# D2C: Diffusion-Decoding Models for Few-Shot Conditional Generation

**Anonymous Author(s)**
Affiliation
Address
email

## Abstract

Conditional generative models of high-dimensional images have many applications, but supervision signals from conditions to images can be expensive to acquire. This paper describes Diffusion-Decoding models with Contrastive representations (D2C), a paradigm for training unconditional variational autoencoders (VAEs) for few-shot conditional image generation. D2C uses a learned diffusion-based prior over the latent representations to improve generation and contrastive self-supervised learning to improve representation quality. D2C can adapt to novel generation tasks conditioned on labels or manipulation constraints, by learning from as few as 100 labeled examples. On conditional generation from new labels, D2C achieves superior performance over state-of-the-art VAEs and diffusion models. On conditional image manipulation, D2C generations are two orders of magnitude faster to produce over StyleGAN2 ones and are preferred by $50\% - 60\%$ of the human evaluators in a double-blind study.

## 1 Introduction

Generative models trained on large amounts of unlabeled data have achieved great success in various domains including images [8, 47, 72, 40], text [53, 2], audio [24, 68, 88, 59], and graphs [34, 64]. However, downstream applications of generative models are often based on various conditioning signals, such as labels [58], text descriptions [57], reward values [96], or similarity with existing data [43]. While it is possible to directly train conditional models, this often requires large amounts of paired data [54, 71] that are costly to acquire. Hence, it would be desirable to learn conditional generative models using large amounts of unlabeled data and as little paired data as possible.

Contrastive self-supervised learning (SSL) methods can greatly reduce the need for labeled data in discriminative tasks by learning effective representations from unlabeled data [90, 35, 33], and have also been shown to improve few-shot learning [37]. It is therefore natural to ask if they can also be used to improve few-shot generation. Latent variable generative models (LVGM) are a natural candidate for this, since they already involve a low-dimensional, structured latent representation of the data they generate. However, popular LVGMs, such as generative adversarial networks (GANs, [32, 47]) and diffusion models [40, 80], lack explicit tractable functions to map inputs to representations, making it difficult to optimize latent variables with SSL. Variational autoencoders (VAEs, [49, 74]), on the other hand, can naturally adopt SSL through their encoder model, but they typically have worse sample quality.

Submitted to 35th Conference on Neural Information Processing Systems (NeurIPS 2021). Do not distribute.

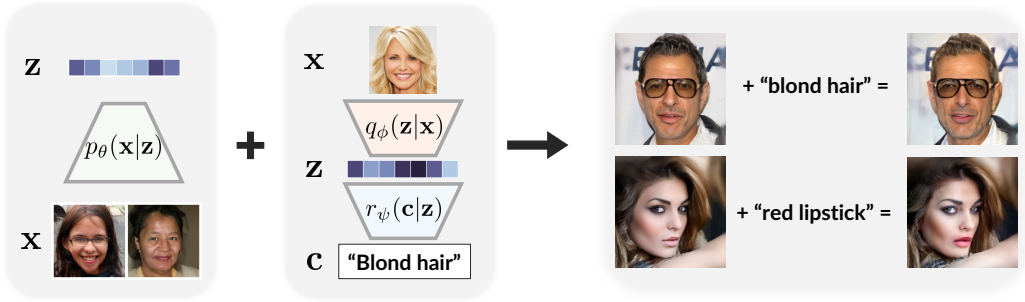



Figure 1: Few-shot conditional generation with the unconditional D2C model (left). With a recognition model over the latent space (middle), D2C can generate samples for novel conditions, such as image manipulation (right). These conditions can be defined with very few labels.

In this paper, we propose Diffusion-Decoding models with Contrastive representations (D2C), a special VAE that is suitable for conditional few-shot generation. D2C uses contrastive self-supervised learning methods to obtain a latent space that inherits the transferrability and few-shot capabilities of self-supervised representations. Unlike other VAEs, D2C learns a diffusion model over the latent representations. This latent diffusion model ensures that D2C uses the same latent distribution for both training and generation. We provide a formal argument to explain why this approach may lead to better sample quality than existing hierarchical VAEs. We further discuss how to apply D2C to few-shot conditional generation where the conditions are defined through labeled examples and/or manipulation constraints. Our approach combines a discriminative model providing conditioning signal and generative diffusion model over the latent space, and is computationally more efficient than methods that act directly over the image space (Figure 1).

We evaluate and compare D2C with several state-of-the-art generative models over 6 datasets. On unconditional generation, D2C outperforms state-of-the-art VAEs and is competitive with diffusion models under similar computational budgets. On conditional generation with 100 labeled examples, D2C significantly outperforms state-of-the-art VAE [87] and diffusion models [80]. D2C can also learn to perform certain image manipulation tasks from as few as 100 labeled examples. Notably, for manipulating images, D2C is two orders of magnitude faster than StyleGAN2 [101] and preferred by $50\% - 60\%$ of human evaluations, which to our best knowledge is the first for any VAE model.

## 2 Background

**Latent variable generative models** A latent variable generative model (LVGM) is posed as a conditional distribution $p_\theta : \mathcal{Z} \to \mathcal{P}(\mathcal{X})$ from a latent variable $\mathbf{z}$ to a generated sample $\mathbf{x}$, parametrized by $\theta$. To acquire new samples, LVGMs draw random latent variables $\mathbf{z}$ from some distribution $p(\mathbf{z})$ and map them to image samples through $\mathbf{x} \sim p_\theta(\mathbf{x}|\mathbf{z})$. Most LVGMs are built on top of three paradigms: variational autoencoders (VAEs, [49, 74]), Normalizing Flows (NFs, [26, 27]), Generative Adversarial Networks (GANs, [32]), and diffusion / score-based generative models [40, 81].

Particularly, VAEs use an inference model from $\mathbf{x}$ to $\mathbf{z}$ for training. Denoting the inference distribution from $\mathbf{x}$ to $\mathbf{z}$ as $q_\phi(\mathbf{z}|\mathbf{x})$, the generative distribution from $\mathbf{z}$ to $\mathbf{x}$ as $p_\theta(\mathbf{x}|\mathbf{z})$, VAEs are trained by minimizing the following upper bound of negative log-likelihood:

$$L_{\text{VAE}} = \mathbb{E}_{\mathbf{x}\sim p_{\text{data}}}[\mathbb{E}_{\mathbf{z}\sim q_\phi(\mathbf{z}|\mathbf{x})}[-\log p(\mathbf{x}|\mathbf{z})] + D_{\text{KL}}(q_\phi(\mathbf{z}|\mathbf{x})\|p(\mathbf{z}))] \tag{1}$$

where $p_{\text{data}}$ is the data distribution and $D_{\text{KL}}$ is the KL-divergence.

**Diffusion models** Diffusion models [78, 40, 80] produce samples by reversing a Gaussian diffusion process. We use the index $\alpha \in [0, 1]$ to denote the particular noise level of an noisy observation $\mathbf{x}^{(\alpha)} = \sqrt{\alpha}\mathbf{x} + \sqrt{1-\alpha}\epsilon$, where $\mathbf{x}$ is the clean observation and $\epsilon \sim \mathcal{N}(0, \boldsymbol{I})$ is a standard Gaussian distribution; as $\alpha \to 0$, the distribution of $\mathbf{x}^{(\alpha)}$ converges to $\mathcal{N}(0, \boldsymbol{I})$. Diffusion models are typically

parametrized as reverse noise models $\epsilon_\theta(\mathbf{x}^{(\alpha)}, \alpha)$ that predict the noise component of $\mathbf{x}^{(\alpha)}$ given a noise level $\alpha$, and trained to minimize $\|\epsilon_\theta(\mathbf{x}^{(\alpha)}, \alpha) - \epsilon\|_2^2$, the mean squared error loss between the true noise and predicted noise. Given any non-decreasing series $\{\alpha_i\}_{i=0}^T$ between 0 and 1, the diffusion objective for a clean sample from the data $\mathbf{x}$ is:

$$\ell_{\text{diff}}(\mathbf{x}; w, \theta) := \sum_{i=0}^{T} w(\alpha_i) \mathbb{E}_{\epsilon \sim \mathcal{N}(0, \boldsymbol{I})}[\|\epsilon - \epsilon_\theta(\mathbf{x}^{(\alpha_i)}, \alpha_i)\|_2^2], \quad \mathbf{x}^{(\alpha_i)} := \sqrt{\alpha_i}\mathbf{x} + \sqrt{1-\alpha_i}\epsilon \tag{2}$$

where $w : \{\alpha_i\}_{i=0}^T \to \mathbb{R}_+$ controls the loss weights for each $\alpha$. When $w(\alpha) = 1$ for all $\alpha$, we recover the denoising score matching objective for training score-based generative models [81].

Given an initial sample $\mathbf{x}_0 \sim \mathcal{N}(0, \boldsymbol{I})$, diffusion models acquires clean samples (*i.e.*, samples of $\mathbf{x}_1$) through a gradual denoising process, where samples with reducing noise levels $\alpha$ are produced (*e.g.*, $\mathbf{x}_0 \to \mathbf{x}_{0.3} \to \mathbf{x}_{0.7} \to \mathbf{x}_1$). In particular, Denoising Diffusion Implicit Models (DDIMs, [80]) uses an Euler discretization of some neural ODE [13] to produce samples (Figure 2, left).

We provide a more detailed description for training diffusion models in Appendix A.1 and sampling from DDIM in Appendix A.2. For conciseness, we use the notation $p^{(\alpha)}(\mathbf{x}^{(\alpha)})$ to denote the marginal distribution of $\mathbf{x}^{(\alpha)}$ under the diffusion model, and $p^{(\alpha_1,\alpha_2)}(\mathbf{x}^{(\alpha_2)} \mid \mathbf{x}^{(\alpha_1)})$ to denote the diffusion sampling process from $\mathbf{x}^{(\alpha_1)}$ to $\mathbf{x}^{(\alpha_2)}$ (assuming $\alpha_1 < \alpha_2$). This notation abstracts away the exact sampling procedure of the diffusion model, which depends on choices of $\alpha$.

**Self-supervised learning of representations** In self-supervised learning (SSL), representations are learned by completing certain pretext tasks that do not require extra manual labeling [65, 23]; these representations can then be applied to other downstream tasks, often in few-shot or zero-shot scenarios. In particular, contrastive representation learning encourages representations to be closer between "positive" pairs and further between "negative" pairs; contrastive predictive coding (CPC, [90]), based on multi-class classification, have been commonly used in state-of-the-art methods [35, 15, 17, 14, 79]. Other non-contrastive methods exist, such as BYOL [33] and SimSiam [16], but they usually require additional care to prevent the representation network from collapsing.

## 3 Problem Statement

**Few-shot conditional generation** Our goal is to learn an unconditional generative model $p_\theta(\mathbf{x})$ such that it is suitable for conditional generation. Let $\mathcal{C}(\mathbf{x}, \mathbf{c}, f)$ describe an event that "$f(\mathbf{x}) = \mathbf{c}$", where $\mathbf{c}$ is a property value and $f(\mathbf{x})$ is a property function that is unknown at training. In conditional generation, our goal is to sample $\mathbf{x}$ such that the event $\mathcal{C}(\mathbf{x}, \mathbf{c}, f)$ occurs for a chosen $\mathbf{c}$. If we have access to some "ground-truth" model that gives us $p(\mathcal{C}|\mathbf{x}) := p(f(\mathbf{x}) = \mathbf{c}|\mathbf{x})$, then the conditional model can be derived from Bayes' rule: $p_\theta(\mathbf{x}|\mathcal{C}) \propto p(\mathcal{C}|\mathbf{x})p_\theta(\mathbf{x})$. These properties $\mathbf{c}$ include (but are not limited to[1]) labels [58], text descriptions [57, 73], noisy or partial observations [11, 5, 44, 22], and manipulation constraints [66]. In many cases, we do not have direct access to the true $f(\mathbf{x})$, so we need to learn an accurate model from labeled data [6] (*e.g.*, $(\mathbf{c}, \mathbf{x})$ pairs).

**Desiderata** Many existing methods are optimized for some known condition (*e.g.*, labels in conditional GANs [8]) or assume abundant pairs between images and conditions that can be used for pretraining (*e.g.*, DALL-E [71] and CLIP [70] over image-text pairs). Neither is the case in this paper, as we do not expect to train over paired data.

While high-quality latent representations are not essential to unconditional image generation (*e.g.*, autoregressive [89], energy-based [29], and some diffusion models [40]), they can be beneficial when we wish to specify certain conditions with limited supervision signals, similar to how SSL representations can reduce labeling efforts in downstream tasks. A compelling use case is detecting and removing biases in datasets via image manipulation, where we should not only address well-known biases a-priori but also address other hard-to-anticipate biases, adapting to societal needs [62].

Therefore, a desirable generative model should not only have high sample quality but also contain informative latent representations. While VAEs are ideal for learning rich latent representations due to being able to incorporate SSL within the encoder, they generally do not achieve the same level of sample quality as GANs and diffusion models.

[1] When $\mathcal{C}$ refers to an event that is always true, we recover unconditioned generation.

Table 1: A comparison of several common paradigms for generative modeling. [Explicit $\mathbf{x} \to \mathbf{z}$]: the mapping from $\mathbf{x}$ to $\mathbf{z}$ is directly trainable, which enables SSL; [No prior hole]: latent distributions used for generation and training are identical (Sec. 4.2), which improves generation; [Non-adversarial]: training procedure does not involve adversarial optimization, which improves training stability.

| **Model family** | Explicit $\mathbf{x} \to \mathbf{z}$ (Enables SSL) | No prior hole (Better generation) | Non-Adversarial (Stable training) |
|---|---|---|---|
| VAE [49, 74], NF [26] | ✔ | ✗ | ✔ |
| GAN [32] | ✗ | ✔ | ✗ |
| BiGAN [28, 30] | ✔ | ✔ | ✗ |
| DDIM [80] | ✗ | ✔ | ✔ |
| **D2C** | ✔ | ✔ | ✔ |

# 4 Diffusion-Decoding Generative Models with Contrastive Learning

To address the above issue, we present Diffusion-Decoding generative models with Contrastive Learning (D2C), an extension to VAEs with high-quality samples and high-quality latent representations, and are thus well suited to few-shot conditional generation. Moreover, unlike GAN-based methods, D2C does not involve unstable adversarial training (Table 1).

As its name suggests, the generative model for D2C has two components – *diffusion* and *decoding*; the *diffusion* component operates over the latent space and the *decoding* component maps from latent representations to images. Let us use the $\alpha$ index notation for diffusion random variables: $\mathbf{z}^{(0)} \sim p^{(0)}(\mathbf{z}^{(0)}) := \mathcal{N}(0, \boldsymbol{I})$ is the "noisy" latent variable with $\alpha = 0$, and $\mathbf{z}^{(1)}$ is the "clean" latent variable with $\alpha = 1$. The generative process of D2C, which we denote $p_\theta(\mathbf{x}|\mathbf{z}^{(0)})$, is then defined as:

$$\mathbf{z}^{(0)} \sim p^{(0)}(\mathbf{z}^{(0)}), \quad \mathbf{z}^{(1)} \sim \underbrace{p_\theta^{(0,1)}(\mathbf{z}^{(1)}|\mathbf{z}^{(0)})}_{\text{diffusion}}, \quad \mathbf{x} \sim \underbrace{p_\theta(\mathbf{x}|\mathbf{z}^{(1)})}_{\text{decoding}}, \tag{3}$$

where $p^{(0)}(\mathbf{z}^{(0)}) = \mathcal{N}(0, \boldsymbol{I})$ is the prior distribution for the diffusion model, $p_\theta^{(0,1)}(\mathbf{z}^{(1)}|\mathbf{z}^{(0)})$ is the diffusion process from $\mathbf{z}^{(0)}$ to $\mathbf{z}^{(1)}$, and $p_\theta(\mathbf{x}|\mathbf{z}^{(1)})$ is the decoder from $\mathbf{z}^{(1)}$ to $\mathbf{x}$. Intuitively, D2C models produce samples by drawing $\mathbf{z}^{(1)}$ from a diffusion process and then decoding $\mathbf{x}$ from $\mathbf{z}^{(1)}$.

In order to train a D2C model, we use an inference model $q_\phi(\mathbf{z}^{(1)}|\mathbf{x})$ that predicts proper $\mathbf{z}^{(1)}$ latent variables from $\mathbf{x}$; this can directly incorporate SSL methods [94], leading to the following objective:

$$L_{\text{D2C}}(\theta, \phi; w) := L_{\text{D2}}(\theta, \phi; w) + \lambda L_{\text{C}}(q_\phi), \tag{4}$$

$$L_{\text{D2}}(\theta, \phi; w) := \mathbb{E}_{\mathbf{x} \sim p_{\text{data}}, \mathbf{z}^{(1)} \sim q_\phi(\mathbf{z}^{(1)}|\mathbf{x})}[-\log p(\mathbf{x}|\mathbf{z}^{(1)}) + \ell_{\text{diff}}(\mathbf{z}^{(1)}; w, \theta)], \tag{5}$$

where $\ell_{\text{diff}}$ is defined as in Eq.(2), $L_C(q_\phi)$ denotes any contrastive predictive coding objective [90] with rich data augmentations [35, 15, 17, 14, 79] (details in Appendix A.3) and $\lambda > 0$ is a weight hyperparameter. The first two terms, which we call $L_{\text{D2}}$, contains a "reconstruction loss" ($-\log p(\mathbf{x}|\mathbf{z}^{(1)})$) and a "diffusion loss" over samples of $\mathbf{z}^{(1)} \sim q_\phi(\mathbf{z}^{(1)}|\mathbf{x})$. We illustrate the D2C generative and inference models in Figure 2, and its training procedure in Appendix A.4.

## 4.1 Relationship to maximum likelihood

The D2 objective ($L_{\text{D2}}$) appears similar to the original VAE objective ($L_{\text{VAE}}$). Here, we make an informal statement that the D2 objective function is deeply connected to the variational lower bound of log-likelihood; we present the full statement and proof in Appendix B.1.

**Theorem 1** (informal)**.** *For any valid $\{\alpha_i\}_{i=0}^T$, there exists some weights $\hat{w} : \{\alpha_i\}_{i=0}^T \to \mathbb{R}_+$ for the diffusion objective such that $-L_{\text{D2}}$ is a variational lower bound to the log-likelihood, i.e.,*

$$-L_{\text{D2}}(\theta, \phi; \hat{w}) \leq \mathbb{E}_{p_{\text{data}}}[\log p_\theta(\mathbf{x})], \tag{6}$$

*where $p_\theta(\mathbf{x}) := \mathbb{E}_{\mathbf{x}_0 \sim p^{(0)}(\mathbf{z}^{(0)})}[p_\theta(\mathbf{x}|\mathbf{z}^{(0)})]$ is the marginal probability of $\mathbf{x}$ under the D2C model.*

*Proof.* (sketch) The diffusion term $\ell_{\text{diff}}$ upper bounds the KL divergence between $q_\phi(\mathbf{z}_1|\mathbf{x})$ and $p_\theta^{(1)}(\mathbf{z}^{(1)})$ for suitable weights [40, 80], which recovers a VAE objective. □

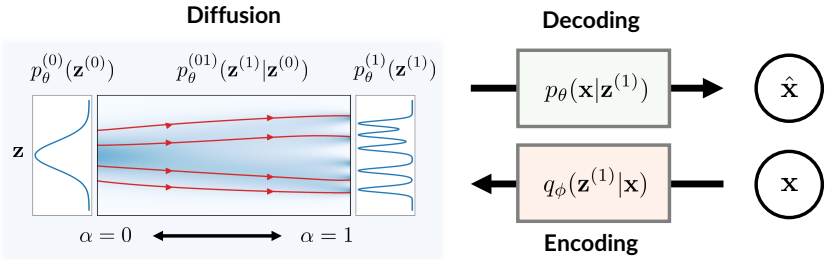



Figure 2: Illustration of components of a D2 model. On top of the encoding and decoding between $\mathbf{x}$ and $\mathbf{z}^{(1)}$, we use a diffusion model to generate $\mathbf{z}^{(1)}$ from a Gaussian $\mathbf{z}^{(0)}$. The red lines describe several smooth ODE trajectories from $\alpha = 0$ to $\alpha = 1$ corresponding to DDIM.

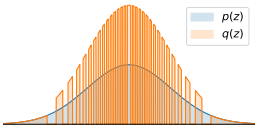
### 4.2 D2 models address latent posterior mismatch in VAEs

While D2C is a special case of VAE, we argue that D2C is non-trivial in the sense that it addresses a long-standing problem in VAE methods [86, 83], namely the mismatch between the prior distribution $p_\theta(\mathbf{z})$ and the aggregate (approximate) posterior distribution $q_\phi(\mathbf{z}) := \mathbb{E}_{p_{\text{data}}(\mathbf{x})}[q_\phi(\mathbf{z}|\mathbf{x})]$. A mismatch could create "holes" [76, 41, 3] in the prior that the aggregate posterior fails to cover during training, resulting in worse sample quality, as many latent variables used during generation are likely to never have been trained on. We formalize this notion in the following definition.

**Definition 1** (Prior hole)**.** *Let $p(\mathbf{z}), q(\mathbf{z})$ be two distributions with $\text{supp}(q) \subseteq \text{supp}(p)$. We say that $q$ has an $(\epsilon, \delta)$-**prior hole** with respect to (the prior) $p$ for $\epsilon, \delta \in (0, 1)$, $\delta > \epsilon$, if there exists a set $S \in \text{supp}(P)$, such that $\int_S p(\mathbf{z})\mathrm{d}\mathbf{z} \geq \delta$ and $\int_S q(\mathbf{z})\mathrm{d}\mathbf{z} \leq \epsilon$.*

Intuitively, if $q_\phi(\mathbf{z})$ has a prior hole with large $\delta$ and small $\epsilon$ (*e.g.*, inversely proportional to the number of training samples), then it is very likely that latent variables within the hole are never seen during training (small $\epsilon$), yet frequently used to produce samples (large $\delta$). Most existing methods address this problem by optimizing certain statistical divergences between $q_\phi(\mathbf{z})$ and $p_\theta(\mathbf{z})$, such as the KL divergence or Wasserstein distance [84]. However, we argue in the following statement that prior holes might not be eliminated even if we optimize certain divergence values to be reasonably low, especially when $q_\phi(\mathbf{z})$ is very flexible. We present the formal statement and proof in Appendix B.2.

**Theorem 2.** *(informal) Let $p_\theta(\mathbf{z}) = \mathcal{N}(0, 1)$. For any $\epsilon > 0$, there exists a distribution $q_\phi(\mathbf{z})$ with an $(\epsilon, 0.49)$-prior hole, such that $D_{\text{KL}}(q_\phi \| p_\theta) \leq \log 2$[2] and $W_2(q_\phi, p_\theta) < \gamma$ for any $\gamma > 0$, where $W_2$ is the 2-Wasserstein distance.*




*Proof.* (sketch) We construct a $q_\phi$ that satisfies these properties (top-right figure). First, we truncate the Gaussian and divide them into regions with same probability mass; then we support $q_\phi$ over half of these regions (so $\delta > 0.49$); finally, we show that the divergences are small enough. □

In contrast to addressing prior holes by optimization, diffusion models eliminate prior holes by construction, since the diffusion process from $\mathbf{z}^{(1)}$ to $\mathbf{z}^{(0)}$ is constructed such that the distribution of $\mathbf{z}^{(\alpha)}$ always converges to a standard Gaussian as $\alpha \to 0$. As a result, the distribution of latent variables used during training is arbitrarily close to that used in generation[3], which is also the case in GANs. Therefore, our argument provides an explanation as to why we observe better sample quality results from GANs and diffusion models than VAEs and NFs.

## 5 Few-shot Conditional Generation with D2C

In this section, we discuss how D2C can be used to learn to perform conditional generation from few-shot supervision. We note that D2C is only trained on images and not with any other data modalities (*e.g.*, image-text pairs [71]) or supervision techniques (*e.g.*, meta-learning [20, 6]).

[2] This is reasonably low for realistic VAE models (NVAE [87] reports a KL divergence of around 2810 nats).
[3] We expand this argument in Appendix B.2.

**Algorithm** We describe the general algorithm for conditional generation from a few images in Algorithm 1, and detailed implementations in Appendix C. With a model over the latent space (denoted as $r_\psi(\mathbf{c}|\mathbf{z}^{(1)})$), we draw conditional latents from an unnormalized distribution with the diffusion prior (line 4). This can be implemented in many ways such as rejection sampling or Langevin dynamics [63, 82, 25].

**Algorithm 1** Conditional generation with D2C

1: **Input** $n$ examples $\{(\mathbf{x}_i, \mathbf{c}_i)\}_{i=1}^n$, property $\mathbf{c}$.
2: Acquire latents $\mathbf{z}_i^{(1)} \sim q_\phi(\mathbf{z}^{(1)}|\mathbf{x})$ for $i \in [n]$;
3: Train model $r_\psi(\mathbf{c}|\mathbf{z}^{(1)})$ over $\{(\mathbf{z}_i^{(1)}, \mathbf{c}_i)\}_{i=1}^n$
4: Sample latents with $\hat{\mathbf{z}}^{(1)} \sim r_\psi(\mathbf{c}|\mathbf{z}^{(1)}) \cdot p_\theta^{(1)}(\mathbf{z}^{(1)})$ (unnormalized);
5: Decode $\hat{\mathbf{x}} \sim p_\theta(\mathbf{x}|\hat{\mathbf{z}}^{(1)})$.
6: **Output** $\hat{\mathbf{x}}$.

**Conditions from labeled examples** Given a few labeled examples, we wish to produce diverse samples with a certain label. For labeled examples we can directly train a classifier over the latent space, which we denote as $r_\psi(\mathbf{c}|\mathbf{z}^{(1)})$ with $\mathbf{c}$ being the class label and $\mathbf{z}^{(1)}$ being the latent representation of $\mathbf{x}$ from $q_\phi(\mathbf{z}^{(1)}|\mathbf{x})$. If these examples do not have labels (*i.e.*, we merely want to generate new samples similar to given ones), we can train a positive-unlabeled (PU) classifier [31] where we assign "positive" to the new examples and "unlabeled" to training data. Then we use the classifier with the diffusion model $p_\theta(\mathbf{z}^{(1)}|\mathbf{z}^{(0)})$ to produce suitable values of $\mathbf{z}^{(1)}$, such as by rejecting samples from the diffusion model that has a small $r_\psi(\mathbf{c}|\mathbf{z}^{(1)})$.

**Conditions from manipulation constraints** Given a few labeled examples, here we wish to learn how to manipulate images. Specifically, we condition over the event that "$\mathbf{x}$ has label $\mathbf{c}$ but is similar to image $\bar{\mathbf{x}}$". Here $r_\psi(\mathbf{c}|\mathbf{z}^{(1)})$ is the unnormalized product between the classifier conditional probability and closeness to the latent $\bar{\mathbf{z}}^{(1)}$ of $\bar{\mathbf{x}}$ (*e.g.*, measured with RBF kernel). We implement line 4 of Alg. 1 with a Lanvegin-like procedure where we take a gradient step with respect to the classifier probability and then correct this gradient step with the diffusion model. Unlike many GAN-based methods [12, 69, 92, 43, 93], D2C does not need to optimize an inversion procedure at evaluation time, and thus the latent value is much faster to compute; D2C is also better at retaining fine-grained features of the original image due to the reconstruction loss.

## 6 Related Work

**Latent variable generative models** Most deep generative models explicitly define a latent representation, except for some energy-based models [39, 29] and autoregressive models [89, 88, 10]. Unlike VAEs and NFs, GANs do not explicitly define an inference model and instead optimize a two-player game. In terms of sample quality, GANs currently achieve superior performance over VAEs and NFs, but they can be difficult to invert even with additional optimization [45, 95, 7]. This can be partially addressed by training reconstruction-based losses with GANs [51, 52]. Moreover, the GAN training procedure can be unstable [9, 8, 60], lack a informative objective for measuring progress [4], and struggle with discrete data [97]. Diffusion models [25] achieves high sample quality without adversarial training, but its latent dimension must be equal to the image dimension.

**Addressing posterior mismatch in VAEs** Most methods address this mismatch problem by improving inference models [61, 48, 85], prior models [86, 3, 83], or objective functions [98, 99, 100, 1, 56]; all these approaches optimize the posterior model to be close to the prior. In Section 4.2, we explain why these approaches do not necessarily remove large "prior holes", so their sample qualities remain relatively poor even after many layers [87, 18]. Other methods adopt a "two-stage" approach [21], which fits a generative model over the latent space of autoencoders [91, 72, 24, 71].

**Conditional generation with unconditional models** To perform conditional generation over an unconditional LVGM, most methods assume access to a discriminative model (*e.g.*, a classifier); the latent space of the generator is then modified to change the outputs of the discriminative model. The disciminative model can operate on either the image space [63, 67, 25] or the latent space [77, 93]. For image space discriminative models, plug-and-play generative networks [63] control the attributes of generated images via Langevin dynamics [75]; these ideas are also explored in diffusion models [82]. Image manipulation methods are based on GANs often operate with latent space discriminators [77, 93]. However, these methods have some trouble manipulating real images because of imperfect reconstruction [102, 7]. This is not a problem in D2C since a reconstruction objective is optimized.

Figure 3: Generated samples on CIFAR-10 (left), fMoW (mid), and FFHQ $256 \times 256$ (right).

# 7 Experiments

We examine the conditional and unconditional generation qualities of D2C over CIFAR-10 [50], CIFAR-100 [50], fMoW [19], CelebA-64 [55], CelebA-HQ-256 [45], and FFHQ-256 [46]. Our D2C implementation is based on the state-of-the-art NVAE [87] autoencoder structure, the U-Net diffusion model [40], and the MoCo-v2 contrastive representation learning method [15]. We keep the diffusion series hyperparameter $\{\alpha_i\}_{i=1}^T$ identical to ensure a fair comparison with different diffusion models. For the contrastive weight hyperparameter $\lambda$ in Equation (4), we consider the value of $\lambda = 10^{-4}$ based on the relative scale between the $L_{\mathrm{C}}$ and $L_{\mathrm{D2}}$; we find that the results are relatively insensitive to $\lambda$. We use 100 diffusion steps for DDIM and D2C unless mentioned otherwise, as running with longer steps is not computationally economical despite tiny gains in FID [80]. We include additional training details, such as architectures, optimizers and learning rates in Appendix C.

Table 2: Quality of representations and generations with LVGMs.

| Model | CIFAR-10 | | | CIFAR-100 | | | fMoW | | |
|---|---|---|---|---|---|---|---|---|---|
| | FID ↓ | MSE ↓ | Acc ↑ | FID ↓ | MSE ↓ | Acc ↑ | FID ↓ | MSE ↓ | Acc ↑ |
| NVAE [87] | 36.4 | **0.25** | 18.8 | 42.5 | 0.53 | 4.1 | 82.25 | **0.30** | 27.7 |
| DDIM [80] | **4.16** | 2.5 | 22.5 | **10.16** | 3.2 | 2.2 | **37.74** | 3.0 | 23.5 |
| D2C (Ours) | 10.15 | 0.76 | **76.02** | 14.62 | **0.44** | **42.75** | 44.7 | 2.33 | **66.9** |

## 7.1 Unconditional generation

For unconditional generation, we measure the sample quality of images using the Frechet Inception Distance (FID, [38]) with 50,000 images. In particular, we extensively evaluate NVAE [87] and DDIM [80], a competitive VAE model and a competitive diffusion model as baselines because we can directly obtain features from them without additional optimization steps[4]. For them, we report mean-squared reconstruction error (MSE, summed over all pixels, pixels normalized to $[0, 1]$) and linear classification accuracy (Acc., measured in percentage) over $\mathbf{z}_1$ features for the test set.

We report sample quality results[5] in Tables 2, and 3. For FID, we outperform NVAE in all datasets and outperform DDIM on CelebA-64 and CelebA-HQ-256, which suggests our results are competitive with state-of-the-art non-adversarial generative models. In Table 2, we additionally compare NVAE, DDIM and D2C in terms of reconstruction and linear classification accuracy. As all three methods contain reconstruction losses, the MSE values are low and comparable. However, D2C enjoys much better linear classification accuracy than the other two thanks to the contrastive SSL component. We further note that training the same contrastive SSL method without $L_{\mathrm{D2}}$ achieves slightly higher $78.3\%$ accuracy on CIFAR-10. We tried improving this via ResNet [36] encoders, but this significantly increased reconstruction error, possibly due to loss of information in average pooling layers.

[4]For DDIM, the latent representations $\mathbf{x}^{(0)}$ are obtained by reversing the neural ODE process.

[5]Due to space limits, we place additional CIFAR-10 results in Appendix D.

Table 3: FID scores over different faces dataset with LVGMs.

| Model | CelebA-64 | CelebA-HQ-256 | FFHQ-256 |
|---|---|---|---|
| NVAE [87] | 13.48 | 40.26 | 26.02 |
| DDIM [80] | 6.53 | 25.6 | - |
| D2C (Ours) | **5.7** | **18.74** | **13.04** |

Table 4: Sample quality as a function of diffusion steps.

| | CIFAR-10 | | | CIFAR-100 | | | CelebA-64 | | |
|---|---|---|---|---|---|---|---|---|---|
| Steps | 10 | 50 | 100 | 10 | 50 | 100 | 10 | 50 | 100 |
| DDPM [40] | 41.07 | 8.01 | 5.78 | 50.27 | 21.37 | 16.72 | 33.12 | 18.48 | 13.93 |
| DDIM [80] | **13.36** | **4.67** | **4.16** | **23.34** | **11.69** | **10.16** | 17.33 | 9.17 | 6.53 |
| D2C (Ours) | 17.71 | 10.11 | 10.15 | 23.16 | 14.62 | 14.46 | **17.32** | **6.8** | **5.7** |

### 7.2 Few-shot conditional generation from examples

We demonstrate the advantage of D2C representations by performing few-shot conditional generation over labels. We consider two types of labeled examples: one has binary labels for which we train a binary classifier; the other is positive-only labeled (*e.g.*, images of female class) for which we train a PU classifier. Our goal here is to generate a diverse group of images with a certain label. We evaluate and compare three models: D2C, NVAE and DDIM. We train a classifier $r_\psi(\mathbf{c}|\mathbf{z})$ over the latent space of these models; we also train a image space classifier and use it with DDIM (denoted as DDIM-I). We run Algorithm 1 for these models, where line 4 is implemented via rejection sampling. As our goal is to compare different models, we leave more sophisticated methods [25] as future work.

We consider performing 8 conditional generation tasks over CelebA-64 with 2 binary classifiers (trained over 100 samples, 50 for each class) and 4 PU classifiers (trained over 100 positively labeled and 10k unlabeled samples). We also report a "naive" approach where we use all the training images (regardless of labels) and compute its FID with the corresponding subset of images (*e.g.*, all images versus blond images). In Table 5, we report the FID score between generated images (5k samples) and real images of the corresponding label. These results suggest that D2C outperforms the other approaches, and is the only one that performs better than the "naive" approach in most cases, illustrating the advantage of contrastive representations for few-shot conditional generation.

Table 5: FID scores for few-shot conditional generation with various types of labeled examples. Naive performs very well for non-blond due to class percentages.

| Method | Classes (% in train set) | D2C | DDIM | NVAE | DDIM-I | Naive |
|---|---|---|---|---|---|---|
| Binary | Male (42%) | **13.44** | 38.38 | 41.07 | 29.03 | 26.34 |
| | Female (58%) | **9.51** | 19.25 | 16.57 | 15.17 | 18.72 |
| | Blond (15%) | **17.61** | 31.39 | 31.24 | 29.09 | 27.51 |
| | Non-Blond (85%) | **8.94** | 9.67 | 16.73 | 19.76 | 3.77 |
| PU | Male (42%) | **16.39** | 37.03 | 42.78 | 19.60 | 26.34 |
| | Female (58%) | **12.21** | 15.42 | 18.36 | 14.96 | 18.72 |
| | Blond (15%) | **10.09** | 30.20 | 31.06 | 76.52 | 27.51 |
| | Non-Blond (85%) | **9.09** | 9.70 | 17.98 | 9.90 | 3.77 |

### 7.3 Few-shot conditional generation from manipulation constraints

Finally, we consider image manipulation where we use binary classifiers that are learned over 50 labeled instances for each class. We perform Amazon Mechanical Turk (AMT) evaluations over two attributes in the CelebA-256 dataset, *blond* and *red lipstick*, over D2C, DDIM, NVAE and

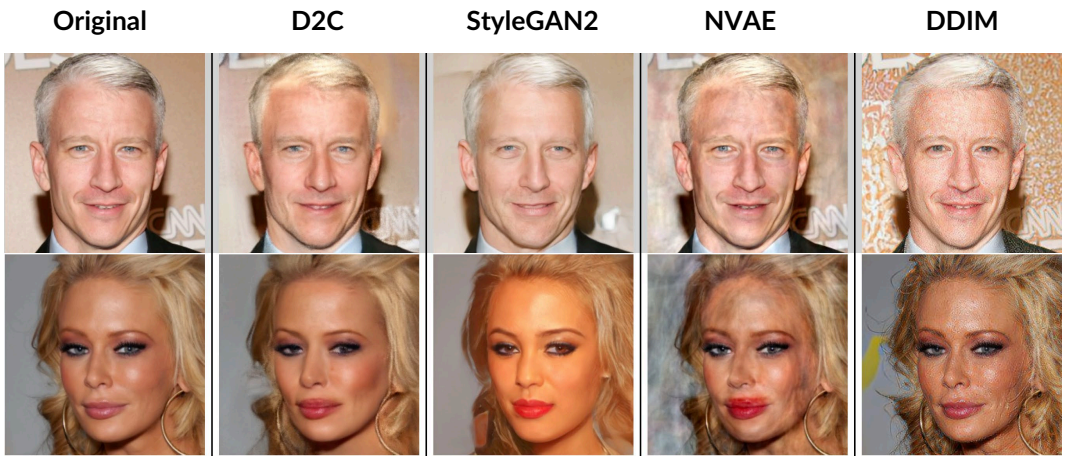



Figure 4: Image manipulation results for *blond* (top) and *red lipstick* (bottom). D2C is better than StyleGAN2 at preserving details of the original image, such as eyes, earrings, and background.

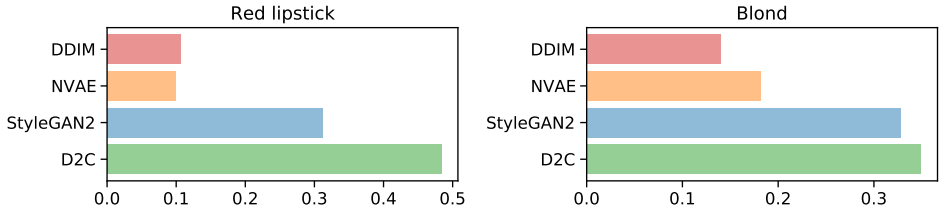



Figure 5: AMT evaluation over image manipulations. $x$-axis shows the percentage that the evaluator selects the image generated from the corresponding model out of 4 images from each model.

StyleGAN2 [47] (see Figure 4). The evaluation is double-blinded: neither we nor the evaluators know the correspondence between generated image and underlying model during the study. We include more details (algorithm, setup and human evaluation) in Appendix C and additional qualitative results (such as *beard* and *gender* attributes) in Appendix D.

In Figure 5, we show the percentage of manipulations preferred by AMT evaluators for each model; D2C slightly outperforms StyleGAN2 for *blond* and significantly outperforms StyleGAN2 for *red lipstick*. When we compare D2C with only StyleGAN2, D2C is preferred over $51.5\%$ for *blond* and $60.8\%$ for *red lipstick*. An additional advantage of D2C is that the manipulation is much faster than StyleGAN2, since the latter requires additional optimization over the latent space to improve reconstruction [101]. On the same Nvidia 1080Ti GPU, it takes 0.013 seconds to obtain the latent code in D2C, while the same takes 8 seconds [101] for StyleGAN2 ($615\times$ slower). As decoding is very fast for both models, D2C generations are around two orders of magnitude faster to produce.

## 8 Discussions and Limitations

We introduced D2C, a VAE-based generative model with a latent space suitable for few-shot conditional generation. To our best knowledge, our model is the first unconditional VAE to demonstrate superior image manipulation performance than StyleGAN2, which is surprising given our use of a regular NVAE architecture. We believe that with better architectures, such as designs from StyleGAN2 or Transformers [42], D2C can achieve even better performance. It is also interesting to formally investigate the integration between D2C and other types of conditions on the latent space, as well as training D2C in conjunction with other domains and data modalities, such as text [71], in a fashion that is similar to semi-supervised learning. Nevertheless, we note that our model have to be used properly in order to mitigate potential negative societal impacts, such as deep fakes.

# References


[1] Alexander A Alemi, Ben Poole, Ian Fischer, Joshua V Dillon, Rif A Saurous, and Kevin Murphy. Fixing a broken ELBO. *arXiv preprint arXiv:1711.00464*, November 2017.

[2] Jyoti Aneja, Harsh Agrawal, Dhruv Batra, and Alexander Schwing. Sequential latent spaces for modeling the intention during diverse image captioning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4261–4270, 2019.

[3] Jyoti Aneja, Alexander Schwing, Jan Kautz, and Arash Vahdat. NCP-VAE: Variational autoencoders with noise contrastive priors. *arXiv preprint arXiv:2010.02917*, October 2020.

[4] Martin Arjovsky, Soumith Chintala, and Léon Bottou. Wasserstein GAN. *arXiv preprint arXiv:1701.07875*, January 2017.

[5] Muhammad Asim, Ali Ahmed, and Paul Hand. Invertible generative models for inverse problems: mitigating representation error and dataset bias. *arXiv preprint arXiv:1905.11672*, May 2019.

[6] Sergey Bartunov and Dmitry Vetrov. Few-shot generative modelling with generative matching networks. In Amos Storkey and Fernando Perez-Cruz, editors, *Proceedings of the Twenty-First International Conference on Artificial Intelligence and Statistics*, volume 84 of *Proceedings of Machine Learning Research*, pages 670–678. PMLR, 2018.

[7] David Bau, Jun-Yan Zhu, Jonas Wulff, William Peebles, Hendrik Strobelt, Bolei Zhou, and Antonio Torralba. Seeing what a gan cannot generate. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4502–4511, 2019.

[8] Andrew Brock, Jeff Donahue, and Karen Simonyan. Large scale GAN training for high fidelity natural image synthesis. *arXiv preprint arXiv:1809.11096*, September 2018.

[9] Andrew Brock, Theodore Lim, James M Ritchie, and Nick Weston. Neural photo editing with introspective adversarial networks. *arXiv preprint arXiv:1609.07093*, 2016.

[10] Tom B Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. Language models are Few-Shot learners. *arXiv preprint arXiv:2005.14165*, May 2020.

[11] Emmanuel J Candes, Justin K Romberg, and Terence Tao. Stable signal recovery from incomplete and inaccurate measurements. *Communications on Pure and Applied Mathematics: A Journal Issued by the Courant Institute of Mathematical Sciences*, 59(8):1207–1223, 2006.

[12] Qifeng Chen and Vladlen Koltun. Photographic image synthesis with cascaded refinement networks. In *ICCV*, 2017.

[13] Ricky T Q Chen, Yulia Rubanova, Jesse Bettencourt, and David Duvenaud. Neural ordinary differential equations. *arXiv preprint arXiv:1806.07366*, June 2018.

[14] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. *arXiv preprint arXiv:2002.05709*, February 2020.

[15] Xinlei Chen, Haoqi Fan, Ross Girshick, and Kaiming He. Improved baselines with momentum contrastive learning. *arXiv preprint arXiv:2003.04297*, March 2020.

[16] Xinlei Chen and Kaiming He. Exploring simple siamese representation learning. *arXiv preprint arXiv:2011.10566*, November 2020.

[17] Xinlei Chen, Saining Xie, and Kaiming He. An empirical study of training Self-Supervised vision transformers. *arXiv preprint arXiv:2104.02057*, April 2021.

[18] Rewon Child. Very deep vaes generalize autoregressive models and can outperform them on images. *arXiv preprint arXiv:2011.10650*, 2020.

[19] Gordon Christie, Neil Fendley, James Wilson, and Ryan Mukherjee. Functional map of the world. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6172–6180, 2018.

[20] Louis Clouâtre and Marc Demers. Figr: Few-shot image generation with reptile. *arXiv preprint arXiv:1901.02199*, 2019.

[21] Bin Dai and David Wipf. Diagnosing and enhancing VAE models. *arXiv preprint arXiv:1903.05789*, March 2019.

[22] Giannis Daras, Joseph Dean, Ajil Jalal, and Alexandros G Dimakis. Intermediate layer optimization for inverse problems using deep generative models. *arXiv preprint arXiv:2102.07364*, February 2021.

[23] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, October 2018.

[24] Prafulla Dhariwal, Heewoo Jun, Christine Payne, Jong Wook Kim, Alec Radford, and Ilya Sutskever. Jukebox: A generative model for music. *arXiv preprint arXiv:2005.00341*, 2020.

[25] Prafulla Dhariwal and Alex Nichol. Diffusion models beat GANs on image synthesis. *arXiv preprint arXiv:2105.05233*, May 2021.

[26] L Dinh, D Krueger, and Y Bengio. NICE: Non-linear independent components estimation. *arXiv preprint arXiv:1410.8516*, 2014.

[27] Laurent Dinh, Jascha Sohl-Dickstein, and Samy Bengio. Density estimation using real NVP. *arXiv preprint arXiv:1605.08803*, May 2016.

[28] Jeff Donahue, Philipp Krähenbühl, and Trevor Darrell. Adversarial feature learning. *arXiv preprint arXiv:1605.09782*, May 2016.

[29] Yilun Du and Igor Mordatch. Implicit generation and generalization in Energy-Based models. *arXiv preprint arXiv:1903.08689*, March 2019.

[30] Vincent Dumoulin, Ishmael Belghazi, Ben Poole, Olivier Mastropietro, Alex Lamb, Martin Arjovsky, and Aaron Courville. Adversarially learned inference. *arXiv preprint arXiv:1606.00704*, June 2016.

[31] Charles Elkan and Keith Noto. Learning classifiers from only positive and unlabeled data. In *14th ACM SIGKDD*, pages 213–220, 2008.

[32] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In Z Ghahramani, M Welling, C Cortes, N D Lawrence, and K Q Weinberger, editors, *Advances in Neural Information Processing Systems 27*, pages 2672–2680. Curran Associates, Inc., 2014.

[33] Jean-Bastien Grill, Florian Strub, Florent Altché, Corentin Tallec, Pierre H Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Avila Pires, Zhaohan Daniel Guo, Mohammad Gheshlaghi Azar, Bilal Piot, Koray Kavukcuoglu, Rémi Munos, and Michal Valko. Bootstrap your own latent: A new approach to Self-Supervised learning. *arXiv preprint arXiv:2006.07733*, June 2020.

[34] Aditya Grover, Aaron Zweig, and Stefano Ermon. Graphite: Iterative generative modeling of graphs. In *International conference on machine learning*, pages 2434–2444. PMLR, 2019.

[35] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. *arXiv preprint arXiv:1911.05722*, November 2019.

[36] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. *arXiv preprint arXiv:1512.03385*, December 2015.

[37] Olivier Henaff. Data-efficient image recognition with contrastive predictive coding. In *International Conference on Machine Learning*, pages 4182–4192. PMLR, 2020.

[38] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. GANs trained by a two Time-Scale update rule converge to a local nash equilibrium. *arXiv preprint arXiv:1706.08500*, June 2017.

[39] Geoffrey E Hinton. Training products of experts by minimizing contrastive divergence. *Neural computation*, 14(8):1771–1800, August 2002.

[40] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *arXiv preprint arXiv:2006.11239*, June 2020.

[41] Matthew D Hoffman and Matthew J Johnson. Elbo surgery: yet another way to carve up the variational evidence lower bound. In *Workshop in Advances in Approximate Bayesian Inference, NIPS*, volume 1, page 2. approximateinference.org, 2016.

[42] Drew A Hudson and C Lawrence Zitnick. Generative adversarial transformers. *arXiv preprint arXiv:2103.01209*, 2021.

[43] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A Efros. Image-to-image translation with conditional adversarial networks. In *CVPR*, 2017.

[44] Zahra Kadkhodaie and Eero P Simoncelli. Solving linear inverse problems using the prior implicit in a denoiser. *arXiv preprint arXiv:2007.13640*, July 2020.

[45] Tero Karras, Timo Aila, Samuli Laine, and Jaakko Lehtinen. Progressive growing of GANs for improved quality, stability, and variation. *arXiv preprint arXiv:1710.10196*, October 2017.

[46] Tero Karras, Samuli Laine, and Timo Aila. A Style-Based generator architecture for generative adversarial networks. *arXiv preprint arXiv:1812.04948*, December 2018.

[47] Tero Karras, Samuli Laine, Miika Aittala, Janne Hellsten, Jaakko Lehtinen, and Timo Aila. Analyzing and improving the image quality of stylegan. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8110–8119, 2020.

[48] Diederik P Kingma, Tim Salimans, Rafal Jozefowicz, Xi Chen, Ilya Sutskever, and Max Welling. Improved variational inference with inverse autoregressive flow. In D D Lee, M Sugiyama, U V Luxburg, I Guyon, and R Garnett, editors, *Advances in Neural Information Processing Systems 29*, pages 4743–4751. Curran Associates, Inc., 2016.

[49] Diederik P Kingma and Max Welling. Auto-Encoding variational bayes. *arXiv preprint arXiv:1312.6114v10*, December 2013.

[50] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. ImageNet classification with deep convolutional neural networks. In F Pereira, C J C Burges, L Bottou, and K Q Weinberger, editors, *Advances in Neural Information Processing Systems 25*, pages 1097–1105. Curran Associates, Inc., 2012.

[51] Anders Boesen Lindbo Larsen, Søren Kaae Sønderby, Hugo Larochelle, and Ole Winther. Autoencoding beyond pixels using a learned similarity metric. In *International conference on machine learning*, pages 1558–1566. PMLR, 2016.

[52] C Li, H Liu, C Chen, Y Pu, L Chen, and others. Alice: Towards understanding adversarial learning for joint distribution matching. *Advances in neural information processing systems*, 2017.

[53] Chunyuan Li, Xiang Gao, Yuan Li, Baolin Peng, Xiujun Li, Yizhe Zhang, and Jianfeng Gao. Optimus: Organizing sentences via pre-trained modeling of a latent space. *arXiv preprint arXiv:2004.04092*, 2020.

[54] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *European conference on computer vision*, pages 740–755. Springer, 2014.

[55] Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Deep learning face attributes in the wild. In *Proceedings of the IEEE international conference on computer vision*, pages 3730–3738, 2015.

[56] Alireza Makhzani, Jonathon Shlens, Navdeep Jaitly, Ian Goodfellow, and Brendan Frey. Adversarial autoencoders. *arXiv preprint arXiv:1511.05644*, 2015.

[57] Elman Mansimov, Emilio Parisotto, Jimmy Lei Ba, and Ruslan Salakhutdinov. Generating images from captions with attention. *arXiv preprint arXiv:1511.02793*, 2015.

[58] Mehdi Mirza and Simon Osindero. Conditional generative adversarial nets. *arXiv preprint arXiv:1411.1784*, November 2014.

[59] Gautam Mittal, Jesse Engel, Curtis Hawthorne, and Ian Simon. Symbolic music generation with diffusion models. *arXiv preprint arXiv:2103.16091*, March 2021.

[60] Takeru Miyato, Toshiki Kataoka, Masanori Koyama, and Yuichi Yoshida. Spectral normalization for generative adversarial networks. *arXiv preprint arXiv:1802.05957*, February 2018.

[61] Shakir Mohamed and Balaji Lakshminarayanan. Learning in implicit generative models. *arXiv preprint arXiv:1610.03483*, October 2016.

[62] Alex Najibi. Racial Discrimination in Face Recognition Technology, 2020.

[63] Anh Nguyen, Jeff Clune, Yoshua Bengio, Alexey Dosovitskiy, and Jason Yosinski. Plug & play generative networks: Conditional iterative generation of images in latent space. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4467–4477, 2017.

[64] Chenhao Niu, Yang Song, Jiaming Song, Shengjia Zhao, Aditya Grover, and Stefano Ermon. Permutation invariant graph generation via score-based generative modeling. In *International Conference on Artificial Intelligence and Statistics*, pages 4474–4484. PMLR, 2020.

[65] Mehdi Noroozi and Paolo Favaro. Unsupervised learning of visual representations by solving jigsaw puzzles. In *European conference on computer vision*, pages 69–84. Springer, 2016.

[66] Taesung Park, Jun-Yan Zhu, Oliver Wang, Jingwan Lu, Eli Shechtman, Alexei A Efros, and Richard Zhang. Swapping autoencoder for deep image manipulation. *arXiv preprint arXiv:2007.00653*, July 2020.

[67] Or Patashnik, Zongze Wu, Eli Shechtman, Daniel Cohen-Or, and Dani Lischinski. StyleCLIP: Text-Driven manipulation of StyleGAN imagery. *arXiv preprint arXiv:2103.17249*, March 2021.

[68] Wei Ping, Kainan Peng, Kexin Zhao, and Zhao Song. Waveflow: A compact flow-based model for raw audio. In *International Conference on Machine Learning*, pages 7706–7716. PMLR, 2020.

[69] Tiziano Portenier, Qiyang Hu, Attila Szabó, Siavash Arjomand Bigdeli, Paolo Favaro, and Matthias Zwicker. Faceshop: Deep sketch-based face image editing. *ACM Transactions on Graphics*, 37(4), 2018.

[70] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. *arXiv preprint arXiv:2103.00020*, February 2021.

[71] Aditya Ramesh, Mikhail Pavlov, Gabriel Goh, Scott Gray, Chelsea Voss, Alec Radford, Mark Chen, and Ilya Sutskever. Zero-Shot Text-to-Image generation. *arXiv preprint arXiv:2102.12092*, February 2021.

[72] Ali Razavi, Aaron van den Oord, and Oriol Vinyals. Generating diverse high-fidelity images with vq-vae-2. *arXiv preprint arXiv:1906.00446*, 2019.

[73] Scott Reed, Zeynep Akata, Xinchen Yan, Lajanugen Logeswaran, Bernt Schiele, and Honglak Lee. Generative adversarial text to image synthesis. In *International Conference on Machine Learning*, pages 1060–1069. PMLR, 2016.

[74] Danilo Jimenez Rezende and Shakir Mohamed. Variational inference with normalizing flows. *arXiv preprint arXiv:1505.05770*, May 2015.

[75] Gareth O Roberts and Jeffrey S Rosenthal. Optimal scaling of discrete approximations to langevin diffusions. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 60(1):255–268, 1998.

[76] Mihaela Rosca, Balaji Lakshminarayanan, and Shakir Mohamed. Distribution matching in variational inference. *arXiv preprint arXiv:1802.06847*, February 2018.

[77] Yujun Shen, Jinjin Gu, Xiaoou Tang, and Bolei Zhou. Interpreting the latent space of gans for semantic face editing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9243–9252, 2020.

[78] Jascha Sohl-Dickstein, Eric A Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. *arXiv preprint arXiv:1503.03585*, March 2015.

[79] Jiaming Song and Stefano Ermon. Multi-label contrastive predictive coding. *arXiv preprint arXiv:2007.09852*, 2020.

[80] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. *arXiv preprint arXiv:2010.02502*, 2020.

[81] Yang Song and Stefano Ermon. Generative modeling by estimating gradients of the data distribution. *arXiv preprint arXiv:1907.05600*, July 2019.

[82] Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. *arXiv preprint arXiv:2011.13456*, 2020.

[83] Hiroshi Takahashi, Tomoharu Iwata, Yuki Yamanaka, Masanori Yamada, and Satoshi Yagi. Variational autoencoder with implicit optimal priors. *Proceedings of the AAAI Conference on Artificial Intelligence*, 33(01):5066–5073, July 2019.

[84] Ilya Tolstikhin, Olivier Bousquet, Sylvain Gelly, and Bernhard Schoelkopf. Wasserstein Auto-Encoders. *arXiv preprint arXiv:1711.01558*, November 2017.

[85] Jakub M Tomczak and Max Welling. Improving variational auto-encoders using householder flow. *arXiv preprint arXiv:1611.09630*, 2016.

[86] Jakub M Tomczak and Max Welling. VAE with a VampPrior. *arXiv preprint arXiv:1705.07120*, May 2017.

[87] Arash Vahdat and Jan Kautz. NVAE: A deep hierarchical variational autoencoder. *arXiv preprint arXiv:2007.03898*, July 2020.

[88] Aaron van den Oord, Sander Dieleman, Heiga Zen, Karen Simonyan, Oriol Vinyals, Alex Graves, Nal Kalchbrenner, Andrew Senior, and Koray Kavukcuoglu. WaveNet: A generative model for raw audio. *arXiv preprint arXiv:1609.03499*, September 2016.

[89] Aaron van den Oord, Nal Kalchbrenner, and Koray Kavukcuoglu. Pixel recurrent neural networks. *arXiv preprint arXiv:1601.06759*, January 2016.

[90] Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, July 2018.

[91] Aaron van den Oord, Oriol Vinyals, and Koray Kavukcuoglu. Neural discrete representation learning. *arXiv preprint arXiv:1711.00937*, November 2017.

[92] Ting-Chun Wang, Ming-Yu Liu, Jun-Yan Zhu, Andrew Tao, Jan Kautz, and Bryan Catanzaro. High-resolution image synthesis and semantic manipulation with conditional gans. In *CVPR*, 2018.

[93] Weihao Xia, Yujiu Yang, Jing-Hao Xue, and Baoyuan Wu. TediGAN: Text-Guided diverse face image generation and manipulation. 2021.

[94] Zhe Xie, Chengxuan Liu, Yichi Zhang, Hongtao Lu, Dong Wang, and Yue Ding. Adversarial and contrastive variational autoencoder for sequential recommendation. *arXiv preprint arXiv:2103.10693*, March 2021.

[95] Yinghao Xu, Yujun Shen, Jiapeng Zhu, Ceyuan Yang, and Bolei Zhou. Generative hierarchical features from synthesizing images. *arXiv e-prints*, pages arXiv–2007, 2020.

[96] Jiaxuan You, Bowen Liu, Rex Ying, Vijay Pande, and Jure Leskovec. Graph convolutional policy network for Goal-Directed molecular graph generation. *arXiv preprint arXiv:1806.02473*, June 2018.

[97] Lantao Yu, Weinan Zhang, Jun Wang, and Yong Yu. Seqgan: Sequence generative adversarial nets with policy gradient. In *Proceedings of the AAAI conference on artificial intelligence*, volume 31, 2017.

[98] Shengjia Zhao, Jiaming Song, and Stefano Ermon. InfoVAE: Information maximizing variational autoencoders. *arXiv preprint arXiv:1706.02262*, June 2017.

[99] Shengjia Zhao, Jiaming Song, and Stefano Ermon. Towards deeper understanding of variational autoencoding models. *arXiv preprint arXiv:1702.08658*, February 2017.

[100] Shengjia Zhao, Jiaming Song, and Stefano Ermon. A lagrangian perspective on latent variable generative models. In *Proc. 34th Conference on Uncertainty in Artificial Intelligence*, 2018.

[101] Jiapeng Zhu, Yujun Shen, Deli Zhao, and Bolei Zhou. In-domain gan inversion for real image editing. In *European Conference on Computer Vision*, pages 592–608. Springer, 2020.

[102] Jiapeng Zhu, Deli Zhao, Bolei Zhou, and Bo Zhang. Lia: Latently invertible autoencoder with adversarial learning. 2019.

# Checklist

1. For all authors...
   (a) Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope? [Yes]
   (b) Did you describe the limitations of your work? [Yes] In Section 8.
   (c) Did you discuss any potential negative societal impacts of your work? [Yes] Details in Section 8 and Appendix E.
   (d) Have you read the ethics review guidelines and ensured that your paper conforms to them? [Yes]
2. If you are including theoretical results...
   (a) Did you state the full set of assumptions of all theoretical results? [Yes]
   (b) Did you include complete proofs of all theoretical results? [Yes] In the Appendix.
3. If you ran experiments...
   (a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? [Yes]
   (b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [No] We report everything except data splits for standard datasets such as CIFAR-10.
   (c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [No] The experiments are simply too expensive (and environmentally harmful) to be run for multiple times in order to evaluate error bars; this is also common practice in the generative modeling literature.
   (d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? [Yes] In Appendix C.
4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...
   (a) If your work uses existing assets, did you cite the creators? [Yes]
   (b) Did you mention the license of the assets? [N/A] CIFAR-10 does not have a license.
   (c) Did you include any new assets either in the supplemental material or as a URL? [N/A] No new assets are included.
   (d) Did you discuss whether and how consent was obtained from people whose data you're using/curating? [N/A] All the datasets are public.
   (e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [N/A] All the datasets are public.
5. If you used crowdsourcing or conducted research with human subjects...
   (a) Did you include the full text of instructions given to participants and screenshots, if applicable? [Yes] In Appendix C.
   (b) Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable? [N/A] No biomedical data is used.
   (c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? [Yes] In Appendix C.