InfoDiffusion: Representation Learning Using Information Maximizing Diffusion Models

Yingheng Wang [1] **Yair Schiff** [1 2] **Aaron Gokaslan** [1 2] **Weishen Pan** [3]
Fei Wang [3] **Christopher De Sa** [1] **Volodymyr Kuleshov** [1 2]

Abstract

While diffusion models excel at generating high-quality samples, their latent variables typically lack semantic meaning and are not suitable for representation learning. Here, we propose InfoDiffusion, an algorithm that augments diffusion models with low-dimensional latent variables that capture high-level factors of variation in the data. InfoDiffusion relies on a learning objective regularized with the mutual information between observed and hidden variables, which improves latent space quality and prevents the latents from being ignored by expressive diffusion-based decoders. Empirically, we find that InfoDiffusion learns disentangled and human-interpretable latent representations that are competitive with state-of-the-art generative and contrastive methods, while retaining the high sample quality of diffusion models. Our method enables manipulating the attributes of generated images and has the potential to assist tasks that require exploring a learned latent space to generate quality samples, e.g., generative design.

1. Introduction

Diffusion models are a family of generative models characterized by high sample quality (Ho et al., 2020; Dhariwal & Nichol, 2021; Rombach et al., 2021). These models achieve state-of-the-art performance across a range of generative tasks, including image generation (Dhariwal & Nichol, 2021; Ramesh et al., 2022), audio synthesis (Kong et al., 2020), and molecule design (Jing et al., 2022; Xu et al., 2022).

[1]Department of Computer Science, Cornell University, Ithaca, NY, USA [2]Department of Computer Science, Cornell Tech, New York City, NY, USA [3]Department of Population Health Sciences, Weill Cornell Medicine, New York City, NY, USA. Correspondence to: Yingheng Wang <yw2349@cornell.edu>, Volodymyr Kuleshov <kuleshov@cornell.edu>.

Proceedings of the 40th International Conference on Machine Learning, Honolulu, Hawaii, USA. PMLR 202, 2023. Copyright 2023 by the author(s).

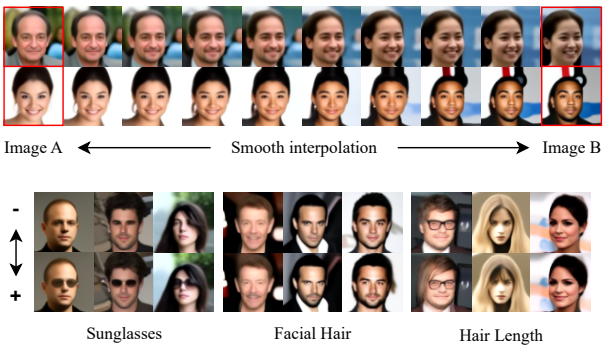


Figure 1. **InfoDiffusion produces semantically meaningful latent space for a diffusion model.** (*Top*) Smooth latent space. (*Bottom*) Disentangled, human-interpretable factors of variation.

However, diffusion models rely on latent variables that typically lack semantic meaning and are not well-suited for the task of representation learning (Yang et al., 2022)—the unsupervised discovery of high-level concepts in data (e.g., topics across news articles, facial features in human photos, clusters of related molecules). This paper seeks to endow diffusion models with a semantically meaningful latent space while retaining their high sample quality.

Specifically, we propose InfoDiffusion, an algorithm that augments diffusion models with low-dimensional latent variables that capture high-level factors of variation in the data. InfoDiffusion relies on variational inference to optimize the mutual information between the low-dimensional latents and the generated samples (Zhao et al., 2017); this prevents expressive diffusion-based generators from ignoring auxiliary latents and promotes their use for storing semantically meaningful and disentangled information (Chen et al., 2016).

The InfoDiffusion algorithm generalizes several existing methods for representation learning (Kingma & Welling, 2013; Makhzani et al., 2015; Higgins et al., 2017). Our method is a principled probabilistic extension of DiffAE (Preechakul et al., 2022) that supports custom priors and discrete latents and improves latents via mutual information regularization. It also extends InfoVAEs (Zhao et al., 2017) to leverage more flexible diffusion-based decoders. See Figure 2 for an overview of our method.

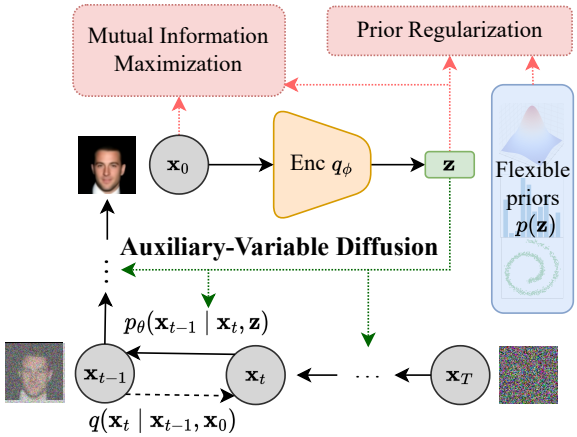


Figure 2. Flow chart demonstrating auxiliary-variable diffusion model with mutual information and prior regularization.

We evaluate InfoDiffusion on a suite of benchmark datasets and find that it learns latent representations that are competitive with state-of-the-art generative and contrastive methods (Chen et al., 2020a;b; Caron et al., 2021), while retaining the high sample quality of diffusion models. Unlike many existing methods, InfoDiffusion finds disentangled representations that accurately capture distinct human-interpretable factors of variation; see Figure 1 for examples.

Contributions In summary, we make the following contributions: (1) we propose a principled probabilistic extension of diffusion models that supports low-dimensional latents; (2) we introduce associated variational learning objectives that are regularized with a mutual information term; (3) we show that these algorithms simultaneously yield high-quality samples and latent representations, achieving competitive performance with state-of-the-art methods on both fronts.

2. Background

A diffusion model defines a latent variable distribution $p(\mathbf{x}_{0:T})$ over data $\mathbf{x}_0$ sampled from the data distribution, as well as latents $\mathbf{x}_{1:T} := \mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_T$ that represent a gradual transformation of $\mathbf{x}_0$ into random Gaussian noise $\mathbf{x}_T$. The distribution p factorizes as a Markov chain

$$p(\mathbf{x}_{0:T}) = p(\mathbf{x}_T) \prod_{t=0}^{T-1} p_\theta(\mathbf{x}_t \mid \mathbf{x}_{t+1}). \tag{1}$$

that maps noise $\mathbf{x}_T$ into data $\mathbf{x}_0$ by "undoing" a noising (or diffusion) process denoted by q. Here we use a learned denoising distribution p_θ, which we parameterize by a neural network with parameters θ.

The noising process q starts from a clean $\mathbf{x}_0$, drawn from the data distribution (denoted by $q(\mathbf{x}_0)$) and defines a sequence of T variables $\mathbf{x}_1, \ldots, \mathbf{x}_T$ via a Markov chain that factorizes as

$$q(\mathbf{x}_{1:T} \mid \mathbf{x}_0) = \prod_{t=1}^{T} q(\mathbf{x}_t \mid \mathbf{x}_{t-1}). \tag{2}$$

In this factorization, we define $q(\mathbf{x}_t \mid \mathbf{x}_{t-1}) = \mathcal{N}(\mathbf{x}_t; \sqrt{\alpha_t}\mathbf{x}_{t-1}, \sqrt{1-\alpha_t}\mathbf{I})$ as a Gaussian distribution centered around a progressively corrupted version of $\mathbf{x}_{t-1}$ with a schedule $\alpha_1, \alpha_2, \ldots, \alpha_T$. As shown in Ho et al. (2020), the marginal distribution of q can be expressed as

$$q(\mathbf{x}_t \mid \mathbf{x}_0) = \mathcal{N}(\mathbf{x}_t; \sqrt{\bar{\alpha}_t}\mathbf{x}_0, \sqrt{1-\bar{\alpha}_t}\mathbf{I}),$$

where $\bar{\alpha}_t = \prod_{s=1}^{t} \alpha_t$ is the cumulative product of the schedule parameters α_t.

Normally, p is trained via maximization of an evidence lower bound (ELBO) objective derived using variational inference:

$$\log p(\mathbf{x}_0) \geq \mathbb{E}_{q(\mathbf{x}_1 \mid \mathbf{x}_0)}[\log p_\theta(\mathbf{x}_0 \mid \mathbf{x}_1)] - \mathrm{KL}(q(\mathbf{x}_T \mid \mathbf{x}_0) || p(\mathbf{x}_T)) - \sum_{t=2}^{T} \mathbb{E}_{q(\mathbf{x}_t \mid \mathbf{x}_0)}[\mathrm{KL}(q(\mathbf{x}_{t-1} \mid \mathbf{x}_t, \mathbf{x}_0) || p_\theta(\mathbf{x}_{t-1} \mid \mathbf{x}_t))],$$

where KL denotes the Kullback–Leibler divergence.

Unsupervised Representation Learning A core aim of generative modeling is representation learning, the unsupervised extraction of latent concepts from data. Generative models $p(\mathbf{x}, \mathbf{z})$ typically represent latent concepts using low-dimensional variables $\mathbf{z}$ that are inferred via posterior inference over $p(\mathbf{z} \mid \mathbf{x})$. VAEs exemplify this framework but do not produce state-of-the-art samples. Conversely, diffusion models produce high-quality samples but lack an interpretable low-dimensional latent space, making them unsuitable for representation learning.

3. Diffusion Models With Auxiliary Latents

This paper seeks to endow diffusion models with a semantically meaningful latent space while retaining their high sample quality. Our strategy is three-fold: (1) in this section, we define a diffusion model family that supports low-dimensional latent variables; (2) in Section 4, we define learning objectives for this model family; (3) in Section 5, we define a regularizer based on mutual information that further encourages the model to learn high-quality latents.

Specifically, we define an auxiliary-variable diffusion model as a probability distribution $p(\mathbf{x}_{0:T}, \mathbf{z})$ that factorizes as:

$$p(\mathbf{x}_{0:T}, \mathbf{z}) = p(\mathbf{x}_T) p(\mathbf{z}) \prod_{t=1}^{T} p_\theta(\mathbf{x}_{t-1} \mid \mathbf{x}_t, \mathbf{z}). \tag{3}$$

This model implements a reverse diffusion process $p_\theta(\mathbf{x}_{t-1} \mid \mathbf{x}_t, \mathbf{z})$ over $\mathbf{x}_{0:T}$ conditioned on auxiliary latents

$\mathbf{z}$ distributed according to a prior $p(\mathbf{z})$. The $\mathbf{z}$ is independent of the forward process because $\mathbf{z}$ is meant to be a latent representation of the input, not a control variable of diffusion.

3.1. Auxiliary Latent Variables and Semantic Prior

The goal of the auxiliary latents $\mathbf{z}$ is to encode a high-level representation of $\mathbf{x}_0$. Unlike $\mathbf{x}_{1:T}$, the $\mathbf{z}$ are not constrained to have a particular dimension and can represent a low-dimensional vector of latent factors of variation. They can be continuous, as well as discrete.

The prior $p(\mathbf{z})$ ensures that we have a principled probabilistic model and enables the unconditional sampling of $\mathbf{x}_0$. The prior can also be used to encode domain knowledge about $\mathbf{z}$—e.g., if we know that the dataset contains K distinct classes, we may set $p(\mathbf{z})$ to be a mixture of K components. Alternatively, we may set $p(\mathbf{z})$ to be a simple distribution from which we can easily sample (e.g., a Gaussian).

3.2. Auxiliary-Variable Diffusion Decoder

The decoder $p_\theta(\mathbf{x}_{t-1}\,|\,\mathbf{x}_t,\mathbf{z})$ is conditioned on the auxiliary latents $\mathbf{z}$. In a trained model, the $\mathbf{z}$ are responsible for high-level concepts (e.g., the age or skin color of a person), while the sequence of $\mathbf{x}_t$ progressively adds lower-level details (e.g., hair texture).

Following previous work (Ho et al., 2020), we define the decoder

$$p_\theta(\mathbf{x}_{t-1}\,|\,\mathbf{x}_t,\mathbf{z}) = \frac{1}{\sqrt{\alpha_t}}\left(\mathbf{x}_t - \frac{1-\alpha_t}{\sqrt{1-\bar{\alpha}_t}}\epsilon_\theta(\mathbf{x}_t,t,\mathbf{z})\right)$$

with a noise prediction network $\epsilon_\theta(\mathbf{x}_{t-1},t,\mathbf{z})$ parameterized by a U-Net (Ronneberger et al., 2015). We condition this network on $\mathbf{z}$ using adaptive group normalization layers (AGN), inspired by Dhariwal & Nichol (2021),

$$\mathrm{AGN}(\mathbf{h},\mathbf{z}) = (1+\mathbf{s}(\mathbf{z}))\cdot\mathrm{GroupNorm}(\mathbf{h}) + \mathbf{b}(\mathbf{z}).$$

Specifically, we implement two successive AGN layers for the auxiliary variable and time embeddings, respectively, to fuse them into each residual block.

4. Learning and Inference Algorithms For Auxiliary-Variable Diffusion Models

Next, we introduce learning algorithms for auxiliary-variable models based on variational inference. We refer to the resulting method as variational auxiliary-variable diffusion.

4.1. Variational Inference for Auxiliary-Variable Models

We apply variational inference twice to form a variational lower bound on the marginal log-likelihood of the data (see the full derivation in Appendix A):

$$\begin{aligned}
\log p(\mathbf{x}_0) &= \log \mathbb{E}_{q_\mathbf{z}}\left[\frac{p(\mathbf{x}_0,\mathbf{z})}{q_\phi(\mathbf{z}\,|\,\mathbf{x}_0)}\right] \\
&\geq \mathbb{E}_{q_\mathbf{z}}\left[\log \mathbb{E}_{q_\mathbf{x}}\left[\frac{p(\mathbf{x}_{0:T},\mathbf{z})}{q_\phi(\mathbf{z}\,|\,\mathbf{x}_0)q(\mathbf{x}_{1:T}\,|\,\mathbf{x}_0)}\right]\right] \\
&\geq \mathbb{E}_{q_\mathbf{x}}\left[\mathbb{E}_{q_\mathbf{z}}\left[\log\frac{p(\mathbf{x}_{0:T},\mathbf{z})}{q_\phi(\mathbf{z}\,|\,\mathbf{x}_0)q(\mathbf{x}_{1:T}\,|\,\mathbf{x}_0)}\right]\right] \\
&= \mathbb{E}_{q_{\mathbf{x}_1}}\left[\mathbb{E}_{q_\mathbf{z}}[\log p_\theta(\mathbf{x}_0\,|\,\mathbf{x}_1,\mathbf{z})]\right] - \mathrm{KL}(q(\mathbf{z}\,|\,\mathbf{x}_0)||p(\mathbf{z})) \\
&\quad - \mathrm{KL}(q(\mathbf{x}_T\,|\,\mathbf{x}_0)||p(\mathbf{x}_T)) - \sum_{t=2}^{T}\mathbb{E}_{q_{\mathbf{x}_t}}\left[\mathbb{E}_{q_\mathbf{z}}[\mathrm{KL}(q_t||p_t)]\right] \\
&:= \mathcal{L}_D(\mathbf{x}_0) \qquad (4)
\end{aligned}$$

where $\mathcal{L}_D(\mathbf{x}_0)$ denotes the ELBO for a variational auxiliary-variable diffusion model, q_t,p_t denote the distributions $q(\mathbf{x}_{t-1}\mid\mathbf{x}_t,\mathbf{x}_0)$ and $p_\theta(\mathbf{x}_{t-1}\mid\mathbf{x}_t,\mathbf{z})$, respectively, $q_\mathbf{z} := q_\phi(\mathbf{z}\mid\mathbf{x}_0)$ is an approximate variational posterior, $q_\mathbf{x} := q(\mathbf{x}_{1:T}\,|\,\mathbf{x}_0)$, and $q_{\mathbf{x}_t} := q(\mathbf{x}_t\,|\,\mathbf{x}_0)$.

We optimize the above objective end-to-end using gradient descent by using the reparameterization trick to backpropagate through samples from $q_\phi(\mathbf{z}\,|\,\mathbf{x}_0)$ (Kingma & Welling, 2013). We use a neural network with parameters ϕ to encode the parameters of the approximate posterior distribution of $\mathbf{z}$.

4.2. Inferring Latent Representations

Once the model is trained, we rely on the approximate posterior $q_\phi(\mathbf{z}\mid\mathbf{x}_0)$ to infer $\mathbf{z}$. In our experiments, we parameterize $q_\phi(\mathbf{z}\,|\,\mathbf{x}_0)$ as a UNet encoder (see Appendix E for more details).

Additionally, we may encode $\mathbf{x}_0$ into a latent-variable $\mathbf{x}_T$, which contains information not captured by the auxiliary variable $\mathbf{z}$—usually details such as texture and high-level frequencies. Our method iteratively runs the diffusion process using the learned noise model $\epsilon_\theta(\mathbf{x}_0,t,\mathbf{z})$:

$$\mathbf{x}_{t+1} = \sqrt{\bar{\alpha}_{t+1}}\hat{\mathbf{x}}_0(\mathbf{x}_t,t,\mathbf{z}) + \sqrt{1-\bar{\alpha}_{t+1}}\epsilon_\theta(\mathbf{x}_t,t,\mathbf{z}),$$

where $\mathbf{z}$ is a latent code and $\hat{\mathbf{x}}_0(\mathbf{x},\ t,\ \mathbf{z}) = \frac{1}{\sqrt{\bar{\alpha}_t}}\left(\mathbf{x}_t - \sqrt{1-\bar{\alpha}_t}\epsilon_\theta(\mathbf{x}_t,t,\mathbf{z})\right)$ is an estimate of $\mathbf{x}_0$ from $\mathbf{x}_t$.

4.3. Discrete Auxiliary-Variable Diffusion

In many settings, latent representations are inherently discrete—e.g., the presence of certain objects in a scene, the choice of topic in a text, etc. Variational auxiliary-variable diffusion supports such discrete variables via relaxation methods for deep latent variable models (Jang et al., 2016).

Specifically, at training time, we replace $\mathbf{z}$ with a continuous relaxation $\mathbf{z}_\tau$ sampled from q using the Gumbel-Softmax technique with a temperature τ. Higher temperatures τ yield

continuous approximations $\mathbf{z}_\tau$ of $\mathbf{z}$; as $\tau \to 0$, $\mathbf{z}_\tau$ approaches a discrete $\mathbf{z}$. We train using a categorical distribution for the prior $p(\mathbf{z})$, and we estimate gradients using the reparameterization trick. We anneal τ over the course of training to keep gradient variance in check. At inference time, we set $\tau = 0$ to obtain fully discrete latents. See Appendix G for more details.

4.4. Sampling Methods

At inference time, our model supports multiple sampling procedures. First, to generate $\mathbf{x}_0$ unconditionally, we can sample from the original prior $p(\mathbf{z})$, as in a VAE (see Appendix D.1 for details on generating high-quality samples with $\mathbf{z} \sim p(\mathbf{z})$). Alternatively, we can utilize a learned prior to potentially improve sample quality (see Appendix D.2 for details on implementing the learned prior used in Section 6). This learned prior is similar to the approach described in DiffAE (Preechakul et al., 2022), where a latent diffusion model is required to enable sampling.

5. InfoDiffusion: Regularizing Semantic Latents By Maximizing Mutual Information

Diffusion models with auxiliary latents face two risks. First, an expressive decoder $p_\theta(\mathbf{x}_{t-1} | \mathbf{x}_t, \mathbf{z})$ may choose to ignore low-dimensional latents $\mathbf{z}$ and generate $\mathbf{x}_{t-1}$ unconditionally (Chen et al., 2016). Second, the approximate posterior $q_\phi(\mathbf{z} | \mathbf{x}_0)$ may fail to match the prior $p(\mathbf{z})$ because the prior regularization term is too weak relative to the reconstruction term (Zhao et al., 2017). This degrades the quality of ancestral sampling as well as that of latent representations.

5.1. Regularizing Auxiliary-Variable Diffusion

We propose dealing with the issues of ignored latents and degenerate posteriors by using two regularization terms—a mutual information term and a prior regularizer. We refer to the resulting algorithm as InfoDiffusion.

Mutual Information Regularization To prevent the diffusion model from ignoring the latents $\mathbf{z}$, we augment the learning objective from Equation (4) with a mutual information term (Chen et al., 2016; Zhao et al., 2017) between $\mathbf{x}_0$ and $\mathbf{z}$ under $q_\phi(\mathbf{x}_0, \mathbf{z})$, the joint distribution over observed data $\mathbf{x}_0$ and latent variables $\mathbf{z}$. Formally, we define the mutual information regularizer as

$$\mathrm{MI}_{\mathbf{x}_0,\mathbf{z}} = \mathbb{E}_{q_\phi(\mathbf{x}_0,\mathbf{z})}\left[\log \frac{q_\phi(\mathbf{x}_0,\mathbf{z})}{q(\mathbf{x}_0) q_\phi(\mathbf{z})}\right]$$

where $q_\phi(\mathbf{z})$ is the marginal approximate posterior distribution—defined as the marginal of the product $q_\phi(\mathbf{z} | \mathbf{x}_0) q(\mathbf{x}_0)$. Intuitively, maximizing mutual information encourages the model to generate $\mathbf{x}_0$ from which we can predict $\mathbf{z}$.

Prior Regularization To prevent the model from learning a degenerate approximate posterior, we regularize the encoded samples $\mathbf{z}$ to look like the prior p. Formally, we define the prior regularizer as

$$\mathcal{R} = \mathrm{D}(q_\phi(\mathbf{z}) || p(\mathbf{z})),$$

where D is any strict divergence.

5.2. A Tractable Objective for InfoDiffusion

We train InfoDiffusion by maximizing a regularized ELBO objective of the form

$$\mathbb{E}_{q(\mathbf{x}_0)}[\mathcal{L}_D(\mathbf{x}_0)] + \zeta \cdot \mathrm{MI}_{\mathbf{x}_0,\mathbf{z}} - \beta \cdot \mathcal{R}, \tag{5}$$

where $\mathcal{L}_D(\mathbf{x}_0)$ is from Equation (4), and $\zeta, \beta > 0$ are scalars controlling the strength of the regularizers.

However, both the mutual information and the prior regularizer are intractable. Following Zhao et al. (2017), we rewrite the above learning objective into an equivalent tractable form, as described in Proposition 5.1 (see Appendix A for the full derivation). Defining $\lambda := \beta - 1$, we have

Proposition 5.1. *The regularized InfoDiffusion objective, Equation (5), can be rewritten as*

$$\begin{aligned}
\mathcal{L}_I = &\mathbb{E}_{q(\mathbf{x}_0,\mathbf{x}_1)}[\mathbb{E}_{q_\mathbf{z}}[\log p_\theta(\mathbf{x}_0|\mathbf{x}_1,\mathbf{z})]] \\
&- \mathbb{E}_{q(\mathbf{x}_0)}[KL(q(\mathbf{x}_T|\mathbf{x}_0)||p(\mathbf{x}_T))] \\
&- \sum_{t=2}^{T} \mathbb{E}_{q(\mathbf{x}_0,\mathbf{x}_t)}[\mathbb{E}_{q_\mathbf{z}}[KL(q(\mathbf{x}_{t-1}|\mathbf{x}_t,\mathbf{x}_0)||p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t,\mathbf{z}))]] \\
&- (1-\zeta)\mathbb{E}_{q(\mathbf{x}_0)}[KL(q_\phi(\mathbf{z}|\mathbf{x}_0)||p(\mathbf{z}))] \\
&- (\lambda+\zeta-1) KL(q_\phi(\mathbf{z})||p(\mathbf{z}))
\end{aligned} \tag{6}$$

We now state that $\mathrm{KL}(q_\phi(\mathbf{z})||p(\mathbf{z}))$ from Equation (6) can be replaced with any strict divergence $\mathrm{D}(q_\phi(\mathbf{z})||p(\mathbf{z}))$ without modifying the original objective in Proposition 5.2 (see Appendix B for the full derivation).

Proposition 5.2. *The term $KL(q_\phi(\mathbf{z})||p(\mathbf{z}))$ in Proposition 5.1 can be replaced with any strict divergence term $D(q_\phi(\mathbf{z})||p(\mathbf{z}))$ and meanwhile the InfoDiffusion objective $\mathcal{L}_I$ is guaranteed to be globally optimized for any fixed value I_0 of $MI_{\mathbf{x}_0,\mathbf{z}}$ when input space $\mathcal{X}_0$ and feature space $\mathcal{Z}$ are continuous spaces, $\zeta \leq 1$, $\lambda \geq 0$, if $p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t,\mathbf{z}) = q(\mathbf{x}_{t-1}|\mathbf{x}_t,\mathbf{x}_0)$ and $q_\phi(\mathbf{z}) = p(\mathbf{z})$.*

Thus, there are a range of divergences that can be compatible with our framework. In our experiments, we consider the maximum mean discrepancy (MMD) (Gretton et al., 2012), defined as:

$$\begin{aligned}
\mathrm{MMD}(q_\phi(\mathbf{z})||p(\mathbf{z})) = &\mathbb{E}_{\mathbf{z},\mathbf{z}' \sim q_\phi(\mathbf{z})}[k(\mathbf{z},\mathbf{z}')] \\
&+ \mathbb{E}_{\mathbf{z},\mathbf{z}' \sim p(\mathbf{z})}[k(\mathbf{z},\mathbf{z}')] \\
&- 2\mathbb{E}_{\mathbf{z} \sim q_\phi(\mathbf{z}), \mathbf{z}' \sim p(\mathbf{z})}[k(\mathbf{z},\mathbf{z}')]
\end{aligned}$$

Table 1. Comparison of InfoDiffusion model to other auto-encoder (*top*) and diffusion (*bottom*) frameworks in terms of enabling semantic latents, discrete latents, custom priors, mutual information maximization (Max MI), and high-quality sample generation (HQ samples).

	Semantic latents	Discrete latents	Custom prior	Max MI	HQ samples
AE	✖	✖	✖	✖	✖
VAE	✔	✔	✔	✖	✖
β-VAE	✔	✔	✔	✔	✖
AAE	✔	✖	✔	✔	✖
InfoVAE	✔	✔	✔	✔	✖
DDPM	✖	✖	✖	✖	✔
DiffAE	✔	✖	✖	✖	✔
InfoDiff	✔	✔	✔	✔	✔

where k is a positive definite kernel. In order to optimize $\mathrm{MMD}(q_\phi(\mathbf{z})||p(\mathbf{z}))$, we use sample-based optimization methods for implicit models. Specifically we estimate expectations over $q_\phi(\mathbf{z})$ by taking empirical averages over samples $\{\mathbf{x}_0^{(i)}\}_{i=1}^N \sim q(\mathbf{x}_0)$.

5.3. Comparing InfoDiffusion to Existing Models

The InfoDiffusion algorithm generalizes several existing methods in the literature. When the decoder performs one step of diffusion ($T=1$), we recover a model that is equivalent to the InfoVAE model (Zhao et al., 2017), up to choices of the decoder architecture. When we additionally choose $\lambda=0$, we recover the β-VAE model (Higgins et al., 2017). When $T=1$ and D is the Jensen-Shannon divergence, we recover adversarial auto-encoders (AAEs) (Makhzani et al., 2015). Our InfoDiffusion method can be seen as an extension of β-VAE, InfoVAE, and AAE to diffusion decoders, similar to how denoising diffusion probabilistic models (DDPM; Ho et al. (2020)) extend VAEs. Finally, when $\zeta=\lambda=0$, we recover the DiffAE model (Preechakul et al., 2022). We further discuss how our method relates to these prior works in Section 7. In Table 1, we detail this comparison to special cases.

6. Experiments

In this section, we evaluate our proposed method by comparing it to several baselines, using metrics that span generation quality, utility of latent space representations, and disentanglement. The baselines we compare against are: a vanilla auto-encoder (AE) (LeCun, 1987), a VAE (Kingma & Welling, 2013; Higgins et al., 2017), an InfoVAE (Zhao et al., 2017), and a DiffAE (Preechakul et al., 2022).

We measure performance on the following datasets: FashionMNIST (Xiao et al., 2017), CIFAR10 (Krizhevsky et al., 2009), FFHQ (Karras et al., 2019), CelebA (Liu et al., 2015), and 3DShapes (Burgess & Kim, 2018). See Appendix C for complete hyperparameter and computational resource details, by dataset.

As discussed in Section 4.4, for InfoDiffusion, we experiment with generating images using either $\mathbf{z}$ drawn from the prior or drawn from a learned latent distribution (denoted as "w/Learned Latent" in Table 2 and Table 3, see Appendix D.2 for details).

6.1. Exploring Latent Representations

We start by exploring three qualitative desirable features of learned representations: (1) their ability to capture high level semantic content, (2) smooth interpolation in latent space translating to smooth changes in generated output, and (3) their utility in downstream tasks.

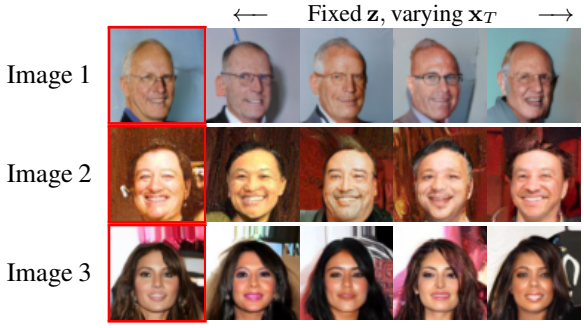


Figure 3. $\mathbf{z}$ captures high-level semantic detail. Varying $\mathbf{x}_T \sim \mathcal{N}(0,1)$ (across the columns in each row) changes lower level detail in the image. Red box indicates original image.

Auxiliary Variables Capture Semantic Information In Figure 3, we demonstrate that our model is able to encode high-level semantic information in the auxiliary variable. For a fixed $\mathbf{z}$ and varying $\mathbf{x}_T$, we find that decoded images change in their low-level features, e.g., background, hair style.

Latent Space Interpolation We begin with two images $\mathbf{x}_0^{(i)}, \mathbf{x}_0^{(j)}$ and retrieve their corresponding noise and auxiliary latent encodings $(\mathbf{z}^{(i)}, \mathbf{x}_T^{(i)}), (\mathbf{z}^{(j)}, \mathbf{x}_T^{(j)})$. Then, for 10 fixed steps $l \in [0,1]$, we generate images from the latent representations $(\mathbf{z}^l, \mathbf{x}_T^l)$ where $\mathbf{z}^l = \cos(l\pi/2)\mathbf{z}^{(i)} + \sin(l\pi/2)\mathbf{z}^{(j)}$ and $\mathbf{x}_T^l = \sin((1-l)\psi)\mathbf{x}_T^{(i)} + \sin(l\psi)\mathbf{x}_T^{(j)}$ are spherical interpolations between the auxiliary latent representation and noise tensors of the two images, with π denoting the angle between $\mathbf{z}^{(i)}$ and $\mathbf{z}^{(j)}$ and ψ the angle between $\mathbf{x}_T^{(i)}$ and $\mathbf{x}_T^{(j)}$. In Figure 5, we see that our model is able to combine the smooth interpolation of variational methods with the high sample quality of diffusion models.

Latent Variables Discover and Predict Class Labels In addition to the qualitative inspection of our latent space,

Figure 4. Finding disentangled dimensions in InfoDiffusion's auxiliary latent variable $\mathbf{z}$. Images are produced through a linear traversal along a particular dimension, spanning values from -1.5 to 1.5.

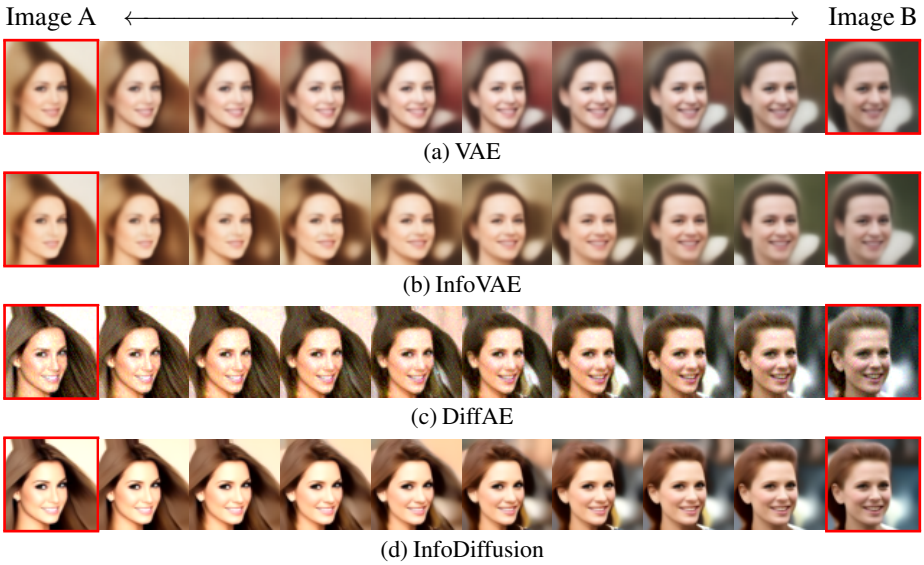


Figure 5. Latent space interpolation for relevant baselines (a-c) and InfoDiffusion (d). InfoDiffusion has a smooth latent space and maintains high image generation quality. Reconstructions of the original images two different images are on the left and right ends of each row and are marked by red boxes.

we run downstream classification tasks on $\mathbf{z}$ to measure its utility, which we report in Table 2 and Table 3 as "Latent Qual." Specifically, we train a logistic regression classifier on the auxiliary latent encodings of images to predict labels and report the accuracy/AUROC (or average accuracy/AUROC if multiple annotations are predicted) on a test set. We split the data into 80% training and 20% test, fit the classifier on the training data, and evaluate on the test set. We repeat this 5-fold and report mean metrics $\pm$ one standard deviation. We also compute FID based on five random sample sets of 10,000 images to obtain mean and standard deviation.

Across datasets, we consistently see that the compact latent representations from our models are most informative of labels. In addition to the utility of the latent space, we generate high-quality images.

6.2. Disentanglement

6.2.1. Finding disentangled dimensions

We find that maximizing mutual information in the InfoDiffusion objective yields disentangled components of our latent representations. For example, in Figure 1, we see several examples of disentangled factors. In Figure 4, we demonstrate this in more detail, traversing a specific dimension of $\mathbf{z}$ that controls smiling from values of -1.5 to 1.5.

6.2.2. Disentanglement Metrics

DCI Score For the 3DShapes dataset, we use the Disentanglement term of the DCI scores proposed in Eastwood & Williams (2018). This disentanglement metric is calculated as follows: for each attribute, a model is trained to predict it using the auxiliary latent vector $\mathbf{z}$. The model must also provide the importance of each dimension of $\mathbf{z}$ in predicting each attribute. Relative importance weights are converted to probabilities that dimension i of $\mathbf{z}$ is important for predicting a given label. The disentanglement score for each dimension of $\mathbf{z}$ is calculated as 1 minus the entropy of the relative importance probabilities. If a dimension is important for predicting only a single attribute, the score will be 1. If a dimension is equally important for predicting all attributes, the disentanglement score will be 0. The disentanglement scores are then averaged, with weights determined by the relative importance

Table 2. Latent quality, as measured by classification accuracies for logistic regression classifiers trained on the auxiliary latent vector $\mathbf{z}$, and FID. We report mean $\pm$ one standard deviation. Darkly shaded cells indicate the best while lightly shaded cells indicate the second best. See Table 10 for the performance of varying hyperameters.

	FASHIONMNIST		CIFAR10		FFHQ	
	LATENT QUAL. ↑	FID ↓	LATENT QUAL. ↑	FID ↓	LATENT QUAL. ↑	FID ↓
AE	0.819±0.003	62.9±2.1	0.336±0.005	169.4±2.4	0.615±0.002	92.3±2.7
VAE	0.796±0.002	63.4±1.6	0.342±0.004	177.2±3.2	**0.622±0.002**	95.4±2.4
BETA-VAE	0.779±0.004	66.9±1.8	0.253±0.003	183.3±3.1	0.588±0.002	99.7±3.4
INFOVAE	0.807±0.003	55.0±1.7	0.357±0.005	160.7±2.5	0.613±0.002	86.9±2.2
DIFFAE	0.835±0.002	8.2±0.3	0.395±0.006	32.1±1.1	0.608±0.001	31.6±1.2
INFODIFFUSION ($\lambda=0.1,\zeta=1$)	**0.839±0.003**	8.5±0.3	**0.412±0.003**	31.7±1.2	0.609±0.002	31.2±1.6
W/LEARNED LATENT		**7.4±0.2**		**31.5±1.8**		**30.9±2.5**

of each dimension across $\mathbf{z}$, to get the DCI disentanglement score. In Table 3, we see that for the 3DShapes dataset, InfoDiffusion attains the highest DCI disentanglement scores.

TAD For the CelebA dataset, we quantify disentanglement using TAD (Yeats et al., 2022), which is a disentanglement metric specifically proposed for this dataset that accounts for the presence of correlated and imbalanced attributes. First, we quantify attribute correlation by calculating the proportion of entropy reduction of each attribute given any other single attribute. Any attribute with an entropy reduction greater than 0.2 is removed. For each remaining attribute, we calculate AUROC score of each dimension of the auxiliary latent vector $\mathbf{z}$ in detecting that attribute. If an attribute can be detected by at least one dimension of $\mathbf{z}$, i.e., AUROC ≥ 0.75, it is considered to be "captured." The TAD score is the summation of the differences of the AUROC between the two most predictive latent dimensions for all captured attributes. In Table 3, we again see that InfoDiffusion has the best disentanglement performance with more captured attributes and higher TAD scores. We additionally note that the InfoDiffusion model balances disentanglement with high-quality generation and good latent space quality.

For calculating DCI on 3DShapes, we follow previous work (Locatello et al., 2019) and treat the attributes as discrete variables, using a gradient boosting classifier implemented by `scikit-learn` (Pedregosa et al., 2011) as our predictor. For disentanglement metric calculation, we split the data into 80% training and 20% test, fit the classifier on the training data, and calculate AUROC on the test data. We repeat this for 5-folds and report mean metrics $\pm$ one standard deviation.

6.3. Discrete Latent Priors

We demonstrate the flexibility of our model by training with a relaxed discrete prior. We train InfoDiffusion with a Relaxed Bernoulli prior (Jang et al., 2016) on the CelebA dataset and find that latent space quality is comparable to other models, with average AUROC of 0.73 (details in Appendix G).

6.4. Comparison to Contrastive Methods

We compare the quality of our learned representations to those from established contrastive learning methods, including SimCLR (Chen et al., 2020a), MOCO-v2 (Chen et al., 2020b), and DINO (Caron et al., 2021). In Table 4, we report average AUROC for classifiers trained on $\mathbf{z}$ to predict CelebA annotations and the TAD scores for disentanglement[1]. Our findings indicate that our latent representations are comparable, and in some instances superior, to these robust baselines. Our approach also has the added benefit of being a generative model. We also note that our model uses a much smaller capacity latent variable compared to these contrastive method baselines.

When comparing to methods with similar latent dimension, InfoDiffusion is able to significantly outperform baseline models. In Table 5, we compare to a fine-tuned, pre-trained encoder of SIMCLR with an additional dense layer that projects to 32 dimensions. We also introduce an another baseline, PDAE (Zhang et al., 2022), which builds an autoencoder based on pre-trained diffusion models. Our method outperforms these alternatives on both the disentanglement and latent quality metrics.

6.5. Exploring InfoDiffusion Modeling Choices

Regularization Coefficients An evaluation of various ζ and λ parameters for InfoDiffusion is presented in Appendix H. We find that prioritizing information maximization improves both generation quality and latent space coherence, with better performance achieved by maintaining a constant λ and increasing ζ. However, assigning ζ values greater

[1]We excluded the "Number of attributes captured" metric for this comparison, as the pre-trained contrastive method baselines use larger latent dimension, which artificially inflates the value for this metric.

Table 3. Disentanglement and latent quality metrics and FID. For 3DShapes, we check the image quality manually and label the models which generate high-quality images with check marks ('Image Qual.'). The visualization of the samples is shown in Figure 9 in Appendix I. For CelebA, 'Attrs.' counts the number of "captured" attributes when calculating the TAD score. 'Latent Quality' is measured as AUROC scores averaged across attributes for logistic regression classifiers trained on the auxiliary latent vector $\mathbf{z}$. We report means $\pm$ one standard deviation for quantitative metrics. Darkly shaded cells indicate the best while lightly shaded cells indicate the second best.

	3DSHAPES		CELEBA			
	DCI ↑	IMAGE QUAL.	TAD↑	ATTRS↑	LATENT QUAL. ↑	FID ↓
AE	0.219±0.001	✖	0.042±0.004	1.0±0.0	0.759±0.003	90.4±1.8
VAE	0.276±0.001	✖	0.000±0.000	0.0±0.0	0.770±0.002	94.3±2.8
BETA-VAE	0.281±0.001	✖	0.088±0.051	1.6±0.8	0.699±0.001	99.8±2.4
INFOVAE	0.134±0.001	✖	0.000±0.000	0.0±0.0	0.757±0.003	77.8±1.6
DIFFAE	0.196±0.001	✔	0.155±0.010	2.0±0.0	0.799±0.002	22.7±2.1
INFODIFFUSION ($\lambda=0.1,\zeta=1$)	0.109±0.001	✔	0.192±0.004	2.8±0.4	**0.848±0.001**	23.8±1.6
W/LEARNED LATENT		✔				**21.2±2.4**
INFODIFFUSION ($\lambda=0.01,\zeta=1$)	**0.342±0.002**	✔	**0.299±0.006**	**3.0±0.0**	0.836±0.002	23.6±1.3
W/LEARNED LATENT		✔				22.3±1.2

Table 4. Representation learning comparison to contrastive methods. 'Gen.' indicates whether the model has generative capabilities. 'Dim.' denotes the latent dimension. Disentanglement is measured by TAD. 'Latent Quality' is measured as AUROC scores averaged across CelebA attributes for logistic regression classifiers trained on latent representations. We report means $\pm$ one standard deviation for quantitative metrics. Darkly shaded cells indicate the best while lightly shaded cells indicate the second best. † denotes that the weights are taken from the PyTorch repository for the method.

CELEBA	GEN.	DIM.	TAD ↑	LATENT QUAL. ↑
SIMCLR†	✖	2048	0.192±0.015	0.812±0.003
MOCO-V2†	✖	2048	0.279±0.025	**0.846±0.001**
DINO†	✖	384	0.000±0.000	0.592±0.003
INFODIFFUSION	✔	32	**0.299±0.006**	0.836±0.002

Table 5. Representation learning comparison to SIMCLR and PDAE with 32-dimensional latents. 'Gen.' indicates whether the model has generative capabilities. 'Attrs.' counts the number of "captured" attributes when calculating the TAD score. 'Latent Quality' is measured as AUROC scores averaged across attributes for logistic regression classifiers trained on $\mathbf{z}$. We report means $\pm$ one standard deviation for quantitative metrics. Darkly shaded cells indicate the best while lightly shaded cells indicate the second best. * denotes that the weights are taken from the PyTorch repository and fine-tuned with an added dense layer. ‡ denotes that the model is re-trained using the codebase provided by this baseline.

CELEBA	GEN.	TAD ↑	ATTRS ↑	LATENT QUAL. ↑
SIMCLR*	✖	0.062±0.005	2.6±0.5	0.757±0.002
PDAE‡	✔	0.009±0.001	1.0±0.0	0.767±0.003
INFODIFF.	✔	**0.299±0.006**	**3.0±0.0**	**0.836±0.002**

than 1 results in instability in the KL divergence term; thus, we cap $\zeta = 1$ for optimal performance. For $\zeta = 1$, we find that our model is robust to the choice of λ, however for the natural image datasets, the optimal setting is $\lambda = 0.1$.

Sampling Method Table 2 and Table 3 provide results for generation using samples extracted from either the prior distribution or a learned latent distribution, as denoted in the "w/Learned Latent" rows. As opposed to DiffAE, which necessitates a latent diffusion model for effective sampling, our model can generate high-quality images using unconditional draws from a prior.

7. Related Work

7.1. Representation Learning in Generative Modeling

VAEs (Kingma & Welling, 2013; Higgins et al., 2017) extend the auto-encoder framework through variational inference algorithms to produce a generative model with semantically meaningful and smooth latent spaces. InfoVAE (Zhao et al., 2017) solves a key failure mode of VAEs through mutual information regularization to improve the quality of the variational posterior. Another paradigm, known as Infogan (Chen et al., 2016), extends generative adversarial networks (GANs; Goodfellow et al. (2020)) by similarly using information maximization. Our approach has the advantage of combining the stable training and generation quality of diffusion models with the representation learning capabilities of these prior works.

7.2. Diffusion Models for Representation Learning

Our work builds upon advances in diffusion models, which enable stable, high-resolution training on varied datasets (Dhariwal & Nichol, 2021; Ho et al., 2020; Saharia et al., 2022; Rombach et al., 2021). Recent work has combined auto-encoders with diffusion models—e.g., DiffAE (Preechakul et al., 2022), a non-probabilistic auto-encoder

Table 6. Analogy between progress in the space of auto-encoders and similar progress for diffusion models.

Method	Non-Probabilistic	Probabilistic Extension	Regularized Extension
Auto-encoders	AE (LeCun, 1987)	VAE (Kingma & Welling, 2013)	InfoVAE (Zhao et al., 2017)
Diffusion models	DiffAE (Preechakul et al., 2022)	Variational Auxiliary-Variable Diffusion Sec. 4	InfoDiffusion Sec. 5

model that produces semantically meaningful latents.

The relationship between our method and DiffAE is analogous to the relationship between InfoVAE (Zhao et al., 2017) and a regular non-probabilistic auto-encoder. Our method augments DiffAE with: (1) a principled probabilistic auxiliary-variable model family and (2) new learning objectives based on variational mutual information maximization. This yields a number of advantages. First, our method allows users to specify domain knowledge through a prior and supports the use of discrete variables. Additionally, our improved objective maximizes mutual information, which empirically yields more useful and disentangled latents.

Table 6 illustrates how our approach relates to previous work on both diffusion models and mutual information regularization by showing an analogy between progress in the space of auto-encoders and similar progress for diffusion models.

8. Conclusion

In this work, we proposed InfoDiffusion, a new learning algorithm based on a diffusion model that uses an auxiliary variable to encode semantically meaningful information. We derive InfoDiffusion from a principled probabilistic extension of diffusion models that relies on variational inference to discover low-dimensional latents. Augmenting this variational auxiliary-variable diffusion framework with mutual information regularization enables InfoDiffusion to simultaneously achieve high-quality sample generation *and* informative latent representations, which we use to control generation and improve downstream prediction.

We evaluate InfoDiffusion on several image datasets and against state-of-the-art generative and representation learning baselines and show that it consistently produces semantically rich and more disentangled latent representations and high-quality images. We expect InfoDiffusion will be useful in generative design and other applications that require both exploring a latent space and quality generation.

Acknowledgements

This work was supported by Tata Consulting Services, the Presidential Life Science Fellowship, the Hal & Inge Marcus PhD Fellowship, and NSF CAREER grants (#1750326, #2046760, and #2145577).

References

Burgess, C. and Kim, H. 3d shapes dataset. https://github.com/deepmind/3dshapes-dataset/, 2018.

Caron, M., Touvron, H., Misra, I., Jégou, H., Mairal, J., Bojanowski, P., and Joulin, A. Emerging properties in self-supervised vision transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 9650–9660, 2021.

Chen, T., Kornblith, S., Norouzi, M., and Hinton, G. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, pp. 1597–1607. PMLR, 2020a.

Chen, X., Duan, Y., Houthooft, R., Schulman, J., Sutskever, I., and Abbeel, P. Infogan: Interpretable representation learning by information maximizing generative adversarial nets. *Advances in neural information processing systems*, 29, 2016.

Chen, X., Fan, H., Girshick, R., and He, K. Improved baselines with momentum contrastive learning. *arXiv preprint arXiv:2003.04297*, 2020b.

Dhariwal, P. and Nichol, A. Diffusion models beat gans on image synthesis. *Advances in Neural Information Processing Systems*, 34:8780–8794, 2021.

Eastwood, C. and Williams, C. K. A framework for the quantitative evaluation of disentangled representations. In *International Conference on Learning Representations*, 2018.

Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., and Bengio, Y. Generative adversarial networks. *Communications of the ACM*, 63(11):139–144, 2020.

Gretton, A., Borgwardt, K. M., Rasch, M. J., Schölkopf, B., and Smola, A. A kernel two-sample test. *The Journal of Machine Learning Research*, 13(1):723–773, 2012.

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.

Ho, J., Jain, A., and Abbeel, P. Denoising diffusion probabilistic models. *Advances in Neural Information Processing Systems*, 33:6840–6851, 2020.

Jang, E., Gu, S., and Poole, B. Categorical reparameterization with gumbel-softmax. *arXiv preprint arXiv:1611.01144*, 2016.

Jing, B., Corso, G., Chang, J., Barzilay, R., and Jaakkola, T. Torsional diffusion for molecular conformer generation. *arXiv preprint arXiv:2206.01729*, 2022.

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.

Kingma, D. P. and Welling, M. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013.

Kong, Z., Ping, W., Huang, J., Zhao, K., and Catanzaro, B. Diffwave: A versatile diffusion model for audio synthesis. *arXiv preprint arXiv:2009.09761*, 2020.

Krizhevsky, A., Hinton, G., et al. Learning multiple layers of features from tiny images. 2009.

LeCun, Y. Phd thesis: Modeles connexionnistes de l'apprentissage (connectionist learning models). 1987.

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.

Locatello, F., Bauer, S., Lucic, M., Raetsch, G., Gelly, S., Schölkopf, B., and Bachem, O. Challenging common assumptions in the unsupervised learning of disentangled representations. In *international conference on machine learning*, pp. 4114–4124. PMLR, 2019.

Makhzani, A., Shlens, J., Jaitly, N., Goodfellow, I., and Frey, B. Adversarial autoencoders. *arXiv preprint arXiv:1511.05644*, 2015.

Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., Desmaison, A., Kopf, A., Yang, E., DeVito, Z., Raison, M., Tejani, A., Chilamkurthy, S., Steiner, B., Fang, L., Bai, J., and Chintala, S. Pytorch: An imperative style, high-performance deep learning library. In Wallach, H., Larochelle, H., Beygelzimer, A., d'Alché Buc, F., Fox, E., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems 32*, pp. 8024–8035. Curran Associates, Inc., 2019.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., and Duchesnay, E. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

Preechakul, K., Chatthee, N., Wizadwongsa, S., and Suwajanakorn, S. Diffusion autoencoders: Toward a meaningful and decodable representation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 10619–10629, 2022.

Ramesh, A., Dhariwal, P., Nichol, A., Chu, C., and Chen, M. Hierarchical text-conditional image generation with clip latents. *arXiv preprint arXiv:2204.06125*, 2022.

Rombach, R., Blattmann, A., Lorenz, D., Esser, P., and Ommer, B. High-resolution image synthesis with latent diffusion models, 2021.

Ronneberger, O., Fischer, P., and Brox, T. U-net: Convolutional networks for biomedical image segmentation. In *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2015: 18th International Conference, Munich, Germany, October 5-9, 2015, Proceedings, Part III 18*, pp. 234–241. Springer, 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., et al. Photorealistic text-to-image diffusion models with deep language understanding. *arXiv preprint arXiv:2205.11487*, 2022.

Xiao, H., Rasul, K., and Vollgraf, R. Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms. *arXiv preprint arXiv:1708.07747*, 2017.

Xu, M., Yu, L., Song, Y., Shi, C., Ermon, S., and Tang, J. Geodiff: A geometric diffusion model for molecular conformation generation. *arXiv preprint arXiv:2203.02923*, 2022.

Yang, L., Zhang, Z., Song, Y., Hong, S., Xu, R., Zhao, Y., Shao, Y., Zhang, W., Cui, B., and Yang, M.-H. Diffusion models: A comprehensive survey of methods and applications. *arXiv preprint arXiv:2209.00796*, 2022.

Yeats, E., Liu, F., Womble, D., and Li, H. Nashae: Disentangling representations through adversarial covariance minimization. In *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XXVII*, pp. 36–51. Springer, 2022.

Zhang, Z., Zhao, Z., and Lin, Z. Unsupervised representation learning from pre-trained diffusion probabilistic models. *Advances in Neural Information Processing Systems*, 35: 22117–22130, 2022.

Zhao, S., Song, J., and Ermon, S. Infovae: Information maximizing variational autoencoders. *arXiv preprint arXiv:1706.02262*, 2017.

A. Proof of Proposition 5.1

We start with the derivation for the ELBO of a Variational Auxiliary-Variable Diffusion Model defined in Equation (4):

$$
\begin{aligned}
\log p(\mathbf{x}_0) &= \log \int p(\mathbf{x}_{0:T}, \mathbf{z}) d\mathbf{x}_{1:T} d\mathbf{z} \\
&= \log \int \frac{p(\mathbf{x}_{0:T}, \mathbf{z}) q(\mathbf{x}_{1:T} | \mathbf{x}_0) q_\phi(\mathbf{z} | \mathbf{x}_0)}{q(\mathbf{x}_{1:T} | \mathbf{x}_0) q_\phi(\mathbf{z} | \mathbf{x}_0)} d\mathbf{x}_{1:T} d\mathbf{z} \\
&= \log \mathbb{E}_{q(\mathbf{x}_{1:T}|\mathbf{x}_0)} \left[\mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x}_0)} \left[\frac{p(\mathbf{x}_{0:T}, \mathbf{z})}{q(\mathbf{x}_{1:T} | \mathbf{x}_0)) q_\phi(\mathbf{z} | \mathbf{x}_0))} \right] \right] \\
&\geq \mathbb{E}_{q(\mathbf{x}_{1:T}|\mathbf{x}_0)} \left[\mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x}_0)} \left[\log \frac{p(\mathbf{x}_{0:T}, \mathbf{z})}{q(\mathbf{x}_{1:T} | \mathbf{x}_0)) q_\phi(\mathbf{z} | \mathbf{x}_0))} \right] \right] \\
&= \mathbb{E}_{q(\mathbf{x}_{1:T}|\mathbf{x}_0)} \left[\mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x}_0)} \left[\log \frac{p(\mathbf{z}) p(\mathbf{x}_T) \prod_{t=1}^T p(\mathbf{x}_{t-1}|\mathbf{x}_t, \mathbf{z})}{q_\phi(\mathbf{z} | \mathbf{x}_0) \prod_{t=1}^T q(\mathbf{x}_t | \mathbf{x}_{t-1})} \right] \right] \\
&= \mathbb{E}_{q(\mathbf{x}_{1:T}|\mathbf{x}_0)} \left[\mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x}_0)} \left[\log \frac{p(\mathbf{z}) p(\mathbf{x}_T) p(\mathbf{x}_0|\mathbf{x}_1, \mathbf{z}) \prod_{t=2}^T p(\mathbf{x}_{t-1}|\mathbf{x}_t, \mathbf{z})}{q_\phi(\mathbf{z} | \mathbf{x}_0) q(\mathbf{x}_1|\mathbf{x}_0) \prod_{t=2}^T q(\mathbf{x}_t | \mathbf{x}_{t-1}, \mathbf{x}_0)} \right] \right] \\
&= \mathbb{E}_{q(\mathbf{x}_{1:T}|\mathbf{x}_0)} \left[\mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x}_0)} \left[\log \frac{p(\mathbf{z})}{q_\phi(\mathbf{z} | \mathbf{x}_0)} + \log \frac{p(\mathbf{x}_T) p(\mathbf{x}_0|\mathbf{x}_1, \mathbf{z})}{q(\mathbf{x}_1|\mathbf{x}_0)} + \sum_{t=2}^T \log \frac{p(\mathbf{x}_{t-1}|\mathbf{x}_t, \mathbf{z})}{q(\mathbf{x}_t | \mathbf{x}_{t-1}, \mathbf{x}_0)} \right] \right] \\
&= \mathbb{E}_{q(\mathbf{x}_{1:T}|\mathbf{x}_0)} \left[\mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x}_0)} \left[\log \frac{p(\mathbf{z})}{q_\phi(\mathbf{z} | \mathbf{x}_0)} + \log \frac{p(\mathbf{x}_T) p(\mathbf{x}_0|\mathbf{x}_1, \mathbf{z})}{q(\mathbf{x}_1|\mathbf{x}_0)} + \sum_{t=2}^T \log \frac{p(\mathbf{x}_{t-1}|\mathbf{x}_t, \mathbf{z})}{\frac{q(\mathbf{x}_{t-1}|\mathbf{x}_t, \mathbf{x}_0) q(\mathbf{x}_t|\mathbf{x}_0)}{q(\mathbf{x}_{t-1}|\mathbf{x}_0)}} \right] \right] \\
&= \mathbb{E}_{q(\mathbf{x}_{1:T}|\mathbf{x}_0)} \left[\mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x}_0)} \left[\log \frac{p(\mathbf{z})}{q_\phi(\mathbf{z} | \mathbf{x}_0)} + \log \frac{p(\mathbf{x}_T) p(\mathbf{x}_0|\mathbf{x}_1, \mathbf{z})}{q(\mathbf{x}_1|\mathbf{x}_0)} + \log \frac{q(\mathbf{x}_1|\mathbf{x}_0)}{q(\mathbf{x}_T|\mathbf{x}_0)} + \sum_{t=2}^T \log \frac{p(\mathbf{x}_{t-1}|\mathbf{x}_t, \mathbf{z})}{q(\mathbf{x}_{t-1}|\mathbf{x}_t, \mathbf{x}_0)} \right] \right] \\
&= \mathbb{E}_{q(\mathbf{x}_{1:T}|\mathbf{x}_0)} \left[\mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x}_0)} \left[\log \frac{p(\mathbf{z})}{q_\phi(\mathbf{z} | \mathbf{x}_0)} + \log \frac{p(\mathbf{x}_T) p(\mathbf{x}_0|\mathbf{x}_1, \mathbf{z})}{q(\mathbf{x}_T|\mathbf{x}_0)} + \sum_{t=2}^T \log \frac{p(\mathbf{x}_{t-1}|\mathbf{x}_t, \mathbf{z})}{q(\mathbf{x}_{t-1}|\mathbf{x}_t, \mathbf{x}_0)} \right] \right] \\
&= \mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x}_0)} \left[\log \frac{p(\mathbf{z})}{q_\phi(\mathbf{z} | \mathbf{x}_0)} \right] + \mathbb{E}_{q(\mathbf{x}_1|\mathbf{x}_0)} \left[\mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x}_0)} [\log p_\theta(\mathbf{x}_0|\mathbf{x}_1, \mathbf{z})] \right] \\
&\quad + \mathbb{E}_{q(\mathbf{x}_T|\mathbf{x}_0)} \left[\log \frac{p(\mathbf{x}_T)}{q(\mathbf{x}_T|\mathbf{x}_0)} \right] + \sum_{t=2}^T \mathbb{E}_{q(\mathbf{x}_{t-1}, \mathbf{x}_t|\mathbf{x}_0)} \left[\mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x}_0)} \left[\log \frac{p(\mathbf{x}_{t-1}|\mathbf{x}_t, \mathbf{z})}{q(\mathbf{x}_{t-1}|\mathbf{x}_t, \mathbf{x}_0)} \right] \right] \\
&= \mathbb{E}_{q(\mathbf{x}_1|\mathbf{x}_0)} \left[\mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x}_0)} [\log p(\mathbf{x}_0 | \mathbf{x}_1, \mathbf{z})] \right] - \mathrm{KL}(q(\mathbf{x}_T || \mathbf{x}_0) || p(\mathbf{x}_T)) - \mathrm{KL}(q_\phi(\mathbf{z} | \mathbf{x}_0) || p(\mathbf{z})) \\
&\quad - \sum_{t=2}^T \mathbb{E}_{q(\mathbf{x}_t|\mathbf{x}_0)} \left[\mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x}_0)} [\mathrm{KL}(q(\mathbf{x}_{t-1}|\mathbf{x}_t, \mathbf{x}_0) || p(\mathbf{x}_{t-1} | \mathbf{x}_t, \mathbf{z}))] \right].
\end{aligned} \tag{7}
$$

Averaging Equation (7) over the data distribution $q(\mathbf{x}_0)$, the prior matching term (the third term in Equation (7)) can be rewritten as:

$$
\begin{aligned}
-\mathbb{E}_{q(\mathbf{x}_0)} \mathrm{KL}(q_\phi(\mathbf{z}|\mathbf{x}_0) || p(\mathbf{z})) &= \mathbb{E}_{q(\mathbf{x}_0)} \left[\mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x}_0)} [\log p(\mathbf{z}) - \log q_\phi(\mathbf{z} | \mathbf{x}_0)] \right] \\
&= \mathbb{E}_{q_\phi(\mathbf{x}_0, \mathbf{z})} \left[\log \frac{p(\mathbf{z})}{q_\phi(\mathbf{x}_0, \mathbf{z})} + \log q(\mathbf{x}_0) \right] \\
&= \mathbb{E}_{q_\phi(\mathbf{x}_0, \mathbf{z})} \left[\log \frac{p(\mathbf{z})}{q_\phi(\mathbf{x}_0|\mathbf{z}) q_\phi(\mathbf{z})} + \log q(\mathbf{x}_0) \right] \\
&= \mathbb{E}_{q_\phi(\mathbf{x}_0, \mathbf{z})} \left[\log \frac{p(\mathbf{z})}{q_\phi(\mathbf{z})} + \log \frac{q(\mathbf{x}_0)}{q_\phi(\mathbf{x}_0|\mathbf{z})} \right] \\
&= \mathbb{E}_{q_\phi(\mathbf{x}_0, \mathbf{z})} \left[\log \frac{p(\mathbf{z})}{q_\phi(\mathbf{z})} + \log \frac{q_\phi(\mathbf{z})}{q_\phi(\mathbf{z}|\mathbf{x}_0)} \right]
\end{aligned}
$$

$$= -\mathrm{KL}(q_\phi(\mathbf{z})||p(\mathbf{z})) - \mathrm{MI}_{\mathbf{x}_0,\mathbf{z}}. \tag{8}$$

If we scale $\mathrm{KL}(q_\phi(\mathbf{z})||p(\mathbf{z}))$ by λ and add a scaled mutual information term between $\mathbf{x}_0$ and $\mathbf{z}$, $\zeta\mathrm{MI}_{\mathbf{x}_0,\mathbf{z}}$, Equation (8) becomes:

$$\begin{aligned}
&-\lambda\mathrm{KL}(q_\phi(\mathbf{z})||p(\mathbf{z})) - \mathrm{MI}_{\mathbf{x}_0,\mathbf{z}} + \zeta\mathrm{MI}_{\mathbf{x}_0,\mathbf{z}} && (9)\\
&= \mathbb{E}_{q_\phi(\mathbf{x}_0,\mathbf{z})}\left[-\lambda\log\frac{q_\phi(\mathbf{z})}{p(\mathbf{z})} - (\zeta-1)\log\frac{q_\phi(\mathbf{z})}{q_\phi(\mathbf{z}|\mathbf{x}_0)}\right]\\
&= \mathbb{E}_{q_\phi(\mathbf{x}_0,\mathbf{z})}\left[-\log\frac{q_\phi(\mathbf{z})^{\lambda+\zeta-1}q_\phi(\mathbf{z}|\mathbf{x}_0)^{1-\zeta}}{p(\mathbf{z})^{\lambda+\zeta-1}p(\mathbf{z})^{1-\zeta}}\right]\\
&= -(\lambda+\zeta-1)\mathrm{KL}(q_\phi(\mathbf{z})||p(\mathbf{z})) - (1-\zeta)\mathbb{E}_{q(\mathbf{x}_0)}\mathrm{KL}(q_\phi(\mathbf{z}|\mathbf{x}_0)||p(\mathbf{z})) && (10)
\end{aligned}$$

Replacing the prior regularization term in Equation (7) with Equation (10) and averaging the remaining terms in Equation (7) over the data distribution $q(\mathbf{x}_0)$, we have our InfoDiffusion ELBO objective as follows:

$$\begin{aligned}
\mathcal{L}_I =& \mathbb{E}_{q(\mathbf{x}_0,\mathbf{x}_1)}\left[\mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x}_0)}[\log p_\theta(\mathbf{x}_0|\mathbf{x}_1,\mathbf{z})]\right] - \mathbb{E}_{q(\mathbf{x}_0)}[\mathrm{KL}(q(\mathbf{x}_T|\mathbf{x}_0)||p(\mathbf{x}_T))]\\
&-\sum_{t=2}^{T}\mathbb{E}_{q(\mathbf{x}_0,\mathbf{x}_t)}\left[\mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x}_0)}[\mathrm{KL}(q(\mathbf{x}_{t-1}|\mathbf{x}_t,\mathbf{x}_0)||p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t,\mathbf{z}))]\right]\\
&-(\lambda+\zeta-1)\mathrm{KL}(q_\phi(\mathbf{z})||p(\mathbf{z})) - (1-\zeta)\mathbb{E}_{q(\mathbf{x}_0)}[\mathrm{KL}(q_\phi(\mathbf{z}|\mathbf{x}_0)||p(\mathbf{z}))] \qquad \square && (11)
\end{aligned}$$

We parameterize p_θ and q_ϕ with neural networks.

B. Proof of Proposition 5.2

Following Zhao et al. (2017), we first rewrite the $\mathcal{L}_I$ objective from Equation (11) with the following changes: (1) we replace the KL divergence between $q_\phi(\mathbf{z})$ and $p(\mathbf{z})$ with any strict divergence D, and (2) we expand the last term of Equation (11) into a KL divergence term and a mutual information term (as in Equation (8)):

$$\begin{aligned}
\mathcal{L}_I =& \mathbb{E}_{q(\mathbf{x}_0,\mathbf{x}_1)}\left[\mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x}_0)}[\log p_\theta(\mathbf{x}_0|\mathbf{x}_1,\mathbf{z})]\right] - \mathbb{E}_{q(\mathbf{x}_0)}\mathrm{KL}(q(\mathbf{x}_T|\mathbf{x}_0)||p(\mathbf{x}_T))\\
&-\sum_{t=2}^{T}\mathbb{E}_{q(\mathbf{x}_0,\mathbf{x}_t)}\left[\mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x}_0)}[\mathrm{KL}(q(\mathbf{x}_{t-1}|\mathbf{x}_t,\mathbf{x}_0)||p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t,\mathbf{z}))]\right]\\
&-(\lambda+\zeta-1)\mathrm{D}(q_\phi(\mathbf{z})||p(\mathbf{z})) - (1-\zeta)\mathrm{KL}(q_\phi(\mathbf{z})||p(\mathbf{z})) - (1-\zeta)\mathrm{MI}_{\mathbf{x}_0,\mathbf{z}}. && (12)
\end{aligned}$$

Note that restricting $\zeta \leq 1$ and $\lambda \geq 0$, we have $1-\zeta \geq 0$ and $\zeta+\lambda-1 \geq 0$. For convenience, we define

$$\begin{aligned}
\eta &:= 1-\zeta \ \geq 0\\
\gamma &:= \zeta+\lambda-1 \ \geq 0
\end{aligned}$$

Then, we consider the rewritten objective Equation (12) in two separate terms:

$$\begin{aligned}
\mathcal{L}_1 &= \mathbb{E}_{q(\mathbf{x}_0,\mathbf{x}_1)}\left[\mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x}_0)}[\log p_\theta(\mathbf{x}_0|\mathbf{x}_1,\mathbf{z})]\right] - \sum_{t=2}^{T}\mathbb{E}_{q(\mathbf{x}_0,\mathbf{x}_t)}\left[\mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x}_0)}[\mathrm{KL}(q(\mathbf{x}_{t-1}|\mathbf{x}_t,\mathbf{x}_0)||p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t,\mathbf{z}))]\right] - \eta\mathrm{MI}_{\mathbf{x}_0,\mathbf{z}}\\
\mathcal{L}_2 &= -\gamma\mathrm{D}(q_\phi(\mathbf{z})||p(\mathbf{z})) - \eta\mathrm{KL}(q_\phi(\mathbf{z})||p(\mathbf{z}))
\end{aligned}$$

We will demonstrate that the two terms are maximized according to the condition in the proposition, for any values of $\eta \geq 0$ and $\gamma \geq 0$. To begin, we examine $\mathcal{L}_1$, for some fixed value of $\mathrm{MI}_{\mathbf{x}_0,\mathbf{z}} = I_0$.

$$\mathcal{L}_1 = \mathbb{E}_{q(\mathbf{x}_0,\mathbf{x}_1)}\left[\mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x}_0)}[\log p_\theta(\mathbf{x}_0|\mathbf{x}_1,\mathbf{z})]\right] - \sum_{t=2}^{T}\mathbb{E}_{q(\mathbf{x}_0,\mathbf{x}_t)}\left[\mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x}_0)}[\mathrm{KL}(q(\mathbf{x}_{t-1}|\mathbf{x}_t,\mathbf{x}_0)||p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t,\mathbf{z}))]\right] - \eta I_0$$

$$
\begin{aligned}
&=\mathbb{E}_{q(\mathbf{x}_0,\mathbf{x}_1)}[\log q(\mathbf{x}_0,\mathbf{x}_1)]-\mathbb{E}_{q(\mathbf{x}_0,\mathbf{x}_1)}\left[\mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x}_0)}\left[\log\frac{q(\mathbf{x}_0,\mathbf{x}_1)}{p_\theta(\mathbf{x}_0|\mathbf{x}_1,\mathbf{z})}\right]\right]\\
&\quad-\sum_{t=2}^{T}\mathbb{E}_{q(\mathbf{x}_0,\mathbf{x}_t)}\left[\mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x}_0)}[\mathrm{KL}(q(\mathbf{x}_{t-1}|\mathbf{x}_t,\mathbf{x}_0)||p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t,\mathbf{z}))]\right]-\eta I_0\\
&=\mathbb{E}_{q(\mathbf{x}_0,\mathbf{x}_1)}[\log q(\mathbf{x}_0,\mathbf{x}_1)]-\mathbb{E}_{q(\mathbf{x}_0|\mathbf{x}_1)}\left[\mathbb{E}_{q(\mathbf{x}_1)}\left[\mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x}_0)}\left[\log\frac{q(\mathbf{x}_0\,|\,\mathbf{x}_1)q(\mathbf{x}_1)}{p_\theta(\mathbf{x}_0|\mathbf{x}_1,\mathbf{z})}\right]\right]\right]\\
&\quad-\sum_{t=2}^{T}\mathbb{E}_{q(\mathbf{x}_0,\mathbf{x}_t)}\left[\mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x}_0)}[\mathrm{KL}(q(\mathbf{x}_{t-1}|\mathbf{x}_t,\mathbf{x}_0)||p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t,\mathbf{z}))]\right]-\eta I_0\\
&=\mathbb{E}_{q(\mathbf{x}_0,\mathbf{x}_1)}[\log q(\mathbf{x}_0,\mathbf{x}_1)]-\mathbb{E}_{q(\mathbf{x}_1)}[\log q(\mathbf{x}_1)]-\mathbb{E}_{q(\mathbf{x}_1)}\left[\mathbb{E}_{q(\mathbf{x}_0|\mathbf{x}_1)}\left[\mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x}_0)}\left[\log\frac{q(\mathbf{x}_0\,|\,\mathbf{x}_1)}{p_\theta(\mathbf{x}_0|\mathbf{x}_1,\mathbf{z})}\right]\right]\right]\\
&\quad-\sum_{t=2}^{T}\mathbb{E}_{q(\mathbf{x}_0,\mathbf{x}_t)}\left[\mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x}_0)}[\mathrm{KL}(q(\mathbf{x}_{t-1}|\mathbf{x}_t,\mathbf{x}_0)||p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t,\mathbf{z}))]\right]-\eta I_0\\
&=\mathbb{E}_{q(\mathbf{x}_0,\mathbf{x}_1)}[\log q(\mathbf{x}_0,\mathbf{x}_1)]-\mathbb{E}_{q(\mathbf{x}_1)}[\log q(\mathbf{x}_1)]-\mathbb{E}_{q(\mathbf{x}_1)}\left[\mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x}_0)}[\mathrm{KL}(q(\mathbf{x}_0\,|\,\mathbf{x}_1)||p_\theta(\mathbf{x}_0|\mathbf{x}_1,\mathbf{z}))]\right]\\
&\quad-\sum_{t=2}^{T}\mathbb{E}_{q(\mathbf{x}_0,\mathbf{x}_t)}\left[\mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x}_0)}[\mathrm{KL}(q(\mathbf{x}_{t-1}|\mathbf{x}_t,\mathbf{x}_0)||p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t,\mathbf{z}))]\right]-\eta I_0
\end{aligned}
$$

For any $p_\theta(\mathbf{x}_{t-1}\mid\mathbf{x}_t,\mathbf{z})$ that optimizes $\mathcal{L}_1$ we have that $\forall\mathbf{z}$, $p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t,\mathbf{z})=q(\mathbf{x}_{t-1}|\mathbf{x}_t,\mathbf{x}_0)$ if $t\geq 2$, and $q(\mathbf{x}_0\,|\,\mathbf{x}_1)=p_\theta(\mathbf{x}_0|\mathbf{x}_1,\mathbf{z})$, then for a fixed value of $\mathrm{MI}_{\mathbf{x}_0,\mathbf{z}}$, the optimal $\mathcal{L}_1$ is

$$
\begin{aligned}
\mathcal{L}_1^*&=\mathbb{E}_{q(\mathbf{x}_0,\mathbf{x}_1)}[\log q(\mathbf{x}_0,\mathbf{x}_1)]-\mathbb{E}_{q(\mathbf{x}_1)}[\log q(\mathbf{x}_1)]-\eta I_0\\
&=-H_q(\mathbf{x}_0,\mathbf{x}_1)+H_q(\mathbf{x}_1)-\eta I_0
\end{aligned}
$$

where we use $H_q(\mathbf{x}_0,\mathbf{x}_1)$ and $H_q(\mathbf{x}_1)$ to denote the entropy of $q(\mathbf{x}_0,\mathbf{x}_1)$ and $q(\mathbf{x}_1)$, respectively. So we only have to independently maximize $\mathcal{L}_2$, subject to fixed some fixed $\mathrm{MI}_{\mathbf{x}_0,\mathbf{z}}=I_0$.

Notice that $\mathcal{L}_2$ is maximized when $q_\phi(\mathbf{z})=p(\mathbf{z})$, and thus any strict divergence D can be substituted for the KL divergence between $q_\phi(\mathbf{z})$ and $p(\mathbf{z})$, as stated in the proposition. We thus need to show that $q_\phi(\mathbf{z})=p(\mathbf{z})$ is possible. When q_ϕ is sufficiently flexible we simply have to partition the support set $\mathcal{A}$ of $p(\mathbf{z})$ into $N=\lceil e^{I_0}\rceil$ subsets $\{A_1,\cdots,A_N\}$, so that each subset satisfies $\int_{A_i}p(\mathbf{z})d\mathbf{z}=1/N$. Similarly we partition the support set $\mathcal{B}$ of $q(\mathbf{x}_0)$ into N subsets $\{B_1,\cdots,B_N\}$, so that each subset satisfies $\int_{B_i}q(\mathbf{x}_0)d\mathbf{x}_0=1/N$. Then we construct $q_\phi(\mathbf{z}\,|\,\mathbf{x}_0)$ mapping each B_i to A_i as follows

$$
q_\phi(\mathbf{z}\,|\,\mathbf{x}_0)=\begin{cases} Np(\mathbf{z}) & \mathbf{z}\in A_i\\ 0 & \text{otherwise}\end{cases}
$$

for any $\mathbf{x}_0\in B_i$. It is easy to see that this distribution is normalized because

$$
\int_{\mathbf{z}}q_\phi(\mathbf{z}\,|\,\mathbf{x}_0)d\mathbf{z}=\int_{A_i}Np(\mathbf{z})d\mathbf{z}=1
$$

Then, the equality $p(\mathbf{z})=q_\phi(\mathbf{z})$ can be established through the construction of the conditional distribution $q_\phi(\mathbf{z}\,|\,\mathbf{x}_0)$. This construction is carried out in a way that, when summed or integrated over all $\mathbf{x}_0$, gives us the unconditional distribution $q_\phi(\mathbf{z})$ that matches the target distribution $p(\mathbf{z})$.

Specifically, to obtain the unconditional distribution $q_\phi(\mathbf{z})$, we need to sum up over all $\mathbf{x}_0$, mathematically:

$$
q_\phi(\mathbf{z})=\int_{\mathbf{x}_0}q_\phi(\mathbf{z}\,|\,\mathbf{x}_0)q(\mathbf{x}_0)d\mathbf{x}_0
$$

Given the way $q_\phi(\mathbf{z}\,|\,\mathbf{x}_0)$ is defined, for a particular $\mathbf{z}\in A_i$, this would mean summing up $Np(\mathbf{z})$ exactly N times (as we have partitioned $\mathcal{B}$ into N subsets and each $\mathbf{x}_0$ in a particular B_i gives the same $Np(\mathbf{z})$). This will result in the equality $q_\phi(\mathbf{z})=p(\mathbf{z})$, hence demonstrating that such a match between $q_\phi(\mathbf{z})$ and $p(\mathbf{z})$ is indeed feasible. In addition,

$$
\mathrm{MI}_{\mathbf{x}_0,\mathbf{z}}=H_q(\mathbf{z})-H_q(\mathbf{z}\,|\,\mathbf{x}_0)
$$

$$
\begin{aligned}
&= H_q(\mathbf{z}) + \int_{\mathcal{B}} q(\mathbf{x}_0) \int_{\mathcal{A}} q_\phi(\mathbf{z}|\mathbf{x}_0) \log q_\phi(\mathbf{z}|\mathbf{x}_0) d\mathbf{z} d\mathbf{x}_0 \\
&= H_q(\mathbf{z}) + \frac{1}{N} \sum_i \int_{B_i} \int_{\mathcal{A}} q_\phi(\mathbf{z}|\mathbf{x}_0) \log q_\phi(\mathbf{z}|\mathbf{x}_0) d\mathbf{z} d\mathbf{x}_0 \\
&= H_q(\mathbf{z}) + \frac{1}{N} \sum_i \int_{A_i} N q_\phi(\mathbf{z}) \log(N q_\phi(\mathbf{z})) d\mathbf{z} \\
&= H_q(\mathbf{z}) + \sum_i \int_{A_i} q_\phi(\mathbf{z})(I_0 + \log q_\phi(\mathbf{z})) d\mathbf{z} \\
&= H_q(\mathbf{z}) + \int_{\mathcal{A}} q_\phi(\mathbf{z})(I_0 + \log q_\phi(\mathbf{z})) d\mathbf{z} \\
&= H_q(\mathbf{z}) + I_0 - H_q(\mathbf{z}) = I_0
\end{aligned}
$$

Then we reached the maximum for both objectives

$$
\begin{aligned}
\mathcal{L}_1^* &= \mathbb{E}_{q(\mathbf{x}_1)} H_q(\mathbf{x}_0|\mathbf{x}_1) - \eta I_0 \\
\mathcal{L}_2^* &= 0
\end{aligned}
$$

so their sum must also be maximized. Under this optimal solution we have that $p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t, \mathbf{z}) = q(\mathbf{x}_{t-1}|\mathbf{x}_t, \mathbf{x}_0)$ if $t \geq 2$, $q(\mathbf{x}_0|\mathbf{x}_1) = p_\theta(\mathbf{x}_0|\mathbf{x}_1, \mathbf{z})$, and $q_\phi(\mathbf{z}) = p(\mathbf{z})$. This implies $p_\theta(\mathbf{x}_0, \mathbf{x}_{1:T}, \mathbf{z}) = q_\phi(\mathbf{x}_0, \mathbf{x}_{1:T}, \mathbf{z})$, which implies $p_\theta(\mathbf{x}_0) = q(\mathbf{x}_0)$. □

C. Additional Experimental Details

In Table 7, we detail the hyperparameters used in training our InfoDiffusion and baseline models, across datasets. We also note that for all of these experiments we use the ADAM optimizer with learning rate $1\mathrm{e}^{-4}$ and train for 50 epochs. Baseline models were trained using the same optimizer, learning rate, and number of epochs. Note that in this table, there are two dimensionalities of $\mathbf{z}$ where the left one is for latent evaluation tasks and the right one is for unconditional generation.

Table 7. Hyperparameters for InfoDiffusion and baseline training. The two dimensionalities of $\mathbf{z}$ correspond to latent evaluation tasks ('Eval.') and unconditional generation ('Gen.').

	Input size	Dim. of $\mathbf{z}$		Num. channels	Num. channel mult.	Batch size	GPU
		Eval.	Gen.				
3DShapes	$3\times64\times64$	10	10	32	1, 2, 4, 8	64	TitanXP
FashionMNIST	$1\times32\times32$	32	256	32	1, 2, 4, 8	128	RTX2080Ti
CIFAR10	$3\times32\times32$	32	256	64	1, 2, 4, 8	128	TitanRTX
FFHQ	$3\times64\times64$	32	256	64	1, 2, 4, 8, 8	64	RTX4090
CelebA	$3\times64\times64$	32	256	64	1, 2, 4, 8, 8	64	TitanRTX

D. Additional Sampling Details

D.1. Sampling from Prior

To facilitate sampling from the original prior, we construct a two-phased sampling procedure for unconditional generation. For timesteps T to $T/2$, we denoise and sample using a pre-trained vanilla denoising diffusion model. In the second phase, for timesteps ranging from $T/2$ to 0, we proceed with sampling utilizing the InfoDiffusion method. We found that empirically this two-phase approach yielded superior samples compared to using InfoDiffusion prior sampling for all timesteps.

D.2. Sampling from Learned Prior

To enable sampling from the learned prior, we train a latent diffusion model, analogous to the DiffAE approach (Preechakul et al., 2022). We first train our InfoDiffusion model. We then compute the latent representation $\mathbf{z}$ for each image in a dataset using the trained $q_\phi(\mathbf{z}|\mathbf{x})$ encoder. Finally, a latent diffusion model is trained on these latent embeddings. To generate using the learned latent, the decoder is conditioned on vectors $\mathbf{z}$ sampled from the latent diffusion model.

E. Illustrations of Network Architecture

Here we provide detailed illustrations of network architectures. Figure 6 shows the UNet encoder's framework, which is used for encoding original input images into low-dimensional latent embeddings. Figure 7 shows the UNet decoder's architecture, which is used as the noise prediction network in InfoDiffusion. Figure 8 shows the details of how we implement our Auxiliary Residual Block (left) and a 1-dimensional version of UNet for latent noise prediction network (right).

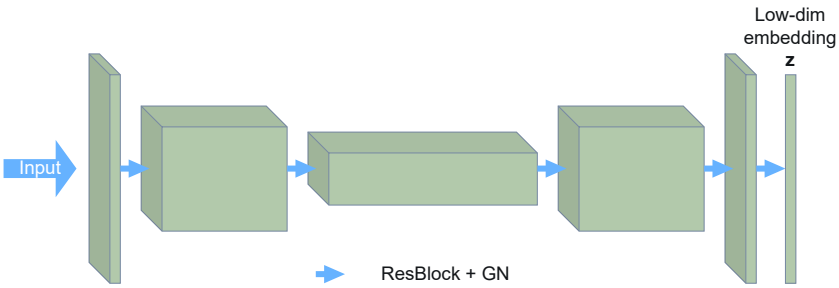


Figure 6. The UNet encoder of InfoDiffusion, for encoding input images into low-dimensional embeddings.

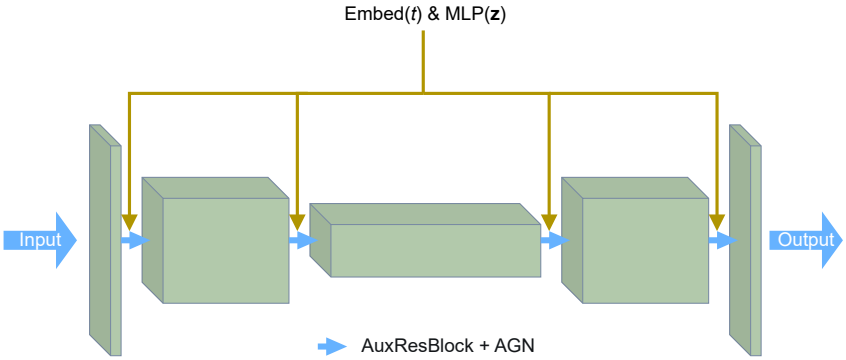


Figure 7. The UNet architecture of InfoDiffusion, conditioned on time embedding t and auxiliary variable $\mathbf{z}$.

F. Ablation: Different Approach for Conditioning on z

We perform an ablation in which we condition on $\mathbf{z}$ only in the bottleneck layer of the UNet, denoted as 'Ours w/ bott. only' in Table 8 and Table 9, as opposed to conditioning at all layers. Our findings indicate that conditioning on $\mathbf{z}$ at all layers ('Ours') offers superior outcomes in terms of FID, latent space quality (measured by the average accuracy/AUROC in predicting attributes from latents), and disentanglement metrics (including TAD and the number of attributes successfully captured).

Table 8. Comparison between two modeling choices: conditioning on $\mathbf{z}$ at just the bottleneck layer of the UNet (denoted as 'Ours w/ bott. only') versus at all layers. 'Latent quality' is measured as classification accuracy/AUROC for logistic regression classifiers trained on the auxiliary latent vector $\mathbf{z}$. We report means $\pm$ one standard deviation. Darkly shaded cells indicate best results.

	FASHIONMNIST		CIFAR10		FFHQ		CELEBA	
	LATENT QUAL. ↑	FID ↓	LATENT QUAL. ↑	FID ↓	LATENT QUAL. ↑	FID ↓	LATENT QUAL. ↑	FID ↓
OURS (WITH BOTT. ONLY)	**0.845±0.003**	29.7±0.8	0.310±0.004	40.2±1.3	0.597±0.002	41.2±1.5	0.680±0.004	26.9±1.7
OURS	0.839±0.003	**7.4±0.2**	**0.412±0.003**	**31.5±1.8**	**0.609±0.002**	**30.9±2.5**	**0.848±0.001**	**21.2±2.4**

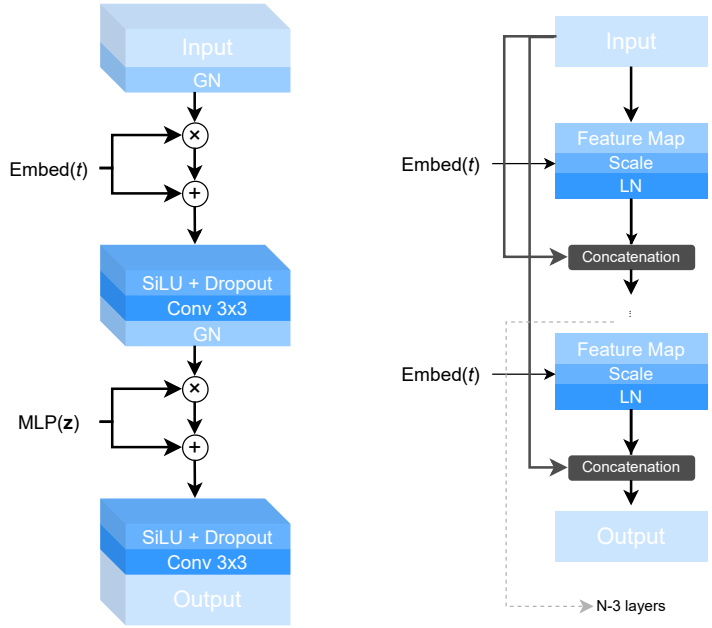


Figure 8. The implementation of Auxiliary Residual Block in InfoDiffusion with Adaptive Group Normalization (*left*); residual connections are not shown. The 1-dimensional version of UNet for latent noise prediction network (*right*).

Table 9. Comparison between two modeling choices: conditioning on **z** at just the bottleneck layer of the UNet (denoted as 'Ours w/ bott. only') versus at all layers. 'Attrs.' counts the number of "captured" attributes when calculating the TAD score. We report means ± one standard deviation. Darkly shaded cells indicate best result.

CELEBA	TAD ↑	ATTRS ↑
OURS (WITH BOTT. ONLY)	0.062±0.005	**3.0±0.0**
OURS	**0.299±0.006**	**3.0±0.0**

G. Discrete Latents

Often factors of variation can be described by categorical or binary variables. For example, the CelebA dataset contains binary annotations for each image indicating the presence or absence of certain attributes, e.g., facial hair. For this and similar datasets, it might be more appropriate to model the auxiliary latent variables as categorical, e.g., a vector of Bernoulli variables, rather than the typical continuous Gaussian distribution.

In order to perform efficient variational inference with binary variables, we use a Relaxed-Bernoulli distribution, which is derived from the Gumbel-Softmax trick (Jang et al., 2016) for categorical variables, an extension of the reparameterization trick (Kingma & Welling, 2013) to categorical distributions. This defines a "soft" or "smooth" version of the Bernoulli distribution, which enables gradient based optimization.

The Gumbel-Softmax distribution, also known as the Concrete distribution, is a way of drawing samples z from a categorical distribution with k classes, but with a differentiable function. If π_i represents the probability of class i, the Gumbel-Softmax distribution is defined as:

$$z_i = \frac{\exp((\log(\pi_i)+g_i)/\tau)}{\sum_{j=1}^{k}\exp((\log(\pi_j)+g_j)/\tau)} \tag{13}$$

where g_i are i.i.d. samples drawn from the Gumbel(0, 1) distribution and τ is the temperature parameter. The temperature parameter controls the randomness of samples. As $\tau \to 0$, samples become one-hot encoded (more deterministic), and as $\tau \to \infty$, samples approach a uniform distribution. For the Relaxed-Bernoulli, we have $k=2$, in Equation (13).

At training time, we add noise proportional to a temperature τ, which we anneal towards 0 as training progresses. For our

experiment, every 1000 steps, we reduce τ by 0.00003 from an initial value of 1 until we reach a minimum value of 0.5. A test time, $\tau = 0$ to allow for discrete sampling.

H. Regularization Coefficients

In Table 10 and Table 11, we copy results from Table 2 and Table 3, respectively, and add results for other choices of the λ and ζ regularization coefficients. We find that maximizing mutual information with $\zeta = 1$ is optimal. For the natural image datasets, $\lambda = 0.1$ yields the best results. For 3DShapes, $\lambda = 0.01$ has better performance. However, we see that our model is robust to this choice, with good latent and generated image quality for both values of λ at $\zeta = 1$.

Table 10. Latent quality, as measured by classification accuracies for logistic regression classifiers trained on the auxiliary latent vector $\mathbf{z}$, and FID. We report mean $\pm$ one standard deviation. Darkly shaded cells indicate the best while lightly shaded cells indicate the second best.

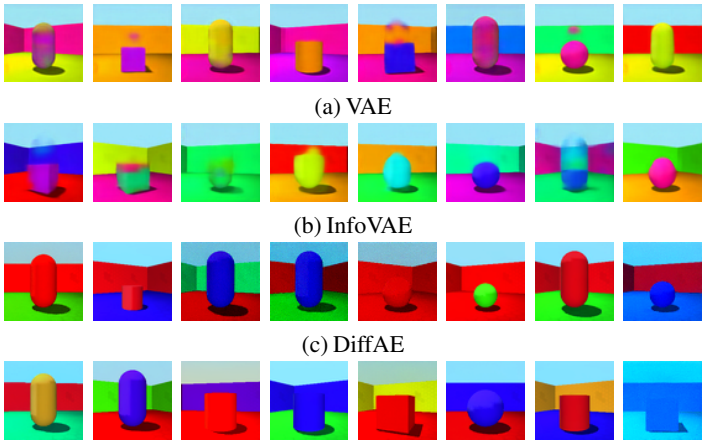
	FASHIONMNIST		CIFAR10		FFHQ	
	LATENT QUAL. ↑	FID ↓	LATENT QUAL. ↑	FID ↓	LATENT QUAL. ↑	FID ↓
AE	0.819±0.003	62.9±2.1	0.336±0.005	169.4±2.4	0.615±0.002	92.3±2.7
VAE	0.796±0.002	63.4±1.6	0.342±0.004	177.2±3.2	**0.622±0.002**	95.4±2.4
BETA-VAE	0.779±0.004	66.9±1.8	0.253±0.003	183.3±3.1	0.588±0.002	99.7±3.4
INFOVAE	0.807±0.003	55.0±1.7	0.357±0.005	160.7±2.5	0.613±0.002	86.9±2.2
DIFFAE	0.835±0.002	8.2±0.3	0.395±0.006	32.1±1.1	0.608±0.001	31.6±1.2
INFODIFFUSION ($\lambda=0.1,\zeta=0.9$)	0.579±0.004	8.9±0.1	0.243±0.003	32.4±1.8	0.540±0.001	33.6±1.5
W/LEARNED LATENT		8.9±0.3		32.3±1.9		33.1±1.3
INFODIFFUSION ($\lambda=0.1,\zeta=0.95$)	0.652±0.005	9.2±0.3	0.228±0.001	32.9±1.4	0.575±0.002	32.8±1.4
W/LEARNED LATENT		8.6±0.4		32.4±1.7		32.3±1.7
INFODIFFUSION ($\lambda=0.1,\zeta=1$)	**0.839±0.003**	8.5±0.3	**0.412±0.003**	31.7±1.2	0.609±0.002	31.2±1.6
W/LEARNED LATENT		**7.4±0.2**		**31.5±1.8**		**30.9±2.5**
INFODIFFUSION ($\lambda=0.01,\zeta=1$)	0.825±0.002	9.4±0.5	0.404±0.007	31.9±1.5	0.589±0.001	32.2±1.5
W/LEARNED LATENT		8.7±0.4		31.8±1.6		31.7±1.3

Table 11. Disentanglement and latent quality metrics and FID. For 3DShapes, we check the image quality manually and label the models which generate high-quality images with check marks ('Image Qual.'). The visualization of the samples is shown in Figure 9 in the Appendix I. For CelebA, 'Attrs.' counts the number of "captured" attributes when calculating the TAD score. 'Latent Quality' is measured as AUROC scores averaged across attributes for logistic regression classifiers trained on the auxiliary latent vector $\mathbf{z}$. We report means $\pm$ one standard deviation for quantitative metrics. Darkly shaded cells indicate the best while lightly shaded cells indicate the second best.

	3DSHAPES		CELEBA			
	DCI ↑	IMAGE QUAL.	TAD↑	ATTRS↑	LATENT QUAL. ↑	FID ↓
AE	0.219±0.001	✖	0.042±0.004	1.0±0.0	0.759±0.003	90.4±1.8
VAE	0.276±0.001	✖	0.000±0.000	0.0±0.0	0.770±0.002	94.3±2.8
BETA-VAE	0.281±0.001	✖	0.088±0.051	1.6±0.8	0.699±0.001	99.8±2.4
INFOVAE	0.134±0.001	✖	0.000±0.000	0.0±0.0	0.757±0.003	77.8±1.6
DIFFAE	0.196±0.001	✔	0.155±0.010	2.0±0.0	0.799±0.002	22.7±2.1
INFODIFFUSION ($\lambda=0.1,\zeta=0.9$)	0.027±0.001	✔	0.000±0.000	0.0±0.0	0.569±0.002	25.9±2.4
W/LEARNED LATENT		✔				24.3±1.5
INFODIFFUSION ($\lambda=0.1,\zeta=0.95$)	0.015±0.001	✔	0.000±0.000	0.0±0.0	0.577±0.008	24.5±2.1
W/LEARNED LATENT		✔				23.8±1.4
INFODIFFUSION ($\lambda=0.1,\zeta=1$)	0.109±0.001	✔	0.192±0.004	2.8±0.4	**0.848±0.001**	23.8±1.6
W/LEARNED LATENT		✔				**21.2±2.4**
INFODIFFUSION ($\lambda=0.01,\zeta=1$)	**0.342±0.002**	✔	**0.299±0.006**	**3.0±0.0**	0.836±0.002	23.6±1.3
W/LEARNED LATENT		✔				22.3±1.2

I. Qualitative Figures on 3DShapes

In Figure 9, we show samples from unconditional generation of the images by different models. The images generated by diffusion models (DiffAE and InfoDiffusion) are of high quality, with clear shapes and boundaries of the objects and backgrounds. The images generated by VAE-based models suffer from distorted shapes and blended objects. The results show that images generated by diffusion models are of higher quality than those generated by the VAE-based methods. Of note, our model is able to maintain high-quality image generation while attaining the best disentanglement metrics compared to the baseline models.



(a) VAE

(b) InfoVAE

(c) DiffAE

(d) InfoDiff

Figure 9. Visualization of image samples from unconditional generation by VAE-based models (a-b) and diffusion models (c-d).

J. Assets

Below we list the libraries and datasets that we use in our experiments with their corresponding citations and licenses (in parentheses).

Libraries We use the following open-source libraries: pytorch (Paszke et al., 2019) (license: BSD), and scikit-learn (Pedregosa et al., 2011) (BSD 3-Clause).

Datasets Our experimental section uses the following datasets: FashionMNIST (Xiao et al., 2017) (MIT), CIFAR10 (Krizhevsky et al., 2009) (MIT), FFHQ (Karras et al., 2019) (Creative Commons BY-NC-SA 4.0), CelebA (Liu et al., 2015), and 3DShapes (Burgess & Kim, 2018) (Apache 2.0).