T$, such that $p(x_t|x_{t-1}) \sim \mathcal{N}\left(x_t; \sqrt{\alpha_t}x_{t-1}, (1-\alpha_t)\mathbf{I}\right)$. Because of the Gaussian transition, one can compute $x_t|x_0$ in a single step, as $x_t = \sqrt{\bar{\alpha}_t}x_0 + \sqrt{1-\bar{\alpha}_t}\epsilon$, where $\bar{\alpha}_t = \prod_{i=1}^t \alpha_i$, and $\epsilon \sim \mathcal{N}(0, \mathbf{I})$. Ho et al. (2020) sample from the data distribution p_{data} by approximating the denoising distribution $p(x_{t-1}|x_t)$ with a Gaussian $p_\theta(x_{t-1}|x_t) = \mathcal{N}\left(x_{t-1}|\mu_\theta(x_t, t); \sigma_t\mathbf{I}\right)$, where σ_t is computed in terms of $(\alpha_t)_{t=1}^T$ and $\mu_\theta(x_t, t) = \frac{1}{\sqrt{\alpha_t}}\left(x_t - \frac{1-\alpha_t}{\sqrt{1-\bar{\alpha}_t}}\epsilon_\theta(x_t, t)\right)$, where $\epsilon_\theta(x_t, t)$ estimates the added noise. Ho et al. (2020) train ϵ_θ to minimize $L_{\text{simple}} := \mathbb{E}_{t,x_0,\epsilon}\left[\|\epsilon - \epsilon_\theta(x_t, t)\|^2\right]$, where t is picked uniformly at random in $1, ..., T$, $\epsilon \sim \mathcal{N}(0, \mathbf{I})$, and x_0 is a data sample. See Appendix A for more details.

Energy-based models (EBMs) (Ngiam et al., 2011; Xie et al., 2016; Du & Mordatch, 2019; 2020; Du et al., 2021b) parameterize a distribution $p_\theta(x)$ through the non-normalized Energy function $E(\cdot)$, as $p_\theta(x) \propto e^{-E(x)}$. Notably, the energy of a product distribution is the sum of energies of each

component. EBMs and diffusion models share functionally equivalent learning objective and sampling method (Liu et al., 2023a). Interestingly, Liu et al. (2023a) osbserved that summing the score from multiple pre-trained diffusion models improves composition ability on CLEVR (Johnson et al., 2016).

2.2. Conditional Generation with Classifier Guidance

In the main body of this work, we study conditional generation using classifier guidance (Song et al., 2021; Dhariwal & Nichol, 2021) as it lets us reuse unconditional models during our explorations. Since classifier-free guidance (CFG) (Ho & Salimans, 2022) typically achieves superior FID (Heusel et al., 2017) and IS (Salimans et al., 2016), we study CFG and present results in Appendix J. We observe interpolation capabilities with both sampling schemes. Practically, DDPMs implement classifier guidance by modifying the mean of the denoising distribution $p_\theta(x_{t-1}|x_t)$ from Section 2.1. Formally, the new mean is computed as $\tilde{\mu}_\theta = \mu_\theta(x_t, t) + \sigma_t^2 g$, where $g = \lambda \nabla_x \log p_\theta(y|x)|_{x=\mu_\theta(x_t,t)}$, $\lambda \in \mathbb{R}$ (guidance strength) is such that $|\lambda| \geq 1$ and $p_\theta(y|x)$ is learned with a classifier.

2.3. Compositional Generalization

We build upon the concept of compositional generalization of Wiedemer et al. (2023) to define interpolation in Section 4.1. Compositional generalization denotes the ability to compose attributes in novel ways. Importantly Wiedemer et al. (2023) introduce the concept of *compositional support*:

Definition 2.1 (Wiedemer et al. (2023))**.** Let P and Q be distributions defined on a latent domain $\mathcal{Z}_1 \times ... \times \mathcal{Z}_K$. We say that P and Q have **compositional support** when P and Q have matching marginals' support. Formally, denoting as P_k the marginalization of P over $\mathcal{Z}_i \neq \mathcal{Z}_k$, $\forall i \neq k$, compositional support between P and Q is achieved when

$$\text{supp } P_k = \text{supp } Q_K, \quad \forall k = 1, ..., K, \tag{1}$$

where supp denotes the support of a distribution.

Wiedemer et al. (2023) argue that compositional generalization (sampling from Q when trained on P) is only possible if the train and test distribution have compositional support. Okawa et al. (2023) suggest that diffusion models have compositional generalization abilities. *Interpolation, formally defined in Section 4.1, is distinct from composition*, since the train and test distributions do not have compositional support. Figure 1 shows visually why interpolation is conceptually independent and separate from composition.

3. Related Work

Compositionality The concept of compositionality broadly implies that a whole can be explained by the sum of its parts, in addition to how components relate to each other. Compositionality is often attributed to Frege, as developed by Pelletier (1994). Prior work in generative modeling have explored compositional aspects of the learned distribution for GANs (Chen et al., 2016; Shoshan et al., 2021), VAEs (Higgins et al., 2017; Georgopoulos et al., 2020), diffusion models (Liu et al., 2023a; Du et al., 2023b), and EBMs (Du et al., 2020; Liu et al., 2021; Du et al., 2021a). Wiedemer et al. (2023) formally defined compositional generalization and proposed sufficient conditions, but specifically excluded the study of interpolation and extrapolation. Additionally, compositional generalization has been linked to causality (Pawlowski et al., 2020; Besserve et al., 2021). Finally, Okawa et al. (2023) study diffusion models when the train and test distributions have compositional support, but do not explore interpolation, the focus of our work (Section 4.1).

Text-to-Image Generation In recent years, diffusion models (Sohl-Dickstein et al., 2015; Song & Ermon, 2019; Ho et al., 2020) showcase impressive abilities to sample high quality compositions based on text prompts (Ramesh et al., 2022; Rombach et al., 2022; Saharia et al., 2022). Other model classes appear able of composition in this setting as well, eg.. Parti (Yu et al., 2022), an autoregressive model, beats Dall·E 2 (Ramesh et al., 2022) and Imagen (Saharia et al., 2022) on FID on MS-COCO (Lin et al., 2015). On the same dataset, GigaGAN (Kang et al., 2023) is also competitive with Stable Diffusion (Rombach et al., 2022). To be able to cleanly separate interpolation from compositions, we focus in pure vision models and leave the exploration of interpolation in text conditioned and multi-modal models to future work.

Advantages of Diffusion Models Diffusion models enjoy stable training, relative ease of scaling (Dhariwal & Nichol, 2021), and offer likelihood estimates of samples (Song et al., 2021). Kadkhodaie et al. (2023) show that models trained on disjoint sets of samples can learn the same high-dimensional density, suggesting strong inductive biases. Closest to our work, Zhu et al. (2023) found that DDPMs can perform few-shot non-compositional extrapolation. Our work differs in several key aspects. First, we focus on zero-shot interpolation, whereas Zhu et al. (2023) focus on few-shot extrapolation. Additionally, we do not introduce new learnable parameters, while their approach requires fitting high-dimensional Gaussians.

Diffusion Models and Out-of-Distribution Detection In *discriminative* contexts, Graham et al. (2023a;b); Liu et al. (2023b) employ diffusion models to identify out-of-

distribution (OOD) examples. Du et al. (2023a) employ diffusion models to generate a dataset of outliers. This dataset is used to train a classifier, as training on borderline outliers enhance OOD detection performance.

Connections between Diffusion Models and Denoising Score-Matching Concurrently to Ho et al. (2020); Nichol & Dhariwal (2021), Song & Ermon (2019; 2020); Song et al. (2021) developed analogous formulations related to score matching (Hyvärinen, 2005; Vincent, 2011). As noted in Du et al. (2023b), the DDPMs training objective is proportional to the denoising score-matching objective of Song & Ermon (2019).

Interpolation in Diffusion Models Previous research has explored using diffusion models for interpolation and attribute editing. While impressive, those works do not constrain the training distribution to measure interpolation capabilities. Specifically, Yue et al. (2024) present results on attribute editing, leveraging a disentangled representation learned from the entire CelebA training data. Kim et al. (2022) conduct attribute editing experiments using pre-trained CLIP models, while Wang & Golland (2023) demonstrate latent space interpolation between pairs of images using pre-trained latent diffusion (Rombach et al., 2022) and CLIP models.

4. Methodology

We demonstrate interpolation capabilities of diffusion models beyond the training distribution. In Section 4.1, we outline the data generation model, present the problem statement and define interpolation formally. We use a guidance mechanism inspired by EBMs, presented in Section 4.2. In Section 4.3, we detail how we extract a subset of examples with clear attributes, required to study interpolation.

4.1. Problem Statement

Data Generation Model We denote the set of examples as $\mathcal{D} := \big\{(x_i, z_i) : x_i \in \mathbb{R}^D; z_i \in [0,1]^L; 1 \leq i \leq N\big\}$, where N denotes the dataset size, x_i represent high-dimensional data such as images, and z_i is a vector of latent variables. Let $z_i(\ell)$ denote the ℓ-th entry of the vector z_i, for $1 \leq \ell \leq L$. Each $z_i(\ell)$ denotes the strength of a certain attribute in sample x_i. For example, if x_i is a picture of a face, $z_i(1)$ could model the strength of the smile, where $z_i(1) \approx 0.1$ is supposed to be non-smiling while $z_k(1) \approx 0.9$ means that x_i represents a clearly smiling face. As explained in the paragraph "Practical considerations", when the dataset does not contain continuous labels, we compute attributes using a classifier. Additionally, if the labels are continuous and take values in $[a, b]$, one can obtain values in $[0, 1]$ via a simple rescaling. We did not observe any drawback to assuming that $z \in [0, 1]$.

Semi-Order on Attributes We assume there exists a ranking between samples based on the latent factor $z_i(\ell)$. However, it is often challenging to discern between examples with latent variables having similar values. For instance, when $z_i(\ell)$ represents the smile attribute, distinguishing between $z_i(\ell) = 0.45$ and $z_j(\ell) = 0.5$ visually is challenging, unlike cases with more noticeable differences, such as $z_i(\ell) = 0.05$ and $z_j(\ell) = 0.45$. This notion of minimal distance for comparison is captured by **semi-orders** (Luce, 1956; Pirlot & Vincke, 1997). The semi-order $\precsim_\epsilon$ compares samples based on the attribute $z_i(\ell)$, assuming there is a distance $\epsilon > 0$ between latent variables $z_i(\ell)$. Formally, for $(x_0, z_0), (x_1, z_1) \in \mathcal{D}$, we say that $z_0(\ell) \precsim_\epsilon z_1(\ell)$ if $z_0(\ell) < z_1(\ell)$ *and* $|z_0(\ell) - z_1(\ell)| > \epsilon$. Conversely, when $|z_0(\ell) - z_1(\ell)| \leq \epsilon$, no meaningful ordering can be established between samples based on attribute ℓ.

Mild and Extreme Attributes Let $z(\ell) \in [0, 1]$ be an attribute of interest and $\precsim_\epsilon$ a semi-order on $[0, 1]$. We define **extreme examples**, as samples with latent variables $z(j) \in [0, \delta] \cup [1 - \delta, 1]$. For a small δ, extreme examples are samples that undoubtedly do or do not have a certain attribute. Conversely, we say that an example has an attribute **mildly present**, when its associated latent is such that $z(\ell) \in [0.5 - \delta, 0.5 + \delta]$. Importantly, we assume that it is visually possible to tell extreme and mild attributes apart from each other. Formally, given the semi-order precision ϵ, we need that $0.5 - 2\delta < \epsilon$ so that mild and extreme examples supports are disjoint.

Interpolation without Mild Attributes We say that a generative model is able to interpolate, if it is trained on extreme examples only, with a small δ such as 0.1 or 0.2, yet the model can generate examples with mild attributes. Note that samples can be extreme according to one or multiple variables. See Figure 1 (right sub-figure) for a visualization. Demonstrating interpolation requires a clear gap between the training support ($[0, \delta] \cup [1 - \delta, 1]$) and region of mild samples ($[0.5-\delta, 0.5+\delta]$). This requirement stems from the need to distinguish from the findings of Zhao et al. (2018), who illustrate that many generative models naturally produce examples slightly different from the training data. In their experiments featuring a toy dataset with colored dots, Zhao et al. (2018) showcase the ability of the model to generate images with a few more or fewer dots than those present in the training set.

Practical Considerations Popular datasets, such as CelebA (Liu et al., 2015) use binary labels instead of labels with a continuous range. This is not surprising considering the difficulty in ranking all examples with respect to each other. As such, instead of observing latents $z_i(j) \in [0, 1]$,

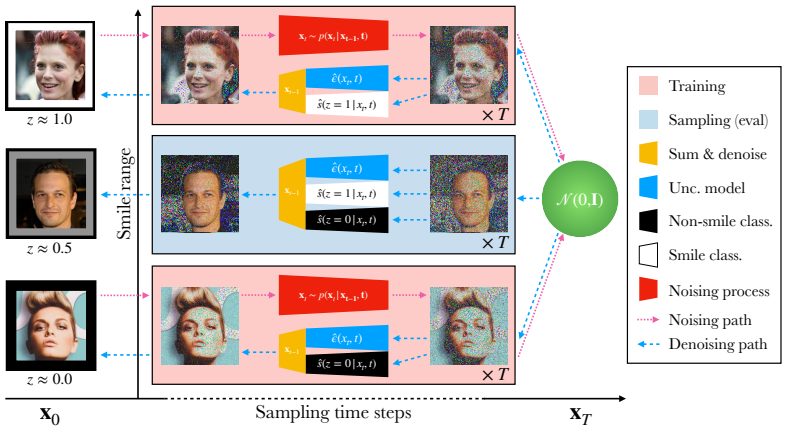


Figure 3: Diagram of the training and sampling process for mild smiles. The classifier and DDPM are trained on *extreme examples only*, i.e. the DDPM is trained on clearly smiling and clearly non-smiling faces. Nonetheless, we demonstrate that DDPMs can generate faces with mild attributes (middle) with a modified sampling scheme, despite *never* encountering those at training. The key for sampling mild attribute is to use the score of the classifier for *both* classes instead of one as in regular classifier-guided sampling. Importantly, we do *not* modify the DDPM training procedure.

we can only access binary variables $\tilde{z}_i(j) \in \{0, 1\}$, while the underlying latents are continuous. Despite binary labels, CelebA contains some diversity in attributes, i.e. it is not composed of extreme examples only. Hence, to demonstrate interpolation, we devise a filtering procedure to detect extreme examples in Section 4.3.

4.2. Multi-Guidance: Sampling using Multiple Scores

In this work we train diffusion models according to the recipe of Nichol & Dhariwal (2021). That is, we train an unconditional diffusion model, and a guidance classifier on noisy samples. At sampling however, we use a different method, inspired by the connections between EBMs and diffusion models (Liu et al., 2023a). We define the product distribution $p_\Pi(x)$, as

$$p_\Pi(x) := \frac{1}{Z} p(x) \prod_{k=1}^{K} \prod_{m=1}^{M} p_k(y_m|x)^{\lambda_{km}}, \qquad (2)$$

where λ_{km} are classifier-guidance parameters, Z is a normalization constant, $p(x)$ is the unconditional data distribution from an unconditional diffusion model, $p_k(\cdot|x)$ are distributions from guidance classifiers and y_m are the (discrete) values that attribute k can take. As described by Liu et al. (2021), sampling from a product distribution can be approximated using the score $\nabla_x \log p_\Pi(x)$, a sum of individual scores from classifiers and unconditional model. Since we train DDPMs, we use $\sum_{k=1}^{K} \sum_{m=1}^{M} \lambda_{km} \nabla_x \log p_k(y_m|x)$ as the guidance score, which generalizes classifier guidance (Section 2.2) to multiple classifiers. We refer to sampling with multiple scores as **multi-guidance**.

4.3. Preparing an Extremal Dataset

Real-world datasets like CelebA, despite their discrete labels, contain some diversity in attributes. For instance, CelebA includes pictures of clearly smiling, non-smiling, and subtly smiling instances (Figure 2). To demonstrate interpolation, we first extract extreme examples, in line with standard practice of dataset filtering of Lakshminarayanan et al. (2017); Fang et al. (2023). More details are given in Appendix D.1. Our procedure relies on two steps: **Firstly**, we detect the presence of an attribute using either a pre-trained model or train a model ourselves. Specifically, for the "Young" attribute, we utilize the FaRL model (Zheng et al., 2022), while training EfficientNet classifiers for other attributes. Using the attribute-dependent model, we extract a set S from CelebA training data, where the model is confident and agrees with original labels. **Secondly**, we train an ensemble of 5 models on distinct subsets of S. We use these models to filter S, keeping only examples where all models of the ensemble confidently agree with the labels, resulting in the set S'. Furthermore, as an additional precaution, we retain only the k most extreme samples on each side of $[0, 1]$ to compose $S^\star$, the DDPM training data. In practice, k is in the range $[5000, 60000]$, depending on the experiments. For instance, in Section 5.2, we study the impact of k on the interpolation capabilities.

Leakage and Memorization We believe our method avoids leakage and present rough estimate of the error probability of the ensemble in Appendix D.3. Furthermore, diffusion models can memorize training examples (Somepalli et al., 2022; 2023) and reproduce them at inference. Therefore, we report the synthetic examples with closest neighbors in the training data using a CLIP model (Radford et al., 2021) in Appendix H. Based on these checks, we believe the

Figure 4: *Synthetic* samples generated with multi-guidance using a DDPM trained on extreme images only. According to the evaluation classifier, the "Smiling" likelihood of the pictures lie in $[0.49, 0.51]$.

model does not memorize examples that potentially slipped through the filtering method. Note that we demonstrate interpolation on synthetic datasets in Section 5.5, where we can explicitly ensure there are no examples with mild attributes. In addition to the explicit verification with the CLIP model, interpolation on synthetic data strongly supports our hypothesis that our DDPMs do not interpolate because of leakage or memorization.

5. Interpolation with One Attribute of Interest

In this section, we study the interpolation capabilities of diffusion models, as defined in Section 4.1. Crucially, we train on $S^\star$, a dataset of **extreme** examples only. We focus on 64×64 images and a single latent factor of interest. In Section 5.1, we showcase interpolation results using CelebA's smile attribute (Liu et al., 2015). We reproduce the results for the "Young" and "Blond_Hair"/"Black_Hair" attributes in Section 5.4. Finally, in Section 5.5, we demonstrate interpolation for a synthetic dataset of images. More details about the hyperparameters and compute costs in Appendix C. Although the results in this section use the U-net architecture from Nichol & Dhariwal (2021) with classifier guidance, we still observed interpolation with other architectures and with classifier-free guidance. Further details are provided in Appendix J. Additionally, we present experiments on additional datasets in Appendix K and Appendix L.

To estimate a proxy for the latent factor values in real-world samples, we train EfficientNet classifiers (Tan & Le, 2021), following Okawa et al. (2023). We refer to those classifiers as **evaluation classifiers**. We improve calibration of the classifiers using temperature scaling (Guo et al., 2017) and use the calibrated probability estimates as our proxy, reported in empirical distribution histograms, such as Figure 5.

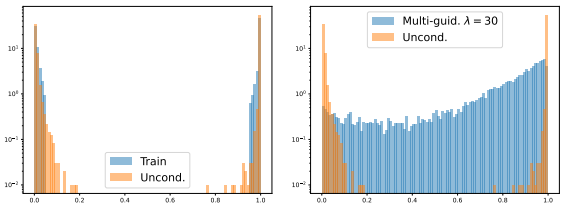


Figure 5: Empirical distribution of pictures according to the evaluation classifier. **Left**: extreme training examples versus samples from the unconditional diffusion. **Right**: Samples from the unconditional model versus images sampled with multi-guidance.

5.1. Smile Interpolation on CelebA

We first focus on the "Smiling" attribute on CelebA as it is relatively balanced, with 48% of samples smiling. Additionally, smiling pictures can intuitively be ranked according to the smiling strength on a scale. We train a regular DDPM, and guidance classifier following Nichol & Dhariwal (2021) on a subset of CelebA, filtered as detailed in Section 4.3.

When sampling 10k examples with multi-guidance ($\lambda_{\text{smile}} = \lambda_{\text{non-smile}} = 30$), we observe a distribution significantly closer to uniform than samples from the unconditional model in Figure 5. Refer to Figure 4 for mildly smiling examples (Appendix G.3 for more). While the distribution of training examples and unconditional samples is similar, we see that multi-guidance successfully synthesizes examples with mild attributes. For evaluation, we compare the empirical distribution of samples against the uniform distribution. The **training distribution** has a mean-squared error (MSE) of 0.56 and Kullback-Leibler divergence (KLD, see Appendix B for details) of 3.17 (lower is better). The histogram of samples from the **unconditional model** trained on extreme examples has an MSE of 0.64 and KLD of 3.5. Nonetheless, the histogram of samples generated with **multi-guidance** has an MSE of 0.13 and KLD of 0.61. Recall that we check for memorization in Appendix H.

5.2. How Little Data Can We Use?

We investigate the degradation in performance as we reduce the dataset size. The initial extreme dataset, detailed in Section 5.1, contains 60k examples. We create subsets containing 30k, 10k, and 5k examples by ranking all extreme samples with the evaluation classifier and selecting the most extreme instances on each side of the interval. For instance, with a subset of 10k examples, we choose the 5k examples with the lowest and highest "Smiling" likelihood. We measure the performance degradation by comparing the empirical distribution against a uniform one. Specifically, we report the MSE and KLD in Table 1. See Appendix G.1 for histograms. Recall that the training distribution with

$\lvert\mathcal{S}_{\text{train}}\rvert$	MSE to uniform ↓	KLD to uniform ↓
60k	0.130	0.607
30k	0.184	0.835
10k	0.186	0.802
5k	0.279	1.185

Table 1: Ablation on interpolation performance when decreasing the training set size (first column). We compute the MSE and Kullback-Leibler divergence (KLD) between the empirical histogram on 10k samples and the uniform distribution. For both MSE and KLD, lower is better.

60k samples achieves an MSE of 0.558 and KLD of 3.161. While the performance decreases as the data size shrinks, we observe that DDPMs retain a qualitative ability to interpolate even when trained on 5k examples. Regarding memorization, it appears that DDPMs trained on 5k examples memorize extreme examples. In Appendix H, we show that the model avoids mere reproduction when trained on larger training sets.

5.3. Sensitivity to the Guidance Parameters

In this section, we study the sensitivity of multi-guidance to variations in λ. We generate 10k samples for varying λ values within the range $[3.5, 10, 30, 50, 75, 100]$ using the DDPM trained on 60k examples, as in Section 5.1.

Interestingly, even with $\lambda = 3.5$, multi-guidance can generate examples with mild attributes. However, we observe that higher values yield better MSE and KLD against uniform. Refer to Appendix G.2 for histograms of the empirical distributions. Using $\lambda = 75$ achieves an MSE of 0.065 and KLD of 0.188, our best results. While the model seem able to interpolate for all values of λ, we found that using the largest values of λ tends to generate images with artifacts more frequently than smaller values of λ. As such, in further experiments, we use $\lambda = 30$ as it achieves interpolation while introducing little artifacts. Nonetheless, this experiment suggests that the interpolation abilities of diffusion models are robust to the choice of guidance parameter. See Appendix I for a discussion of quality and diversity metrics for different values of λ.

5.4. Interpolation on Alternative Attributes

To ensure our results generalize across attributes, we extend our study to interpolation based on the age and hair color on CelebA. Similar to the smile attribute, we compute the distribution of synthetic samples using an evaluation classifier[2]. Regarding hyperparameters, we reuse $\lambda = 30$ from Section 5.3, to assess how our method generalizes to other cases without tuning. We train the DDPM as in Section 5.1. As in previous experiments, DDPMs demonstrate interpolation abilities. Refer to Figure 8 and Appendix G.3 for samples with mild attributes according to the "Young" and hair color attributes. We did not observe any instances of individuals with a clearly divided hair color, featuring half of the hair in black and the other half in blond in the training set.

Age Attribute While 48% of samples are labeled as "Smiling", young people are over-represented in CelebA (78%). After filtering with an ensemble of classifier, we retain the 7k oldest and the 15k youngest examples according to the evaluation classifier. Furthermore, we over-sample the "Old" class by a factor of 2 during training. The distribution of unconditional samples is concentrated at the borders of the interval, while multi-guidance sampling generates examples in the middle of the histogram. The distribution of multi-guidance samples has an MSE of 0.146 and a KLD of 0.656 against uniform, showing slight differences compared to the smile interpolation experiment. Refer to Figure 6a for histograms of samples created with the unconditional model and multi-guidance.

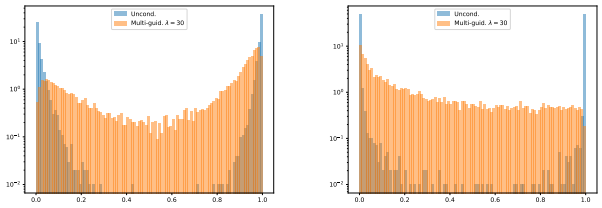
Hair Color Attribute Refer to Figure 6b for the empirical distribution of generated samples. Unlike for the smile and age attribute, the unconditional model trained on blond and black haired individual generates examples with mild attributes (28 samples out of 10k within $[0.2, 0.8]$). A manual review reveals that the majority of these samples feature faces with brown hair. Some instances display clear blond or black hair, while others show variations like blond hair in low light or black hair in high light, which triggers the evaluation classifier. Hence, we still believe that our filtering method avoids leakage. The distribution of multi-guidance samples has an MSE of 0.135 and a KLD of 0.464 against uniform. We observed that the FaRL model was unreliable in predicting the "Blond_Hair" and "Black_Hair" attributes for CelebA (see Appendix D.2), hence we do not use it for filtering.

5.5. Interpolation on a Controlled Synthetic Distribution

We posit that interpolation occurs because, although the model does not encounter the complete $[0, 1]$ range of attributes at training, it observes sufficient variation of extreme attributes (as modeled by δ in Section 4.1). Inspired by Zhao et al. (2018), we design a synthetic dataset of extreme examples with small variations to assess our hypothesis.

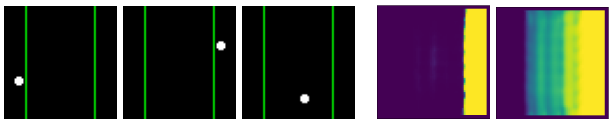
Consider 64×64 images with black background, containing a single white circle, centered in (x, y), with a radius of two

[2]The evaluation classifiers for the "Smiling" and "Young" attributes required minimal calibration (temperature of 1.05), while the "Blond_Hair" and "Black_Hair" classifiers required a temperature of 1.245. See Guo et al. (2017) for details.



(a) (b)

Figure 6: Comparing train distribution and multi-guidance distribution. **Left**: "Young" attribute. **Right**: "Blond_Hair"/"Black_Hair" attribute. There are 28 examples from the unconditional model (out of 10k) in $[0.2, 0.8]$ in the hair color plot. The majority of them have brown hair. See Section 5.4 for more details.



(a) (b)

Figure 7: Synthetic samples from the toy distribution. The green lines are not part of the training examples, and only serve to delimit the training regions (left/right) and unobserved region (center) for the reader. Sub-figure (b) depicts heatmaps where the color at position (x, y) denotes the prediction from the guidance classifier, for images with a circle centered in (x, y). Yellow colors represent high probabilities while low probabilities are shown in dark blue. The left heatmap is for a regular run while the right one is for a classifier trained with spectral normalization. We see that the transition in the unseen region (middle) is much smoother when using spectral normalization. Guidance with the left classifier yields 73% of samples in the middle section while the right classifier yields slightly above 90%.

pixels. Let $z_i \in [0, 1]$ denote the (normalized) x coordinate of the circle's center (see Figure 7a). We construct the training set S_{train} to contain images with $z_i \in [0, 0.2] \cup [0.8, 1.0]$ only. We train a regular DDPM and sample with multi-guidance. The classifier is trained on labels $\tilde{z}_i = \mathbb{1}\{z_i > 0.5\}$. We train 10 classifiers and sample with multi-guidance with $\lambda = 30$ for both left and right classes.

We measure the interpolation ability of our model using the fraction of generated samples with a circle in the middle region of the image ($z_i \in [0.2, 0.8]$). We refer to the fraction as "accuracy". When sampling with multi-guidance, we achieve an accuracy of $78\% \pm 8.7\%$ (mean and standard deviation over 10 runs). The accuracy is computed using an

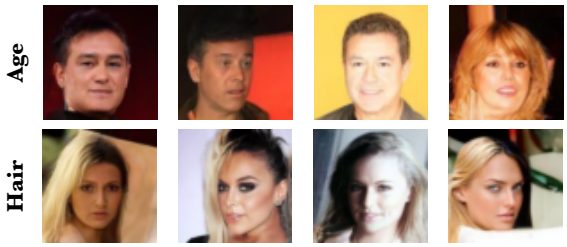


Figure 8: Samples with likelihood in $[0.49, 0.51]$ according to their respective evaluation classifier. Samples in the first row come from the "Young" interpolation experiments, while the second row interpolates according to the hair color.

edge detection algorithm implemented with OpenCV[3]. Note that $4.25\% \pm 2.45\%$ of the samples feature either zero or two circles. This behavior aligns with Zhao et al. (2018), who made analogous observations for other classes of generative models, sometimes synthesizing samples with slightly more or fewer objects than observed in training images.

We observe that spectral normalization (Miyato et al., 2018) on the classifier improves the smoothness of predictions in the unseen region (see Figure 7b) and that sampling with such a smoothed multi-guidance results in more mild samples (above 90%). We obtain an accuracy of $75\% \pm 13.3\%$ with spectral normalization, slightly lower than regular training on average. The large variance is explained by training runs with smooth transition between classes, which achieve significantly higher accuracy, but do not occur at every run. Studying regularization methods for real-world datasets is left for future work.

6. Interpolation with Two Factors of Interest

Having explored distributions with a single variable, we extend our investigation to interpolation with two variables, specifically, the "Smiling" and "Young" attributes. Due to space constraints, a detailed discussion is deferred to Appendix F. While the model does not sample uniformly on the $[0, 1]^2$ square, Figure 9 suggests that DDPMs can also interpolate on multiple variables. We believe that training on synthetic samples and/or regularization could bring the distribution closer to uniform.

7. What about Extrapolation?

In this study, we focus on zero-shot interpolation between extreme cases, i.e. we assume access to extremal samples $\underline{z}, \overline{z} \in \mathcal{D}$ s.t. $\underline{z} \precsim_\epsilon z_{intra} \precsim_\epsilon \overline{z}$,. On a high level, extrapolation is about generating examples with latent variables z located outside the convex hull of $z_i \in \mathcal{D}$, i.e. $z_{extra} \precsim_\epsilon \underline{z}$ or $\overline{z} \precsim_\epsilon z_{extra}$. We hypothesize that extrapolation for DDPMs

[3] https://opencv.org

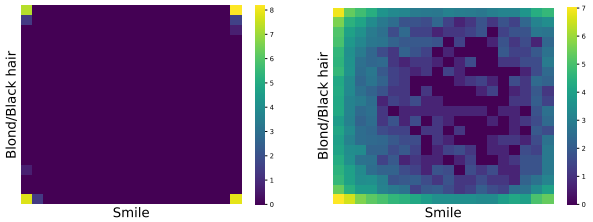


Figure 9: Log-scale heatmaps depicting results of two-variable interpolation. **(Left)**: unconditional model. **(Right)** multi-guidance with $\lambda = 30$ for all four corners. Although multi-guidance may not cover the space as comprehensively in higher dimensions, DDPM still exhibits interpolation capabilities.

requires two key elements. First, the unconditional diffusion model should be capable to generate examples significantly different from the training distribution. Second, we require a way to guide the diffusion model toward these distant examples. Impressive work by Zhu et al. (2023) on few-shot generalization suggests that DDPMs indeed learn generic geometric priors, allowing them to sample from completely unrelated domains, if guided in that direction by using a set of reference images. Unfortunately, in the zero shot setting it is difficult to create a guidance for latent features, as classifiers trained with cross-entropy tend to find the simplest model that explains the data, a phenomenon referred to as *simplicity bias* (Shah et al., 2020; Jacobsen et al., 2020). One option is to directly tackle the simplicity bias, using algorithm such as developed by Pagliardini et al. (2022), however further constrained towards the desired latent factors. Alternatively, we believe that a classifier trained on synthetic data could guide diffusion models far from the training support. Our preliminary experiments in Appendix E support this claim, where we successfully generated faces with green lips, despite the unconditional diffusion model lacking exposure to such instances. Nonetheless, the question of zero shot extrapolation far from the training domain is complex and left for future work.

8. Discussion

Our work studies interpolation abilities of DDPMs. Notably, we observed that in both real-world and synthetically controlled scenarios, DDPMs are capable of interpolation, and generate more diverse data than what they were trained on. We achieve interpolation through the multi-guidance procedure (Section 4.2), and demonstrate its applicability across various datasets, including across multiple joint latent factors. Our work contributes to the growing body of evidence suggesting that the inductive biases of diffusion models align well with real-world signals, particularly in image data, and show that a simple modification to the sampling procedure unlocks the interpolation abilities of DDPMs.

8.1. Where are the Interpolation Capabilities coming from?

We believe that understanding the interpolation, and more generally the out-of-distribution capabilities of DMs is an exciting area of study. Our standing hypothesis regarding those capabilities is strongly influenced by the work of Zhu et al. (2023). In DDIM, Song et al. (2022) demonstrate that any diffusion model, even an untrained one, can generate a latent representation leading to any image when sampled deterministically with DDIM. Therefore, in principle, **every diffusion model admits sampling trajectories that result in any real image**. While most sampling paths are unlikely to be encountered in practice, Zhu et al. (2023) demonstrated that given a few out-of-domain images, DMs can generate out-of-domain images, even if the domain in question is significantly different from the training distribution. Crucially, the work of Zhu et al. (2023) defines a novel guidance procedure using a few out-of-domain images. Impressively, Zhu et al. (2023) successfully generate human faces with a DM trained on dog pictures. This surprising result suggests that new sampling procedures might be sufficient to unlock out-of-domain generalization. In this work, we demonstrate that DDPMs can generate samples lying outside of their training distribution, without requiring additional samples (zero-shot result). Importantly, we assume there exist underlying continuous latent variables that determine the realization of certain attributes. We further assume that only part of the latent variable support is observed during training (Section 4.1).

8.2. Limitations and Future Work

We defer the exploration of few-shot extrapolation, e.g. through synthetic data, to future work. Additionally, the interpolation between certain attributes is ill-defined. For instance, zero-shot interpolating between blond and black hair might produce faces with half black, half blond hair, shades of brown hair, or other results. Thus, we believe that few-shot learning could prune unwanted interpolation behavior. As for extrapolation, this could potentially be tackled with synthetic data. Moreover, we observed in Section 5.5 that spectral normalization improves the interpolation ability on synthetic images. Exploring this phenomenon more in depth, as well as other means of regularization is a promising venue for future work. While our approach allows sampling images with mild attributes, reliable sampling over narrow range of the latents remains elusive. Such ability could allow specifying a very precise skin tone for example. Therefore, future work may explore alternative sampling algorithms, such as in Markov Chain Monte Carlo (MCMC) methods, as studied in Du et al. (2023b). Finally, we observed that DDPMs struggle to interpolate on the continuous range for certain datasets, such as rotation of objects in 3D.

Impact Statement

Our research on the interpolation abilities of diffusion models holds promise for improving fairness in machine learning. By showcasing the model's capacity to generate diverse attributes covering the range of a latent factor from a subset of examples, our work can contribute to mitigating biases in training data. However, it's essential to recognize potential misuse. While our work is on low-dimensional images of insufficient quality for deception, the method could potentially be improved and exploited for deepfake generation in higher resolutions. Such concerns are inherent to the development of most novel generative modeling techniques. Importantly, our focus is not technically advancing the state of the art in image generation. Instead, we merely study the properties of diffusion models and show what is *already* possible with current models.

This demonstration also has concrete implications on the use of generative models as tools in creating new content. At time of writing there are ongoing lawsuits concerning the generation of copyrighted content (Marcus & Southen, 2024) and while the examples discussed in our citation are clearly infringing even *if* they were interpolated in the manner we discuss (which we find unlikely), the ability to interpolate and potentially *extrapolate* across latent factors hints at the possibility at truly novel synthesized content - which then itself would complicate discussions around the level of derivative vs. original "work" present in the generated content.

In no way should our work be cited as "proof" that any models involved in such lawsuits are not infringing. It requires further development in dataset provenance attestation and legal frameworks to make any such claims. Presumably, our findings will play a rule in developing these legal frameworks. Until then, we think it is important to err on the side of caution and not train production models on any content for which the artists or copyright holders have not explicitly consented to be included as part of generative model training.

Further, while we are excited about the idea of using interpolated factors to de-bias classifiers, e.g. by developing methods for holding all factors except the interpolated one constant in order to remove spurious correlations with e.g. gender or skin color, the model used for such debiasing might itself be flawed and/or introduce new biases. This means any such use requires careful verification and ideally further work on provably selective editing via interpolation.

Acknowledgements

We appreciate the constructive discussions with the reviewers during the rebuttal period, which contributed to the final quality of this paper. We are also grateful to the area chairs, senior area chairs, and the organizers of ICML'24 for their work. Additionally, we thank Ronan Collobert for his questions and suggestions. This work was supported by the Hasler Foundation Program: Hasler Responsible AI (project number 21043). This research was sponsored by the Army Research Office and was accomplished under Grant Number W911NF-24-1-0048. This work was supported by the Swiss National Science Foundation (SNSF) under grant number 200021_205011.

References

Berthelot, D., Raffel, C., Roy, A., and Goodfellow, I. Understanding and improving interpolation in autoencoders via an adversarial regularizer. *arXiv preprint arXiv:1807.07543*, 2018.

Besserve, M., Sun, R., Janzing, D., and Schölkopf, B. A theory of independent mechanisms for extrapolation in generative models, 2021.

Chen, X., Duan, Y., Houthooft, R., Schulman, J., Sutskever, I., and Abbeel, P. Infogan: Interpretable representation learning by information maximizing generative adversarial nets, 2016.

Dhariwal, P. and Nichol, A. Diffusion models beat gans on image synthesis, 2021.

Du, X., Sun, Y., Zhu, X., and Li, Y. Dream the impossible: Outlier imagination with diffusion models, 2023a.

Du, Y. and Mordatch, I. Implicit generation and modeling with energy based models. In Wallach, H., Larochelle, H., Beygelzimer, A., d'Alché-Buc, F., Fox, E., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019. URL https://proceedings.neurips.cc/paper_files/paper/2019/file/378a063b8fdb1db941e34f4bde584c7d-Paper.pdf.

Du, Y. and Mordatch, I. Implicit generation and generalization in energy-based models, 2020.

Du, Y., Li, S., and Mordatch, I. Compositional visual generation and inference with energy based models, 2020.

Du, Y., Li, S., Sharma, Y., Tenenbaum, J. B., and Mordatch, I. Unsupervised learning of compositional energy concepts, 2021a.

Du, Y., Li, S., Tenenbaum, J., and Mordatch, I. Improved contrastive divergence training of energy based models, 2021b.

Du, Y., Durkan, C., Strudel, R., Tenenbaum, J. B., Dieleman, S., Fergus, R., Sohl-Dickstein, J., Doucet, A., and Grathwohl, W. Reduce, reuse, recycle: Compositional generation with energy-based diffusion models and mcmc, 2023b.

Duchi John, S. J. Multivariate normal distribution: Kullback-leibler divergence, 05 2020. URL https://statproofbook.github.io/P/mvn-kl.html. Accessed on June 6, 2024.

Fang, A., Jose, A. M., Jain, A., Schmidt, L., Toshev, A., and Shankar, V. Data filtering networks, 2023.

Frege, G. and Gabriel, G. Philosophical and mathematical correspondence. *(No Title)*, 1980.

Gandikota, R., Orgad, H., Belinkov, Y., Materzyńska, J., and Bau, D. Unified concept editing in diffusion models, 2023.

Georgopoulos, M., Chrysos, G., Pantic, M., and Panagakis, Y. Multilinear latent conditioning for generating unseen attribute combinations, 2020.

Graham, M. S., Pinaya, W. H. L., Tudosiu, P.-D., Nachev, P., Ourselin, S., and Cardoso, M. J. Denoising diffusion models for out-of-distribution detection, 2023a.

Graham, M. S., Pinaya, W. H. L., Wright, P., Tudosiu, P.-D., Mah, Y. H., Teo, J. T., Jäger, H. R., Werring, D., Nachev, P., Ourselin, S., and Cardoso, M. J. Unsupervised 3d out-of-distribution detection with latent diffusion models, 2023b.

Guo, C., Pleiss, G., Sun, Y., and Weinberger, K. Q. On calibration of modern neural networks, 2017.

Güngör, A., Dar, S. U., Şaban Öztürk, Korkmaz, Y., Bedel, H. A., Elmas, G., Ozbey, M., and Çukur, T. Adaptive diffusion priors for accelerated mri reconstruction. *Medical Image Analysis*, 88:102872, 2023. ISSN 1361-8415. doi: https://doi.org/10.1016/j.media.2023.102872. URL https://www.sciencedirect.com/science/article/pii/S1361841523001329.

Härkönen, E., Hertzmann, A., Lehtinen, J., and Paris, S. Ganspace: Discovering interpretable gan controls. *Advances in neural information processing systems*, 33: 9841–9850, 2020.

Heusel, M., Ramsauer, H., Unterthiner, T., Nessler, B., Klambauer, G., and Hochreiter, S. Gans trained by a two time-scale update rule converge to a nash equilibrium. *CoRR*, abs/1706.08500, 2017. URL http://arxiv.org/abs/1706.08500.

Higgins, I., Matthey, L., Pal, A., Burgess, C., Glorot, X., Botvinick, M., Mohamed, S., and Lerchner, A. 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.

Ho, J. and Salimans, T. Classifier-free diffusion guidance, 2022.

Ho, J., Jain, A., and Abbeel, P. Denoising diffusion probabilistic models, 2020.

Hyvärinen, A. Estimation of non-normalized statistical models by score matching. *J. Mach. Learn. Res.*, 6:695–709, dec 2005. ISSN 1532-4435.

Jacobsen, J.-H., Behrmann, J., Zemel, R., and Bethge, M. Excessive invariance causes adversarial vulnerability, 2020.

Johnson, J., Hariharan, B., van der Maaten, L., Fei-Fei, L., Zitnick, C. L., and Girshick, R. Clevr: A diagnostic dataset for compositional language and elementary visual reasoning, 2016.

Kadkhodaie, Z., Guth, F., Simoncelli, E. P., and Mallat, S. Generalization in diffusion models arises from geometry-adaptive harmonic representation, 2023.

Kang, M., Zhu, J.-Y., Zhang, R., Park, J., Shechtman, E., Paris, S., and Park, T. Scaling up gans for text-to-image synthesis. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023.

Karras, T., Laine, S., and Aila, T. A style-based generator architecture for generative adversarial networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 4401–4410, 2019.

Kim, G., Kwon, T., and Ye, J. C. Diffusionclip: Text-guided diffusion models for robust image manipulation, 2022.

Kingma, D. P. and Ba, J. Adam: A method for stochastic optimization, 2017.

Kingma, D. P. and Dhariwal, P. Glow: Generative flow with invertible 1x1 convolutions. *Advances in neural information processing systems*, 31, 2018.

Kong, C., Jeon, D., Kwon, O., and Kwak, N. Leveraging off-the-shelf diffusion model for multi-attribute fashion image manipulation, 2022.

Lakshminarayanan, B., Pritzel, A., and Blundell, C. Simple and scalable predictive uncertainty estimation using deep ensembles, 2017.

Lin, T.-Y., Maire, M., Belongie, S., Bourdev, L., Girshick, R., Hays, J., Perona, P., Ramanan, D., Zitnick, C. L., and Dollár, P. Microsoft coco: Common objects in context, 2015.

Liu, N., Li, S., Du, Y., Tenenbaum, J. B., and Torralba, A. Learning to compose visual relations, 2021.

Liu, N., Li, S., Du, Y., Torralba, A., and Tenenbaum, J. B. Compositional visual generation with composable diffusion models, 2023a.

Liu, Z., Luo, P., Wang, X., and Tang, X. Deep learning face attributes in the wild. In *Proceedings of International Conference on Computer Vision (ICCV)*, December 2015.

Liu, Z., Zhou, J. P., Wang, Y., and Weinberger, K. Q. Unsupervised out-of-distribution detection with diffusion inpainting, 2023b.

Luce, R. D. Semiorders and a theory of utility discrimination. *Econometrica*, 24(2):178–191, 1956. ISSN 00129682, 14680262. URL http://www.jstor.org/stable/1905751.

Luo, C. Understanding diffusion models: A unified perspective, 2022.

Marcus, G. and Southen, R. Generative AI Has a Visual Plagiarism Problem - IEEE Spectrum, 2024. URL https://spectrum.ieee.org/midjourney-copyright.

Miyato, T., Kataoka, T., Koyama, M., and Yoshida, Y. Spectral normalization for generative adversarial networks, 2018.

Molloy, M. and Reed, B. *The Chernoff Bound*, pp. 43–46. Springer Berlin Heidelberg, Berlin, Heidelberg, 2002. ISBN 978-3-642-04016-0. doi: 10.1007/978-3-642-04016-0_5. URL https://doi.org/10.1007/978-3-642-04016-0_5.

Ngiam, J., Chen, Z., Koh, P. W., and Ng, A. Y. Learning deep energy models. In *Proceedings of the 28th international conference on machine learning (ICML-11)*, pp. 1105–1112, 2011.

Nichol, A. and Dhariwal, P. Improved denoising diffusion probabilistic models. *CoRR*, abs/2102.09672, 2021. URL https://arxiv.org/abs/2102.09672.

Okawa, M., Lubana, E. S., Dick, R. P., and Tanaka, H. Compositional abilities emerge multiplicatively: Exploring diffusion models on a synthetic task, 2023.

Pagliardini, M., Jaggi, M., Fleuret, F., and Karimireddy, S. P. Agree to disagree: Diversity through disagreement for better transferability, 2022.

Paplham, J. and Franc, V. A call to reflect on evaluation practices for age estimation: Comparative analysis of the state-of-the-art and a unified benchmark, 2023.

Pawlowski, N., Coelho de Castro, D., and Glocker, B. Deep structural causal models for tractable counterfactual inference. In Larochelle, H., Ranzato, M., Hadsell, R., Balcan, M., and Lin, H. (eds.), *Advances in Neural Information Processing Systems*, volume 33, pp. 857–869. Curran Associates, Inc., 2020. URL https://proceedings.neurips.cc/paper_files/paper/2020/file/0987b8b338d6c90bbedd8631bc499221-Paper.pdf.

Peebles, W. and Xie, S. Scalable diffusion models with transformers, 2023.

Pelletier, F. J. The principle of semantic compositionality. *Topoi*, 13(1):11–24, Mar 1994. ISSN 1572-8749. doi: 10.1007/BF00763644. URL https://doi.org/10.1007/BF00763644.

Pirlot, M. and Vincke, P. *Semiorders*. Theory and Decision Library B. Springer, Dordrecht, Netherlands, 1997 edition, June 1997.

Radford, A., Metz, L., and Chintala, S. Unsupervised representation learning with deep convolutional generative adversarial networks, 2016.

Radford, A., Kim, J. W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., Krueger, G., and Sutskever, I. Learning transferable visual models from natural language supervision, 2021.

Rajabi, A. and Garibay, O. O. Towards fairness in ai: Addressing bias in data using gans. In Stephanidis, C., Kurosu, M., Chen, J. Y. C., Fragomeni, G., Streitz, N., Konomi, S., Degen, H., and Ntoa, S. (eds.), *HCI International 2021 - Late Breaking Papers: Multimodality, eXtended Reality, and Artificial Intelligence*, pp. 509–518, Cham, 2021. Springer International Publishing. ISBN 978-3-030-90963-5.

Ramesh, A., Dhariwal, P., Nichol, A., Chu, C., and Chen, M. Hierarchical text-conditional image generation with clip latents, 2022.

Recht, B., Roelofs, R., Schmidt, L., and Shankar, V. Do imagenet classifiers generalize to imagenet?, 2019.

Rombach, R., Blattmann, A., Lorenz, D., Esser, P., and Ommer, B. High-resolution image synthesis with latent diffusion models, 2022.

Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein,

M., Berg, A. C., and Fei-Fei, L. Imagenet large scale visual recognition challenge, 2015.

Saharia, C., Chan, W., Saxena, S., Li, L., Whang, J., Denton, E., Ghasemipour, S. K. S., Ayan, B. K., Mahdavi, S. S., Lopes, R. G., Salimans, T., Ho, J., Fleet, D. J., and Norouzi, M. Photorealistic text-to-image diffusion models with deep language understanding, 2022.

Sajjadi, M. S. M., Bachem, O., Lucic, M., Bousquet, O., and Gelly, S. Assessing generative models via precision and recall, 2018.

Salimans, T., Goodfellow, I., Zaremba, W., Cheung, V., Radford, A., and Chen, X. Improved techniques for training gans, 2016.

Sattigeri, P., Hoffman, S. C., Chenthamarakshan, V., and Varshney, K. R. Fairness gan: Generating datasets with fairness properties using a generative adversarial network. *IBM Journal of Research and Development*, 63(4/5):3:1–3:9, 2019. doi: 10.1147/JRD.2019.2945519.

Shah, H., Tamuly, K., Raghunathan, A., Jain, P., and Netrapalli, P. The pitfalls of simplicity bias in neural networks, 2020.

Shoshan, A., Bhonker, N., Kviatkovsky, I., and Medioni, G. Gan-control: Explicitly controllable gans. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 14083–14093, October 2021.

Sohl-Dickstein, J., Weiss, E. A., Maheswaranathan, N., and Ganguli, S. Deep unsupervised learning using nonequilibrium thermodynamics, 2015.

Somepalli, G., Singla, V., Goldblum, M., Geiping, J., and Goldstein, T. Diffusion art or digital forgery? investigating data replication in diffusion models, 2022.

Somepalli, G., Singla, V., Goldblum, M., Geiping, J., and Goldstein, T. Understanding and mitigating copying in diffusion models, 2023.

Song, J., Meng, C., and Ermon, S. Denoising diffusion implicit models, 2022.

Song, Y. and Ermon, S. Generative modeling by estimating gradients of the data distribution. In Wallach, H., Larochelle, H., Beygelzimer, A., d'Alché-Buc, F., Fox, E., and Garnett, R. (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.

Song, Y. and Ermon, S. Improved techniques for training score-based generative models, 2020.

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

Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., and Wojna, Z. Rethinking the inception architecture for computer vision, 2015.

Tan, M. and Le, Q. V. Netv2: Smaller models and faster training. *CoRR*, abs/2104.00298, 2021. URL https://arxiv.org/abs/2104.00298.

Tan, S., Shen, Y., and Zhou, B. Improving the fairness of deep generative models without retraining, 2021.

Upchurch, P., Gardner, J., Pleiss, G., Pless, R., Snavely, N., Bala, K., and Weinberger, K. Deep feature interpolation for image content changes. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 7064–7073, 2017.

van den Oord, A., Kalchbrenner, N., Vinyals, O., Espeholt, L., Graves, A., and Kavukcuoglu, K. Conditional image generation with pixelcnn decoders, 2016.

Vincent, P. A connection between score matching and denoising autoencoders. *Neural Computation*, 23(7):1661–1674, 2011. doi: 10.1162/NECO_a_00142.

Wang, C. and Golland, P. Interpolating between images with diffusion models. 2023.

Welling, M. and Teh, Y. W. Bayesian learning via stochastic gradient langevin dynamics. In *Proceedings of the 28th International Conference on International Conference on Machine Learning*, ICML'11, pp. 681–688, Madison, WI, USA, 2011. Omnipress. ISBN 9781450306195.

Wiedemer, T., Mayilvahanan, P., Bethge, M., and Brendel, W. Compositional generalization from first principles, 2023.

Williams, E. Semantic vs. syntactic categories. *Linguistics and philosophy*, pp. 423–446, 1983.

Xie, J., Lu, Y., Zhu, S.-C., and Wu, Y. N. A theory of generative convnet, 2016.

Yu, J., Xu, Y., Koh, J. Y., Luong, T., Baid, G., Wang, Z., Vasudevan, V., Ku, A., Yang, Y., Ayan, B. K., Hutchinson, B., Han, W., Parekh, Z., Li, X., Zhang, H., Baldridge, J., and Wu, Y. Scaling autoregressive models for content-rich text-to-image generation, 2022.

Yue, Z., Wang, J., Sun, Q., Ji, L., Chang, E. I.-C., and Zhang, H. Exploring diffusion time-steps for unsupervised representation learning, 2024.

Zhao, S., Ren, H., Yuan, A., Song, J., Goodman, N., and Ermon, S. Bias and generalization in deep generative models: An empirical study, 2018.

Zheng, Y., Yang, H., Zhang, T., Bao, J., Chen, D., Huang, Y., Yuan, L., Chen, D., Zeng, M., and Wen, F. General facial representation learning in a visual-linguistic manner, 2022.

Zhu, Y., Wu, Y., Deng, Z., Russakovsky, O., and Yan, Y. Unseen image synthesis with diffusion models, 2023.

A. Additional Background on Diffusion Models

We use the denoising diffusion probabilistic model (DDPM) formalism, introduced in Ho et al. (2020) and include relevant background from Nichol & Dhariwal (2021); Dhariwal & Nichol (2021); Du et al. (2023b). For a clear long format introduction, please refer to the excellent work of Luo (2022).

A.1. Forward Process

Diffusion models approximate the data distribution p_{data} by reversing a Markov Chain (MC), called the forward process, that transforms p_{data} into white noise. The forward MC does not have any learnable parameters and is defined as

$$p_{\text{fwd}}(x_0, x_1, ..., x_T) = p_0(x_0) \prod_{t=1}^{T} p_t(x_t|x_{t-1}), \tag{3}$$

where $0 \leq t \leq T$ and the variables $x_1, ..., x_T$ are corrupted with Gaussian noise. Specifically, for $t > 0$, $p_t(x_t|x_{t-1}) := \mathcal{N}(x_t; \sqrt{1-\beta_t}x_{t-1}, \beta_t \mathbf{I})$ and $p_0(x_0) = p_{\text{data}}(x_0)$. The $(\beta_t)_{t=1}^T$ coefficients are called the *noise schedule* and are picked such that $p_T(x_T) \approx \mathcal{N}(x_T; 0, \mathbf{I})$. Using Gaussian transitions allows one to compute $x_t|x_0$ in one step. From Ho et al. (2020), defining $\alpha_t := 1 - \beta_t$, $\bar{\alpha}_t = \prod_{i=1}^{t} \alpha_i$ and $\epsilon \sim \mathcal{N}(0, \mathbf{I})$, we get

$$x_t = \sqrt{\bar{\alpha}_t} x_0 + \sqrt{1 - \bar{\alpha}_t}\epsilon, \tag{4}$$

A.2. Backward Process

The backward process is used to sample from p_{data} by iteratively denoising samples from $p_T \sim \mathcal{N}(0, \mathbf{I})$. Although $p_{\text{fwd}}(x_{t-1}|x_t)$ depends on the entire data distribution, $p_{\text{fwd}}(x_{t-1}|x_t, x_0)$ is Gaussian (Ho et al., 2020). Using Bayes theorem, one finds $p_{\text{fwd}}(x_{t-1}|x_t, x_0) = \mathcal{N}\left(x_{t-1}; \tilde{\mu}(x_t, x_0), \tilde{\beta}_t \mathbf{I}\right)$, where

$$\tilde{\mu}(x_t, x_0) := \frac{\sqrt{\bar{\alpha}_t}}{1 - \bar{\alpha}_t} \left(x_0 + (1 - \bar{\alpha}_{t-1})x_t\right), \tag{5}$$

$$\tilde{\beta}_t := \frac{1 - \bar{\alpha}_{t-1}}{1 - \bar{\alpha}_t}\beta_t. \tag{6}$$

DDPMs approximate $p_{\text{fwd}}(x_{t-1}|x_t, x_0)$ with a distribution $p_\theta(x_{t-1}|x_t)$, implemented with a denoising neural network. Formally, $p_\theta(x_{t-1}|x_t) \sim \mathcal{N}\left(x_{t-1}; \mu_\theta(x_t, t), \Sigma_\star(x_t, t)\right)$. Ho et al. (2020) use a fixed covariance, being either $\Sigma_\star(x_t, t) = \beta_t \mathbf{I}$ or $\Sigma_\star(x_t, t) = \tilde{\beta}_t \mathbf{I}$. The neural network typically computes $\mu_\theta(x_t, t)$ in one of three ways. First, output $\mu_\theta(x_t, t)$ directly. Second, approximate $x_\theta(x_t, t) \approx x_0$ and compute $\mu_\theta(x_t, t)$ by replacing x_0 in Equation (5). Finally, predict the noise to remove from x_t to obtain x_0, denoted by $\epsilon_\theta(x_t, t)$. Manipulating Equation (4) and Equation (5) yields

$$\mu_\theta(x_t, t) = \frac{1}{\sqrt{\alpha_t}} \left(x_t - \frac{\beta_t}{\sqrt{1 - \bar{\alpha}_t}}\epsilon_\theta(x_t, t)\right). \tag{7}$$

Note that Ho et al. (2020) obtain their best results when modeling $\epsilon_\theta(x_t, t)$. While Ho et al. (2020) keep $\Sigma_\star(x_t, t)$ fixed, Nichol & Dhariwal (2021) use a learnable $\Sigma_\star(x_t, t) = \Sigma_\theta(x_t, t)$, computed as

$$\Sigma_\theta(x_t, t) = \exp\left(v \log \beta_t + (1 - v) \log \tilde{\beta}_t\right), \tag{8}$$

where v comes from a neural network.

A.3. Learning Objective

One can directly optimize the variational bound L_{vlb}, which writes as

$$\begin{aligned}
L_{\text{vlb}} &:= L_0 + L_1 + ... + L_{T-1} + L_T \\
L_0 &:= \mathbb{E}_{p_{\text{fwd}}(x_1|x_0)}\left[-\log p_\theta(x_0|x_1)\right] \\
L_{t-1} &:= D_{\text{KL}}\left(p(x_{t-1}|x_t, x_0) \parallel p_\theta(x_{t-1}|x_t)\right) \\
L_T &:= D_{\text{KL}}\left(p(x_T|x_0) \parallel p_T(x_T)\right),
\end{aligned} \tag{9}$$

where the Kullback-Leibler divergence (KLD) D_{KL} between Gaussians admits a simple expression (Duchi John, 2020). Choosing a noise schedule $(\beta_t)_{t=1}^T$ such that x_T is white noise makes L_T independent of any learnable parameters. While optimizing L_{vlb} is possible, Ho et al. (2020) obtained their best results in terms of *Fréchet Inception Distance* (FID) (Heusel et al., 2017) using a fixed covariance $\Sigma_\star(x_t, t)$ and the objective $L_{\text{simple}} := \mathbb{E}_{t,x_0,\epsilon}\left[\|\epsilon - \epsilon_\theta(x_t, t)\|^2\right]$, where t is picked uniformly at random in $1, ..., T$, $x_0 \sim p_{\text{data}}$ and $\epsilon \sim \mathcal{N}(0, I)$ is the noise used to compute x_t through Equation (4). As shown in Ho et al. (2020), L_{vlb} is equivalent to L_{simple} up to a reweighing of the L_t terms. While competitive in terms of FID, training DDPMs with L_{simple} achieves lower likelihood than likelihood-based models such as van den Oord et al. (2016). Nichol & Dhariwal (2021) achieved competitive likelihood while retaining the FID of Ho et al. (2020) with three improvements. First, a novel noise schedule $(\beta_t)_{t=1}^T$, especially suited to 64×64 and 32×32 images, defined as

$$f(t) = \cos^2\left(\frac{1 + \frac{t}{T} + s}{1 + s} \cdot \frac{\pi}{2}\right),\ \bar{\alpha}_t = \frac{f(t)}{f(0)}. \tag{10}$$

Second, using the learned covariance $\Sigma_\theta(x_t, t)$ from Equation (8), and third, optimizing $L_{\text{hybrid}} := L_{\text{simple}} + \lambda L_{\text{vlb}}$. Importantly, they used a stop-gradient on $\mu_\theta(x_t, t)$ in L_{vlb} so that it drives the learning of $\Sigma_\theta(x_t, t)$ only. Nichol & Dhariwal (2021) used $\lambda = 0.001$ so that L_{vlb} does not overwhelm L_{simple}.

B. Additional Background on the Kullback-Leibler Divergence

In generative modelling, one wants to approximate an empirical distribution p_{data} using a learned distribution p_θ. To do so, a common objective is the Kullback-Leibler divergence (KLD), defined in the discrete case as

$$D_{\text{KL}}(p||q) := \sum_x p(x) \log \frac{p(x)}{q(x)}. \tag{11}$$

The Kullback-Leibler divergence is relevant for generative modeling because it is minimized when p and q are equal. Note that the KLD is not symmetric with respect to p and q. Hence, when approximating p_{data}, it remains important to realize whether we should minimize $D_{KL}(p_{\text{data}}||p_\theta)$, called **forward** KLD or $D_{KL}(p_\theta||p_{\text{data}})$, called **reverse** KLD. To understand the difference between the two, it is good to think of extreme examples. Those examples are inspired from this useful resource.

Forward KLD Imagine that sample x_0 captures a large portion of the total mass of p_{data}, e.g. if x_0 comes from one of the modes of the distribution. At the same time, imagine that the learned distribution p_θ does not capture this part of the distribution, hence $p_\theta(x_0) \approx 0$. Then, the term associated with x_0 in the KLD, i.e. $p_{\text{data}}(x_0) \log \frac{p_{\text{data}}(x_0)}{p_\theta(x_0)}$ will be large. As such, the forward KLD is called **zero-avoiding**, because to minimize the forward KLD, p_θ should assign positive mass to all x such that $p_{\text{data}}(x) > 0$.

Reverse KLD Imagine that we want to minimize $D_{KL}(p_\theta||p_{\text{data}})$. Assume that $p_{\text{data}}(x_0) \approx 0$. Recall that x_0 only appears in the term $p_\theta(x_0) \log \frac{p_\theta(x_0)}{p_{\text{data}}(x_0)}$ of the KLD. If the learned distribution assigns weight to x_0, then the term will be large. Therefore, the reverse KLD is said to be **zero-forcing**, i.e. the learned distribution is discouraged to put any mass on unlikely samples.

KLD used by Diffusion Models The KL formulation used in DDPMs (Ho et al., 2020; Nichol & Dhariwal, 2021) is the forward KL divergence, i.e. assuming the model converges and has enough capacity, it should cover all modes of the data distribution.

KLD used in Evaluations Since we train DDPMs with the forward KL divergence, we also compare the empirical distribution of histograms using the forward KLD: $D_{KL}(p_{\text{hist}}||p_{\mathcal{U}})$, where p_{hist} is the empirical distribution with respect to samples and $p_{\mathcal{U}}$ is uniform with the same support as p_{hist}.

C. Details on Compute and Training

Compute Costs Training the unconditional diffusion model on 64×64 images required using 2 GPUs and 18GiB of memory on each when using FP16. Training for 250k steps ranged from 19 to 21 hours on A100 40GiB or RTX4090

respectively. In parallel, training the classifier for 150k steps took 8 hours on a single GPU in FP32 (less stable) and required 12 GiB of memory. In contrast, training the unconditional model on 128×128 images required 41GiB of memory and took 34 hours, while the classifier took 13 hours to train on 128×128 images. Finally, multi-guidance does not introduce additional cost at sampling versus regular classifier guidance, since the score with respect to all classes can be computed in a single forward/backward pass.

Hyperparameters In this work, we train DDPMs with classifier guidance following best practices from Nichol & Dhariwal (2021). We train the diffusion model for 250k steps with learned denoising process variance, a learning rate of $1e-4$, no weight decay, an EMA rate of 0.9999, 4000 diffusion steps, and the cosine noise schedule presented in Equation (10), well-suited for 64×64 images. Our model also predicts the variance of the reverse process and optimizes the objective L_{hybrid}. We use time embedding vectors of dimension 128 and 3 residual blocks per resolution, with attention at resolution 32, 16, and 8. We train the guidance classifier for 150k steps, with a learning rate of $3e-4$ and no weight decay. Both models are trained with the Adam optimizer (Kingma & Ba, 2017). Further details on these parameters can be found in Nichol & Dhariwal (2021). Our GitHub repository[4] contains the code and configuration to reproduce our results. We sample new images using 250 steps instead of 4000, as it induces minimal change in FID.

D. Evaluation and Data Processing Pipelines

D.1. Extracting an Extreme "Smiling" Subset from CelebA

Despite being labeled with binary attributes, the samples in real-world datasets such as CelebA (Liu et al., 2015) cover a wide range of variations. For instance, the CelebA dataset contains clearly as well as mildly smiling faces, in the sense of Section 4.1. In this section, we present our filtering method for the "Smiling" attribute.

Our approach involves two steps and is inspired by Lakshminarayanan et al. (2017); Fang et al. (2023). Initially, we train a smile classifier on the entire CelebA, denoted as $\mathcal{D} = \{(x_i, y_i) : x_i \in \mathbb{R}^d, y_i \in \{0, 1\}, 1 \leq i \leq N\}$, where $N \approx 163$k. We filter the dataset to keep examples where the classifier is confident and agrees with the CelebA labels. That is, when $p_\theta(y_i|x_i) > \tau_1$. The resulting dataset is denoted as $\mathcal{D}_{\text{filter}}$. In the second step, we randomly extract 5 subsets from $\mathcal{D}_{\text{filter}}$ and train 5 classifiers, denoted as p_{θ_k}, for $1 \leq k \leq 5$. The final dataset, $\mathcal{D}^\star \subseteq \mathcal{D}$, is made of examples where the 5 classifiers are confident and agree with the original labels, i.e., $p_{\theta_k}(y_i|x_i) > \tau_2$. The thresholds τ_1 and τ_2 are chosen such that both the original classifier and the ensemble of 5 achieve a precision of 1 on a held-out, manually labeled dataset of 200 examples. We manually construct this held-out set with samples where people are either clearly smiling (e.g. large grin with clearly visible teeth) or clearly non-smiling. We explicitly discard samples where there we are in doubt whether the person is smiling or not. Additionally, we label the images three times to minimize the risk of mistake.

After training all the classifiers, we visualized examples from the validation set where the ensemble *confidently* disagrees with the CelebA labels. There were no such images labeled as smiling, yet around 7000 images labeled as non-smiling were misclassified. Upon manually inspecting 500 of these, we agreed with the classifier's prediction more than the original labels. This comforts us in the soundness of our method. Subsequently, we examined 500 randomly chosen samples from $\mathcal{D}^\star \subseteq \mathcal{D}$ for each class, and concluded that all of them correspond to extreme examples, clearly smiling or non-smiling.

D.2. FaRL Failures on Hair Color

The FaRL model by Zheng et al. (2022), as stated by the authors, is trained to "learn a universal facial representation that boosts all face analysis tasks." While the model was found to be a strong backbone for age estimation by Paplham & Franc (2023), we observed that FaRL is less reliable than a classifier trained on CelebA for detecting the smile and hair color attributes. When discriminating between the blond and black attributes on full sized CelebA images (218×178), FaRL achieves an accuracy of 0.89 on the validation set. We strictly followed the instructions on the official GitHub repo. On the other hand, our simple EfficientNet classifier trained on downscaled 64×64 images achieves an accuracy of 0.98. We observed similar results for the smile attribute. Thus, we only use FaRL for age estimation. Figure S10 display images where FaRL predicts a probability above 0.999 for the incorrect class. We do not believe those examples are particularly challenging.

[4]The URL will be provided upon de-anonymization of this work

Figure S10: Examples of images where FaRL fails blatantly. The first row depicts individuals with black hair, while the second row shows some with blond hair. For both rows, FaRL assigns a hair color likelihood < 0.01.

D.3. Rough Estimate of Errors of The Filtering Method

Recht et al. (2019) studies on ImageNet (Russakovsky et al., 2015) that when transferring from the train distribution to an unseen test set, an accuracy drop of 3% to 15% is expected. Note that CelebA is a simpler dataset than ImageNet since it contains only faces while ImageNet contains diverse classes going from specific breed of dogs to drumsticks. Let's assume the worst-case, i.e. that each classifier in the ensemble is accurate only 85% of the time only. A sample with mild attributes remains in $S^\star$ only if each of the five models confidently classifies it as having clear attributes. Assuming reasonable independence among classifiers, given that models are not trained on the same data, one can approximate the error probability for a single sample as $(15/100)^5$. We approximate the probability that at least k samples are incorrectly included in the dataset, $P(X \geq k)$, using the Chernoff bound (Molloy & Reed, 2002). This approximation is reasonable with our parameters ($p = (15/100)^5$, $n = 60$k, number of samples kept, $k > np$, number of errors). We find that $P(X \geq 10) \approx 0.089$, $P(X \geq 15) \approx 5e-4$ and $P(X \geq 20) \approx 7.2e-7$.

E. Extrapolation on the Lips Color Attribute

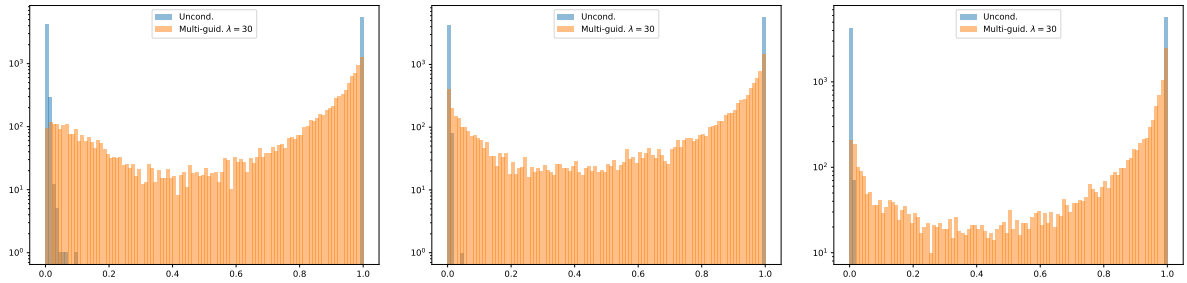
While investigating interpolation, we conducted preliminary experiments on extrapolation using synthetic data for training the guidance classifier. Specifically, we generated synthetic data utilizing dlib[5] to detect lips in CelebA pictures and overlay a green polygon on them. Intriguingly, using the "green lips" classifier enables the DDPM to generate images with green lips, as depicted in Figure S14a. Although the green patch may not always be perfectly centered on the lips, the unconditional diffusion model never encountered faces with green lips, hence the process is solely driven by the classifier.

F. Additional Material on Interpolation on Two Variables

We explore interpolation involving two variables. To create the training dataset, we retain extremes samples for both smile and hair color attributes. As such, we simply retain the intersection of extreme examples from experiments on smile and hair color, resulting in approximately $11,000$ examples. The heatmap visualizations in Figure 9 illustrate samples categorized by hair color and smile evaluation classifiers. We train the guidance classifier on 4 classes, corresponding to the corners of the heatmap. With a guidance parameter set to $\lambda = 30$ across all classes, we observe interpolation abilities of the model.

G. Additional Samples and Histograms

In this section, we present additional samples and histograms that could not fit in the main body for space reasons.



(a) (b) (c)

Figure S11: Multi-guidance performance when reducing the training data size. **(a)**: With 30k training samples. **(b)**: With 10k training samples. **(c)**: With 5k training samples. While using 30k or 10k maintains similar performance, the interpolation capability starts to degrade with 5k, even though DDPM retains its ability to interpolate.

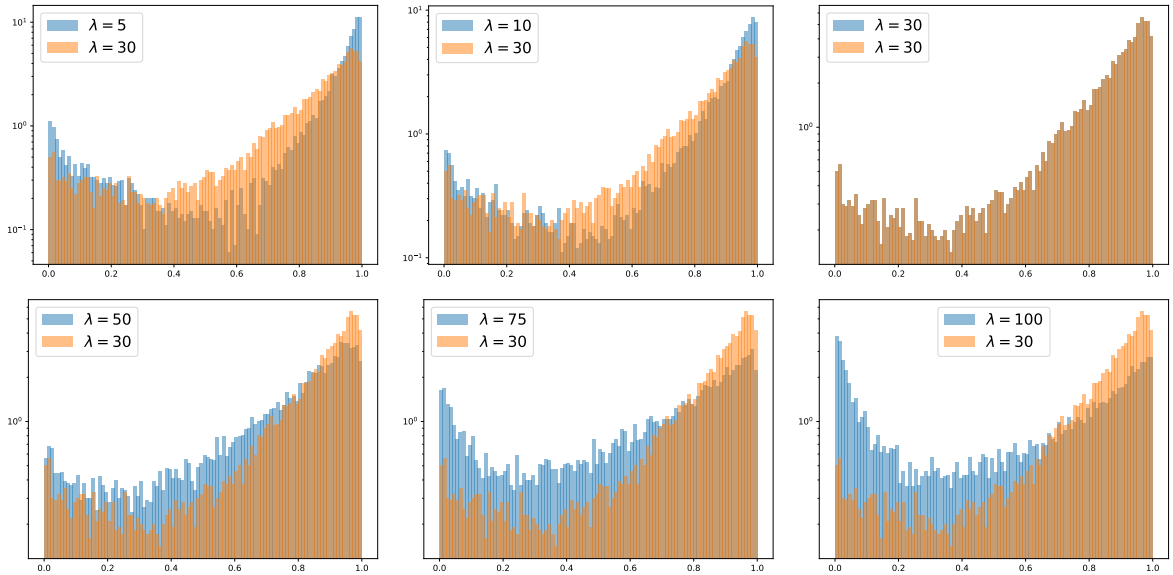


Figure S12: Histograms of the smile attribute when varying the guidance parameter λ.

G.1. Ablation on the Training Data Size

We present histograms of the empirical distributions according to the evaluation classifier on the smile attribute in Figure S11. As we saw in Section 5.2, the KLD and MSE is similar when training on 30k or 10k examples, and deteriorates substantially when training on 5000 samples only. Naturally, the best performance in terms of KLD and MSE is achieved with 60k examples, as the DDPM is exposed to more diversity.

G.2. Ablation on the Classifier Guidance Strength Parameter

In Figure S13, we observe that increasing the guidance strength almost monotonically improve the interpolation performance according to the MSE and KLD. In Figure S12, we display the empirical distribution of samples according to the smile attribute, using models trained on 60k examples. [t]

[5] `http://dlib.net`

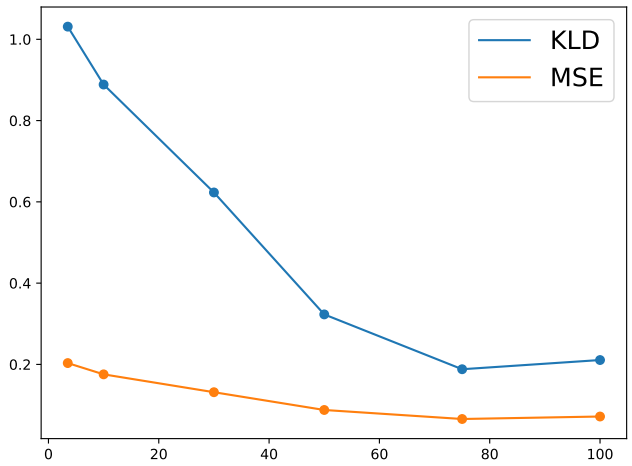


Figure S13: KLD and MSE between the empirical distribution (histograms in Figure S12) and the uniform distribution. The best performance is achieved for $\lambda = 75$.

Table S2: FID, Precision (P) and Recall (R) for different guidance parameters. The columns with "full" contain the metric computed between multi-guidance samples and $50k$ random samples from the training data. The columns with "mild" contain the metric computed between multi-guidance samples and the $10k$ training samples with predicted smile attribute closest to 0.5 in ℓ_1 distance.

Guid. strength	FID (mild)	FID (full)	P (mild)	P (full)	R (mild)	R (full)
0.0	**12.7215**	**8.1421**	0.7564	0.7876	**0.4755**	0.484
3.5	13.8471	10.17	0.8064	0.8164	0.4143	0.4190
10	14.4841	10.72	**0.8182**	**0.8195**	0.4135	0.4148
30	14.9275	11.34	0.8018	0.8051	0.4353	0.4250
50	14.6142	12.06	0.7756	0.7797	0.4365	0.4275
75	14.7289	13.11	0.7349	0.7317	0.4328	0.4225
100	15.1242	13.58	0.6549	0.6623	0.4713	**0.5383**

G.3. Additional Samples

In this section, we present additional samples that could not fit in the main body. Specifically, in Figure S14a, we showcase samples with **green lips**, from our preliminary extrapolation experiment. In Figure S14b, we display examples with **mild smiles**. In Figure S14c, we present interpolated samples between **young and old** individuals. Finally, in Figure S14d we showcase examples when interpolating between the **blond and black hair** colors.

H. Checking for Memorization

As memorization has been observed in diffusion models (Somepalli et al., 2022), we rigorously examine whether our DDPMs memorize the training data, especially when reducing the size of the training set. In the figures below, we present synthetic examples with their nearest neighbor based on CLIP embeddings (Radford et al., 2021). While models trained on 5k images exhibit memorization, increasing the number of images mitigates this effect. Refer to Figures S15, S16, S17, S18 for the examples with nearest neighbors in the training data for different data sizes.

I. Measuring Sample Quality and Diversity

We evaluate synthetic image quality and diversity using the FID (Heusel et al., 2017) and Precision and Recall (Sajjadi et al., 2018) metrics. We compare synthetic images against two sets of real images: 50k randomly selected samples from the CelebA training data and 10k examples with continuous labels nearest to 0.5 under the ℓ_1 distance. These subsets allow us to assess how well multi-guidance sampling captures both the entire data distribution and the section containing mild

examples only. Results for different values of λ are presented in Table S2.

FID The trends in FID scores align with our manual inspection, showing increasingly frequent artifacts in samples generated with multi-guidance for larger values of λ. Note that our FID values are higher than those typically reported in state-of-the-art diffusion models. This discrepancy can be attributed to our shorter training runs for computational efficiency. Our primary goal is to demonstrate interpolation behavior rather than to surpass the state-of-the-art in diffusion models.

Precision and Recall We observe that Precision and Recall correlates less with our manual assessment. Indeed, multi-guidance sampling captures mild smiles better than unconditional sampling, leading us to expect higher recall for multi-guidance, particularly when comparing synthetic samples with the most mild ones. Regarding Precision, we were surprised to find that computing precision for either the entire distribution or just the mild part resulted in similar trends. These findings suggest that the embedding model used to compute the Precision and Recall, the Inception network of Szegedy et al. (2015), might struggle to capture the nuances of our problem. Throughout the paper, we primarily use $\lambda = 30$ for guidance, as we found it to introduces minimal artifacts while effectively generating mild attribute examples.

J. Interpolation with Multi-Guidance under Additional Settings

We observe interpolation behavior in diffusion models by varying the architecture, as well as when sampling with classifier-free guidance (Ho & Salimans, 2022). Additionally, we attempted to train a DDPM using the $\mathcal{L}_{\text{VLB}}$ objective, but the model did not fit the training distribution well; in other words, unconditional sampling did not produce realistic-looking samples. Due to our limited computational budget, we did not extensively tune hyperparameters to successfully train a model using $\mathcal{L}_{\text{VLB}}$. Refer to Figure S19 for histograms.

Alternative architecture Instead of using the U-net architecture proposed by Nichol & Dhariwal (2021), we replaced it with the diffusion transformer introduced by Peebles & Xie (2023). As shown in Figure S19a, diffusion models implemented with a transformer also demonstrate interpolation capabilities. This suggests that the ability to interpolate does not depend on the choice of architecture.

Sampling with Classifier-Free Guidance Instead of classifier guidance (Dhariwal & Nichol, 2021), one can sample from a diffusion model using classifier-free guidance (Ho & Salimans, 2022). When adapting multi-guidance to classifier-free guidance, we still observe interpolation capabilities of the diffusion model. Interestingly, the shape of the distribution when sampling with classifier-free guidance differs from samples generated with classifier guidance. Specifically, there is little-to-no mass on the edges of the histogram. See Figure S19b for the histogram. We adapt the classifier-free sampling algorithm by using the following score for sampling:

$$s_{\text{cfg}}(x) = -\lambda s_\theta(x, \varnothing) + (1-\lambda)\frac{1}{2}(s_\theta(x, 0) + s_\theta(x, 1)) \tag{12}$$

where $s_\theta(x, i)$ is the score learned by the model for the class i, and $s_\theta(x, \varnothing)$ denotes the unconditional score.

Noise schedule and diffusion variance We observed that interpolation occurs with both the cosine and linear noise schedules from (Nichol & Dhariwal, 2021), as well as with fixed and learned variance of the diffusion process.

K. Demonstrating Interpolation on Objects of Different Sizes

In addition to the experiments detailed in the main body, we conducted an experiment to measure interpolation using a dataset of real-world objects resized to various dimensions. In this experiment, we define extreme examples as pictures with objects of the 20% smallest and largest sizes. Examples from the training data as well as histograms are displayed in Figure S20. Since we know the ground-truth object size in this experiment, we train the evaluation classifier to regress on the continuous attribute in the range $[0, 1]$ instead of training it on discretized labels.

L. Multi-Guidance on ImageNet

Sampling with multi-guidance from a model trained on ImageNet using the scores for the classes "Jay" and "Hen" produces images of blue chickens. Examples can be viewed in Figure S21. This result is interesting, but we caution that it is not

immediately clear that there exists a continuous range from "Hen" to "Jay". Nonetheless, we decided to share this since reviewers were curious about experiments on larger datasets such as ImageNet.

M. Interpolation between DDIM Latents

Building upon prior work on generative models such as Radford et al. (2016); Upchurch et al. (2017); Berthelot et al. (2018); Kingma & Dhariwal (2018); Karras et al. (2019); Härkönen et al. (2020), we study interpolation in the noisy latent space of diffusion models trained on extreme examples only. As for the rest of this paper, this investigation is crucial as prior research either trains models on the whole data distribution or relies on embeddings from multi-modal models trained on internet-scale datasets (Song et al., 2022; Wang & Golland, 2023). Similar to DDIM (Song et al., 2022), we define the latent space of diffusion models as the partially denoised samples, i.e. samples at non-zero noise levels. Interpolation within the latent space is achieved through a multi-step process: we first compute the latent representations of two real images after injecting noise for 2000 steps, then perform spherical linear interpolation (slerp), and finally, apply deterministic denoising using DDIM without noise injection. Although we experimented with regular linear interpolation, we found that slerp consistently achieves superior results. Figure S22 displays interpolation results between pairs of samples.

N. Attribute Editing without Training on Mild Examples

Following discussions with reviewers during the rebuttal phase, we conducted experiments on attribute editing using diffusion models trained exclusively on extreme examples. While previous work has shown successful interpolation in unconstrained settings (Kim et al., 2022; Kong et al., 2022; Gandikota et al., 2023; Yue et al., 2024), we demonstrate that even when trained solely on extreme examples, diffusion models can still exhibit interpolation capabilities. Specifically, we use the same DDPM model from in the interpolation experiments detailed in Section 5.1. We perform attribute editing by first mapping the original image to DDPM latents using 2000 reversing steps (half the number used during training). Subsequently, we denoise using regular classifier guidance with a guidance strength of 30. We use DDIM with noise injection using a linearly decaying schedule $(\eta_t)_{t=1}^{2000}$ such that $\eta_t = \frac{t}{2000}$. For further details on η_t, please refer to the original DDIM paper (Song et al., 2022). Using this straightforward approach, we observed that DDPMs could transform non-smiling faces into mildly smiling ones. CelebA images with their "smile" attribute edited can be seen in Figure S23.

(a)

(b)

(c)

(d)

Figure S14: Additional *synthesized* samples. **(a)**: Extrapolation with classifier trained on synthetic data. **(b)**: Interpolation on smile attribute. **(c)**: Interpolation on age attribute. **(d)**: Interpolation on hair color.

Figure S15: Checking for memorization when training on 60k extreme examples from CelebA. The generated samples are on the left, while the right side showcases the nearest neighbors according to CLIP embeddings

Figure S16: Checking for memorization when training on 30k extreme examples from CelebA. The generated samples are on the left, while the right side showcases the nearest neighbors according to CLIP embeddings.

Figure S17: Checking for memorization when training on 10k extreme examples from CelebA. The generated samples are on the left, while the right side showcases the nearest neighbors according to CLIP embeddings.

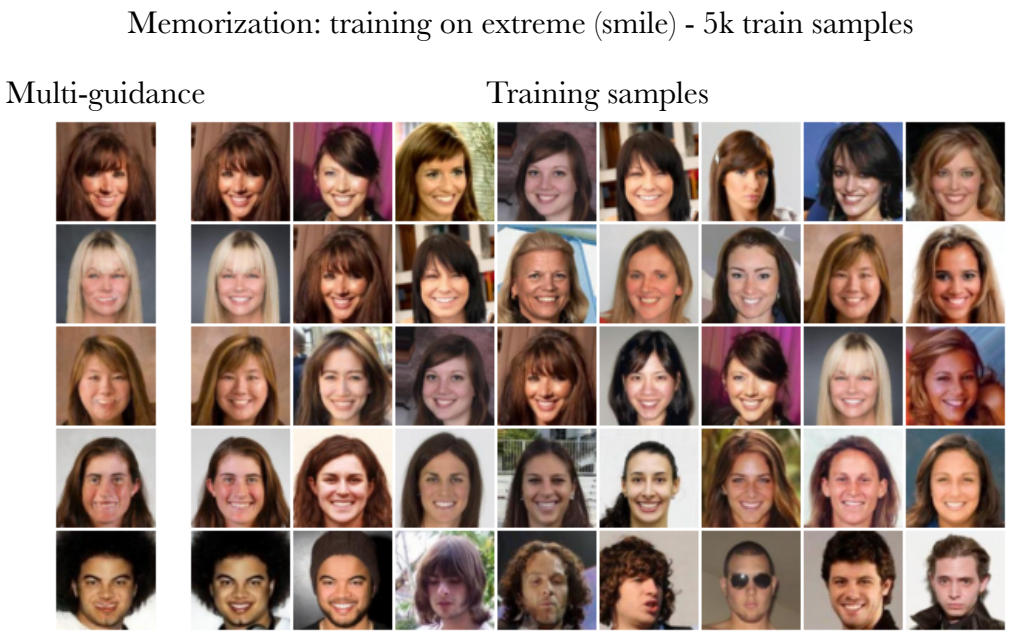
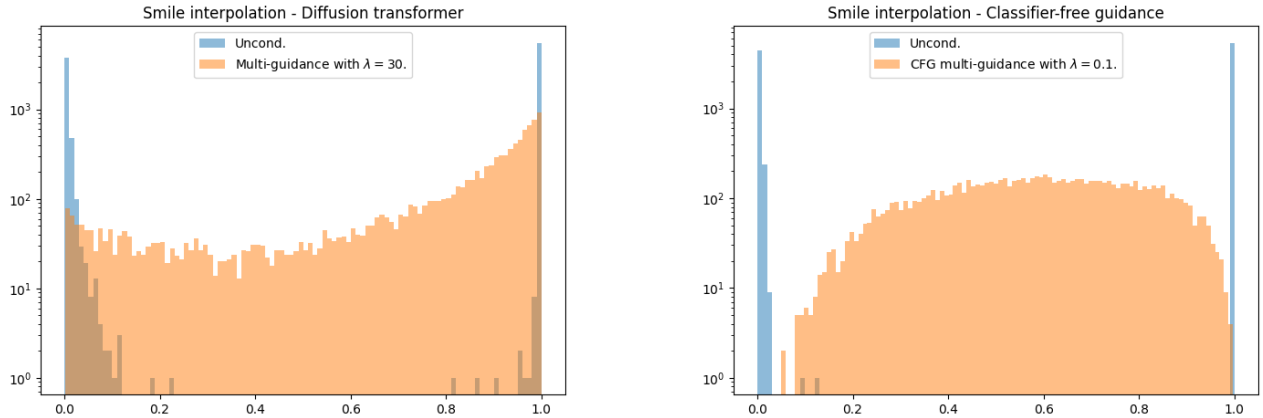
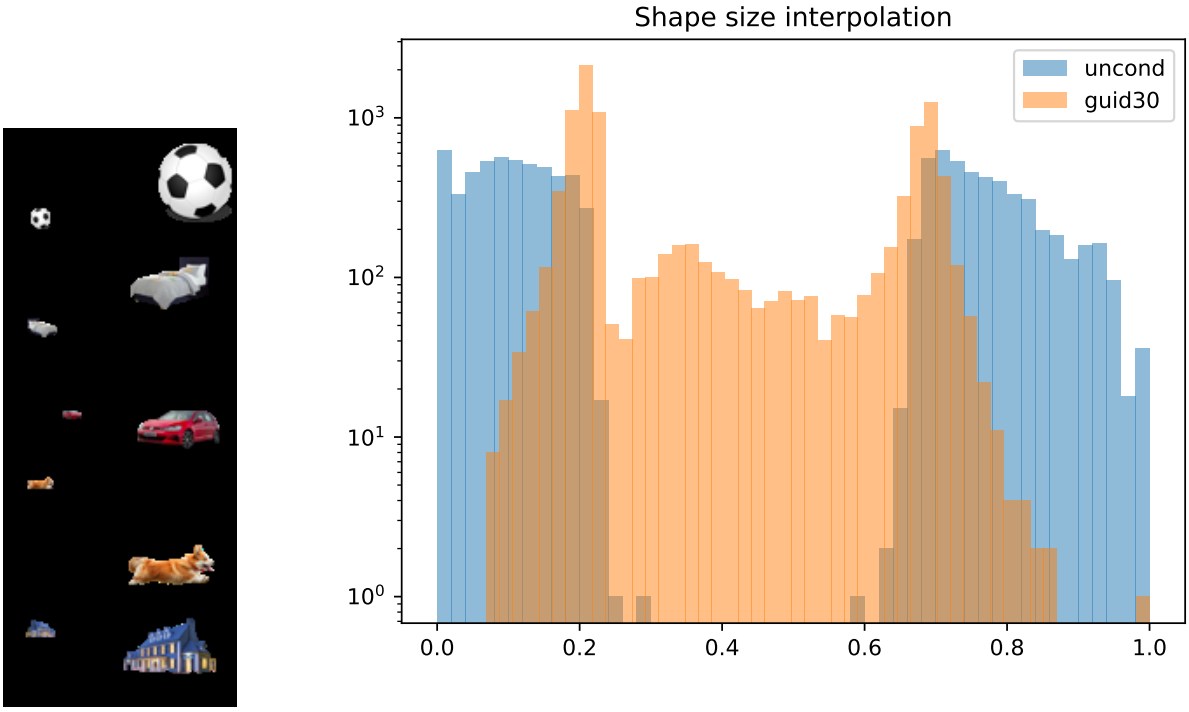


Figure S18: Checking for memorization when training on 5k extreme examples from CelebA. The generated samples are on the left, while the right side showcases the nearest neighbors according to CLIP embeddings.



(a) (b)

Figure S19: Distribution histogram of smile sampled with (a): a diffusion transformer instead of a U-net; (b): when sampling with classifier-free guidance.



(a) (b)

Figure S20: Left: training examples with rescaled real-world objects. Right: distribution of object sizes when sampling from the unconditional model or with multi-guidance.

Figure S21: Samples generated with multi-guidance between the classes "Hen" and "Jay". While we believe that interpolation between thoses classes is not well-defined, we include this result as it may interest the reader.

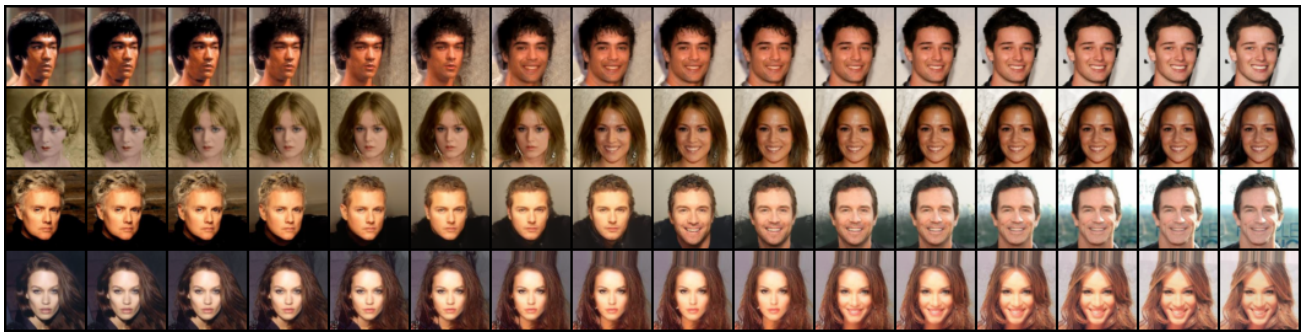


Figure S22: Spherical linear interpolation (slerp) between ddim latents. We see that intermediate regions of the latent space get decoded as mild smiles by the diffusion model trained solely on extreme examples.

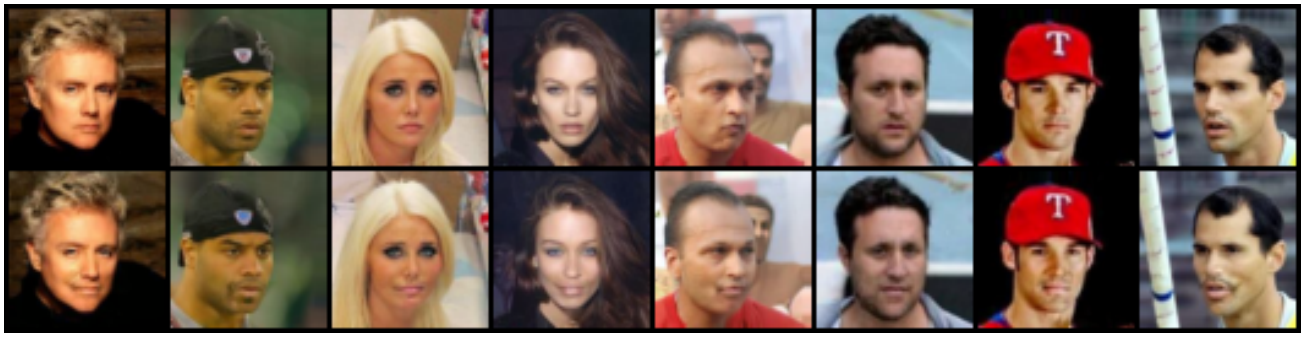


Figure S23: Attribute edition using classifier guidance. Given a real image, we inject noise and decoding with regular classifier guidance yields images with mild attributes.