Disentanglement Learning via Topology

Nikita Balabin [*] [1] **Daria Voronkova** [*] [1] [2] **Ilya Trofimov** [1] **Evgeny Burnaev** [1] [2] **Serguei Barannikov** [1] [3]

Abstract

We propose TopDis (Topological Disentanglement), a method for learning disentangled representations via adding a multi-scale topological loss term. Disentanglement is a crucial property of data representations substantial for the explainability and robustness of deep learning models and a step towards high-level cognition. The state-of-the-art methods are based on VAE and encourage the joint distribution of latent variables to be factorized. We take a different perspective on disentanglement by analyzing topological properties of data manifolds. In particular, we optimize the topological similarity for data manifolds traversals. To the best of our knowledge, our paper is the first one to propose a differentiable topological loss for disentanglement learning. Our experiments have shown that the proposed TopDis loss improves disentanglement scores such as MIG, FactorVAE score, SAP score, and DCI disentanglement score with respect to state-of-the-art results while preserving the reconstruction quality. Our method works in an unsupervised manner, permitting us to apply it to problems without labeled factors of variation. The TopDis loss works even when factors of variation are correlated. Additionally, we show how to use the proposed topological loss to find disentangled directions in a trained GAN.

1. Introduction

Learning disentangled representations is a fundamental challenge in deep learning, as it has been widely recognized that achieving interpretable and robust representations is crucial for the success of machine learning models (Bengio et al., 2013). Disentangled representations, in which each component of the representation corresponds to one factor of variation (Desjardins et al., 2012; Bengio et al., 2013; Cohen & Welling, 2014; Kulkarni et al., 2015; Chen et al., 2016; Higgins et al., 2017; Tran et al., 2021; Feng et al., 2018; Gonzalez-Garcia et al., 2018), have been shown to be beneficial in a variety of areas within machine learning. One key benefit of disentangled representations is that they enable effective domain adaptation, which refers to the ability of a model to generalize to new domains or tasks. Studies have shown that disentangled representations can improve performance in unsupervised domain adaptation (Yang et al., 2019; Peebles et al., 2020; Zou et al., 2020). Additionally, disentangled representations have been shown to be useful for zero-shot and few-shot learning, which are techniques for training models with limited labeled data (Bengio et al., 2013). Disentangled representations have also been shown to enable controllable image editing, which is the ability to manipulate specific aspects of an image while keeping the rest of the image unchanged (Wei et al., 2021; Wang & Ponce, 2021). This type of control can be useful in a variety of applications, such as image synthesis, style transfer, and image manipulation.

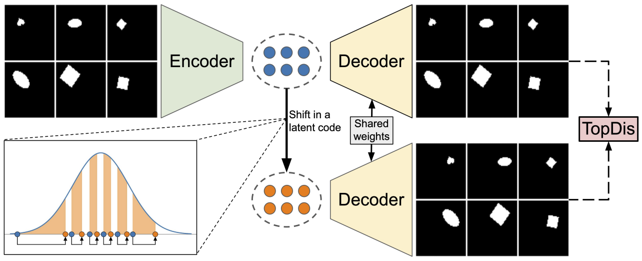


Figure 1. The TopDis pipeline process involves the following steps: encoding a batch of data samples, applying shift in a latent code, decoding both the original and the shifted latents, and finally calculating the TopDis loss between the two resulting point clouds, for details see Section 3.2.

Furthermore, disentangled representations are also believed to be a vital component for achieving high-level cognition. High-level cognition refers to the ability of a model to understand and reason about the world, and disentangled representations can play a key role in achieving this goal (Bengio, 2018).

One line of research for finding disentangled representations is to modify the Variational Autoencoder (VAE) (Kingma & Welling, 2013) using some intuition, formal-

[*] Equal contribution [1] Skolkovo Institute of Science and Technology, Moscow, Russia [2] AIRI, Moscow, Russia [3] CNRS, IMJ, Paris Cité University *Proceedings of the 41^{st} International Conference on Machine Learning*, Vienna, Austria. PMLR 235, 2024. Copyright 2024 by the author(s).

izing statistical independence of latent components (Higgins et al., 2017; Chen et al., 2018; Kim & Mnih, 2018), or the group theory based definition of disentanglement (Yang et al., 2021). Another line is to modify Generative Adversarial Networks (GANs) (Goodfellow et al., 2014; Chen et al., 2016; Lin et al., 2020; Peebles et al., 2020; Wei et al., 2021) to enforce the change in a particular component being predictable or independent in some sense from other components.

At the same time, Locatello et al. (2019) stated the impossibility of fully unsupervised learning of disentangled representation with a statistical approach. However, empirical evidence shows that disentanglement learning is possible, probably due to inductive bias either in the model or the dataset (Michlo et al., 2023; Rolinek et al., 2019). We follow Higgins et al. (2018), Section 3, where it is pointed out that one can achieve disentanglement w.r.t. the natural decomposition through active intervention, which in our case takes the form of the proposed group(oid) action shifts. Also, our work is based on exploring topological properties of a data manifold. Thus, statistical arguments of Locatello et al. (2019) do not apply in our case.

The proposed approach is grounded in the manifold hypothesis (Goodfellow et al., 2016) which posits that data points are concentrated in a vicinity of a low-dimensional manifold. For disentangled representations, it is crucial that the manifold has a specific property, namely, small topological dissimilarity between a point cloud given by a batch of data points and another point cloud obtained via the symmetry group(oid) action shift along a latent space axis. To estimate this topological dissimilarity, we utilize the tools from topological data analysis (Barannikov, 1994; Chazal & Michel, 2017). We then develop a technique for incorporating the gradient of this topological dissimilarity measure into the training of VAE-type models.

Our contributions are the following:

- We propose TopDis (Topological Disentanglement), a method for unsupervised learning of disentangled representations via adding to a VAE-type loss the topological objective;
- Our approach uses group(oid) action shifts preserving the Gaussian distribution;
- We improve the reconstruction quality by applying gradient orthogonalization;
- Experiments show that the proposed TopDis loss improves disentanglement metrics (MIG, FactorVAE score, SAP score, DCI disentanglement score) with respect to state-of-the-art results. Our method works even when factors of variation are correlated.

We release out code: https://github.com/nikitabalabin/TopDis

2. Related Work

In generative models, disentangled latent space can be obtained by designing specific architectures of neural networks (Karras et al., 2019) or optimizing additional loss functions. While the latter approach can admit supervised learning (Kulkarni et al., 2015; Kingma et al., 2014; Paige et al., 2017; Mathieu et al., 2016; Denton et al., 2017), the most challenging but practical approach is unsupervised learning of disentangled representations since the underlying factors of variation are typically unknown for real data.

The Variational Autoencoder (VAE) (Kingma & Welling, 2013), a widely used generative model, is not able to achieve disentanglement alone. To address this limitation, researchers have proposed different modifications of VAE such as β-VAE (Higgins et al., 2017), which aims to promote disentanglement by increasing the weight of Kullback–Leibler (KL) divergence between the variational posterior and the prior. To overcome the known trade-off between reconstruction quality and disentanglement (Sikka et al., 2019), some researchers have proposed to use the concept of total correlation. In β-TCVAE (Chen et al., 2018), the KL divergence between the variational posterior and the prior is decomposed into three terms: index-code mutual information, total correlation (TC), and dimension-wise KL. The authors propose to penalize the TC as the most important term for learning disentangled representations. However, estimation of the three terms of decomposition is challenging, and the authors propose a novel framework for training with the TC-decomposition using minibatches of data. The authors of FactorVAE (Kim & Mnih, 2018) proposed an additional discriminator which encourages the distribution of latent factors to be factorized and hence independent across the dimensions without significantly reducing the reconstruction loss. Recently Estermann & Wattenhofer (2023) proposed DAVA, an adversarial framework for learning disentangled representations with dynamical hyperparameters tuning. Moor et al. (2020) proposed a topological loss term for autoencoders that helps harmonise the topology of the data space with the topology of the latent space.

Locatello et al. (2019) conduct a comprehensive empirical evaluation of a large amount of existing models for learning disentangled representations, taking into account the influence of hyperparameters and initializations. They find that the FactorVAE method achieves the best quality in terms of disentanglement and stability while preserving the reconstruction quality of the generated images.

Approaches to interpretation of neural embeddings are developed in (Bertolini et al., 2023; Zhang et al., 2018; Zhou et al., 2018). In the work (Shukla et al., 2018), the authors study a geometry of deep generative models for disentangled representations. Tools of topological data analy-

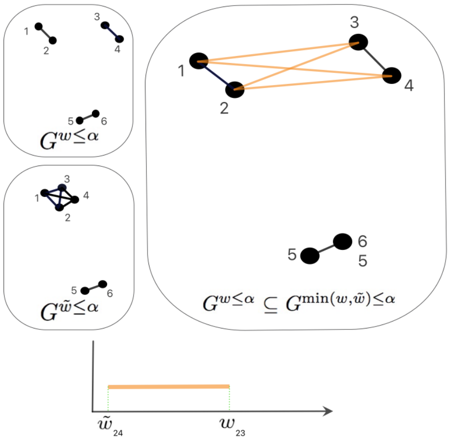

1
2
3
4
5
6
$G^{w\leq\alpha}$
$G^{\tilde{w}\leq\alpha}$
$G^{w\leq\alpha} \subseteq G^{\min(w,\tilde{w})\leq\alpha}$
$\tilde{w}_{24}$
w_{23}

Figure 2. An example of RTD calculation.

sis were previously applied to disentanglement evaluation (Barannikov et al., 2022; Zhou et al., 2021). In (Barannikov et al., 2022), topological dissimilarity in data submanifolds corresponding to slices in latent space for a simple synthetic dataset was compared.

3. Background

3.1. Variational Autoencoder

The Variational Autoencoder (VAE) (Kingma & Welling, 2013) is a generative model that encodes an object x_n into a set of parameters of the posterior distribution $q_\phi(z|x_n)$, represented by an encoder with parameters ϕ. Then it samples a latent representation from this distribution and decodes it into the distribution $p_\theta(x_n|z)$, represented by a decoder with parameters θ. The prior distribution for the latent variables is denoted as $p(z)$. In this work, we consider the factorized Gaussian prior $p(z) = N(0, I)$, and the variational posterior for an observation is also assumed to be a factorized Gaussian distribution with the mean and variance produced by the encoder. The standard VAE model is trained by minimizing the negative Evidence Lower Bound (ELBO) averaged over the empirical distribution:

$$\mathcal{L}_{VAE} = \mathcal{L}_{rec} + \mathcal{L}_{KL}, \text{ where}$$

$$\mathcal{L}_{rec} = -\frac{1}{N}\sum_{n=1}^{N} \mathbb{E}_q \left[\log p_\theta\left(x_n \mid z\right)\right],$$

$$\mathcal{L}_{KL} = -\frac{1}{N}\sum_{n=1}^{N} \mathrm{KL}\left(q_\phi\left(z \mid x_n\right) || \, p(z)\right).$$

3.2. Representation Topology Divergence

Representation Topology Divergence (RTD) (Barannikov et al., 2022) is a topological tool comparing two point clouds $X, \tilde{X}$ with one-to-one correspondence between points. RTD compares multi-scale topological features together with their localization. The distances inside clouds $X, \tilde{X}$ define two weighted graphs G^w, $G^{\tilde{w}}$ with the same vertex set X, $w_{AB} = \mathrm{dist}(A, B)$, $\tilde{w}_{AB} = \mathrm{dist}(\tilde{A}, \tilde{B})$. For a threshold α, the graphs $G^{w\leq\alpha}$, $G^{\tilde{w}\leq\alpha}$ are the α-neighborhood graphs of X and $\tilde{X}$. RTD tracks the differences in multi-scale topology between $G^{w\leq\alpha}$, $G^{\tilde{w}\leq\alpha}$ by comparing them with the graph $G^{\min(w,\tilde{w})\leq\alpha}$, which contains an edge between vertices A and B iff an edge between A and B is present in either $G^{w\leq\alpha}$ or $G^{\tilde{w}\leq\alpha}$. Increasing α from 0 to the diameter of X, the connected components in $G^{w\leq\alpha}(X)$ change from $|X|$ separate vertices to one connected component with all vertices. Let α_1 be the scale at which a pair of connected components C_1, C_2 of $G^{w\leq\alpha}$ becomes joined into one component in $G^{\min(w,\tilde{w})\leq\alpha}$. Let at some $\alpha_2 > \alpha_1$, the components C_1, C_2 become also connected in $G^{w\leq\alpha}$. *R-Cross-Barcode*$_1(X, \tilde{X})$ is the multiset of intervals like $[\alpha_1, \alpha_2]$, see Figure 2. Longer intervals indicate in general the essential topological discrepancies between X and $\tilde{X}$. By definition, RTD is the half-sum of interval lengths in R-Cross-Barcode$_1(\tilde{X}, X)$ and R-Cross-Barcode$_1(X, \tilde{X})$. By simplicial complexes based formal definition, R-Cross-Barcode is the barcode of the graph $\hat{\mathcal{G}}^{w,\tilde{w}}$ from (Barannikov et al., 2022), see Appendix L.

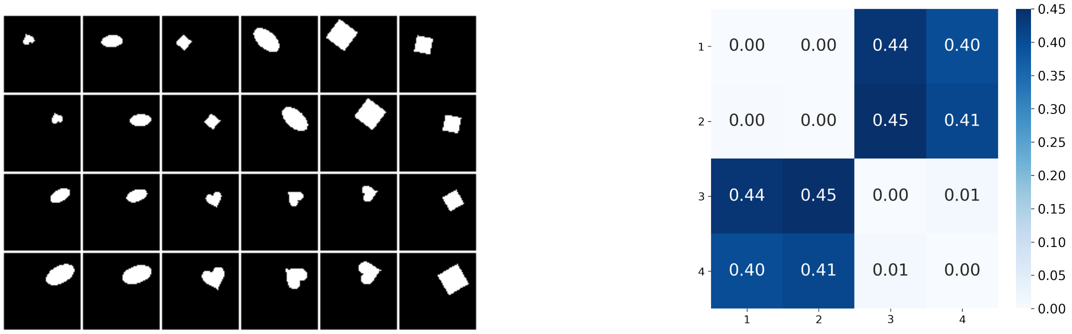
Figure 2 illustrates the calculation of RTD. The case with three clusters in X merging into two clusters in $\tilde{X}$ is shown. Edges of $G^{\tilde{w}\leq\alpha}$ not in $G^{w\leq\alpha}$, are colored in orange. In this example there are exactly four edges of different weights $(13), (14), (23), (24)$ in the point clouds X and $\tilde{X}$. The unique topological feature in *R-Cross-Barcode*$_1(X, \tilde{X})$ in this case is born at the threshold $\tilde{w}_{24}$ when the difference in the cluster structures of the two graphs arises, as the points 2 and 4 are in the same cluster at this threshold in $G^{\min(w,\tilde{w})\leq\alpha}$ and not in $G^{w\leq\alpha}$. This feature dies at the threshold $\alpha_2 = w_{23}$ since the clusters $(1, 2)$ and $(3, 4)$ are merged at this threshold in $G^{w\leq\alpha}$.

The differentiation of RTD is described in (Trofimov et al., 2023), see also Appendix Q.

4. Method

4.1. Topology-aware Loss for Group(oid) Action

First, we provide a simple example demonstrating the relevance of topology-aware loss. The disentanglement is illustrated commonly by traversals along axes in latent space. Figure 3a presents an example of various shifts in the latent space for the dSprites dataset with known factors of variations. In Figure 3b we demonstrate that transformations in disentangled directions have minimal topological dissimilarities as measured by RTD between two sets of samples. As illustrated by this example, minimization of RTD should favor the decomposition of latent space to dis-



(a) Example of traversals in dSprites dataset.

(b) RTD between point clouds represented as rows in Figure 3a.

Figure 3. Left: rows represent point clouds (mini-batches). The 1st row represents a random batch of samples; the 2nd row is obtained by equally shifting samples from the 1st row to the right; the 3rd row is placed the same as 2nd, but all objects are randomly transformed; the 4th row is a scaling of samples from 3rd row. The RTD value between the 1st and 2nd point clouds is zero, as RTD between the 3rd and 4th rows. While RTD between the 2nd and 3rd rows is large because the topological structures of these two clouds are not similar.

entangled directions.

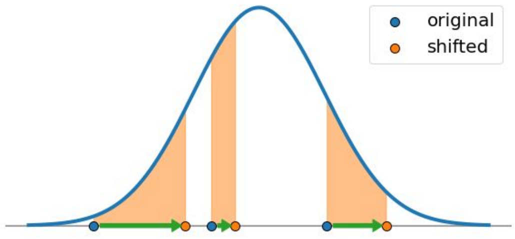


Figure 4. Shift of real line preserving $N(0, 1)$, $C = 1/8$. The three orange curvilinear rectangles have the same area: $F(z_{\text{shifted}}) - F(z) = 1/8$

As we explain below, the minimization of topology divergence is implied by the continuity of the symmetry Lie group(oid) action on data distribution.

Definition of VAE-based disentangled representation. We propose that the disentangled VAE for data distribution in a space Y consists of (cf. (Higgins et al., 2018)):

1. The encoder $h : Y \to Z$ and the decoder $f : Z \to Y$ neural networks, $Z = \mathbb{R}^n$, maximizing ELBO, with the standard $N(0, I)$ prior distribution on Z.
2. Symmetry Lie group(oid) actions on distributions in Y and Z, such that the decoder and the encoder are equivariant with respect to group(oid) action, $f(g(z)) = g(f(z)), h(g(x)) = g(h(x)), g \in G$.
3. A decomposition $G = G_1 \times \ldots \times G_n$, where $G_i \simeq \mathbb{R}$ are 1-parameter Lie subgroup(oid)s. We distinguish two cases arising in examples: a) G_i, G_j are commuting with each other: $g_i g_j = g_j g_i$ for $g_i \in G_i, g_j \in G_j$, b) $g_i g_j \approx g_j g_i$ up to higher order $\mathcal{O}(\log ||g_i|| \log ||g_j||)$.
4. The Lie group(oid) G action on the latent space decomposes and each G_i acts only on a single latent variable z_i, preserving the prior $N(0, 1)$ distribution on z_i; it follows from Proposition 4.1 that G_i acts on z_i via the shifts (1).

The concept of Lie groupoid is a formalization of continuous symmetries, when, contrary to group in (Higgins et al., 2018), symmetry action is not necessarily applicable to all points. We gather necessary definitions in Appendix M.

To relate topological features to disentangled representations, the keys are the **continuity** and the **existence of inverse** properties for the Lie action on the support of the model distribution $p_\theta(x)$ in the disentangled model. Transformations with these properties are called homeomorphisms in topology. For a map to be a homeomorphism, a fundamental requirement is that it must induce an isomorphism in homology, meaning it must preserve the topological features. By design, minimizing TopDis guarantees the maximal preservation of the topological features by the Lie action transformations. We prove a proposition which strengthens this relation further in Appendix U.

In other words, since the Lie group(oid) symmetry action by $g \in G_i$ on the support of data distribution in Y is continuous and invertible, the collection of topological features of its any subset should be preserved under g. This can be verified by RTD measuring the discrepancy in topological features at multiple scales of two data samples. Given a sample of images, TopDis loss is given by RTD between the reconstructed sample and the sample reconstructed from the group(oid) shifted latent codes, described in Section 4.2. If the two collections of topological features at multiple scales are preserved by g, then TopDis loss between the two samples is small.

4.2. Group(oid) Action Shifts Preserving the Gaussian Distribution.

Given a batch of data samples, $X = x_1, \ldots, x_N$, we sample the corresponding latent representations, $z_n \sim q_\phi(z|x_n)$, and the reconstructed samples, $\hat{x}_n \sim p_\theta(x|z_n)$. To ensure that the shifts in a latent code preserve the prior

Gaussian distribution, we propose the shifts defined by the equation:

$$z_{\text{shifted}} = F^{-1}(F(z \mid \rho, \sigma^2) + C \mid \rho, \sigma^2) \qquad (1)$$

Shifts in the latent space are performed using the *cumulative* function $F(z \mid \rho, \sigma^2)$ of the Gaussian distribution. The mean value ρ and variance σ^2 of the distribution are calculated as the empirical mean and variance of the latent code for the given sample of the data, see Algorithm 1.

Proposition 4.1. *a) For any fixed ρ, σ, the equation (1) defines a local action of the additive group $\{C \mid C \in \mathbb{R}\}$ on real line. b) This abelian group(oid) action preserves the $N(\rho, \sigma^2)$ Gaussian distribution density. c) Conversely, if a local (group(oid)) action of this abelian group preserves the $N(\rho, \sigma^2)$ distribution then the action is given by formula (1).*

See Appendix B for the proof and more details. This shifting is illustrated in Figure 4.

In the following Proposition 4.2, we bridge the gap between the proposed definition of disentanglement and the definition originating from Bengio et al. (2013). We demonstrate that the group(oid) action shift preserves the aggregate posterior latent code independence as implied by the definition from Bengio et al. (2013). This suggests that the two frameworks can be combined in practice.

Let $q(z)$ be the aggregate posterior distribution over the latent space, aggregated over the whole dataset X. And let $q(z_i)$ be the similar aggregate distribution over the latent code z_i. The formula (1) is valid and defines symmetry shifts if we replace the standard normal distribution with any distribution over the real line, we use it with the distribution $q(z_i)$ over the $i-$th latent codes.

Proposition 4.2. *a) If the distribution $q(z)$ is factorized into product $q(z) = \prod_i q(z_i)$, then the shift defined by the formula (1) acting on a single latent code z_i and leaving other latent codes fixed, preserves the latent space distribution $q(z)$. This defines the G_i groupoid action on z for any i, whose action can be then extended to points of the initial dataset X with the help of the decoder-encoder. b) Conversely, if $q(z)$ is preserved for any i by the shifts acting on z_i and defined via formula (1) from the distribution $q(z_i)$, then $q(z) = \prod_i q(z_i)$.*

The proof is given in Appendix C.

Algorithm 1 Latent traversal with a shift in the latent space.

```
Input: $z \in \mathbb{R}^{N \times d}$ – an array of latent representations from encoder. $C$ – the shift value. $F(z \mid \rho, \sigma^2)$ – the cumulative function for $\mathcal{N}(\rho, \sigma^2)$ distribution.
  $i \sim \{1, \dots, d\}$, random choice of latent code
  $s \sim \{-C, C\}$, random choice of shift direction.
  $\rho \leftarrow \text{mean}(z^{(i)})$, empirical mean value for $i$-th latent representation along batch.
  $\sigma^2 \leftarrow \text{var}(z^{(i)})$, empirical variance for the $i$-th latent representation along batch.
  $p \leftarrow F(z^{(i)} \mid \rho, \sigma^2)$, p-values of batch along $i$-th latent code, $p \in \mathbb{R}^N$
  $\mathcal{J} = \{j \mid p_j + s \in (0,1)\}$, valid set of the batch elements that can be shifted
  $z_{\text{original}} \leftarrow \{z_j \mid j \in \mathcal{J}\}$, batch of valid original latents
  $z^{(i')}_{\text{shifted}} \leftarrow z^{(i')}_{\text{original}}$, copy of latents $z^{(i')}_{\text{original}}$, $i' \neq i$
  $z^{(i)}_{\text{shifted}} \leftarrow \{F^{-1}(p_j + s \mid \rho, \sigma^2) \mid j \in \mathcal{J}\}$, apply the shift only along the $i$-th latent code.
Return: $z_{\text{original}},\ z_{\text{shifted}}$ – valid original and shifted latents. $z_{\text{original}},\ z_{\text{shifted}} \in \mathbb{R}^{|\mathcal{J}| \times d}$
```

Algorithm 2 The TopDis loss.

```
Input: $X \in \mathbb{R}^{N \times C \times H \times W}$, VAE parameters $\phi, \theta$, $p \in \{1, 2\}$ – an exponent, $C$ – the shift scale.
  $\mu_z, \sigma_z^2 \leftarrow q_\phi(z|X)$, posterior parameters from encoder given batch $X$.
  $z_{\text{original}},\ z_{\text{shifted}}$ – valid original and shifted latents, obtained by Algorithm 1
  $\hat{X}_{\text{original}} \sim p_\theta(x|z_{\text{original}})$, a reconstruction of initial batch $X$
  $\hat{X}_{\text{shifted}} \sim p_\theta(x|z_{\text{shifted}})$, a generation of modified $X$ after applying shift along some fixed latent code.
  $\mathcal{L}_{\mathcal{TD}} \leftarrow \text{RTD}^{(p)}(\hat{X}_{\text{original}}, \hat{X}_{\text{shifted}})$
Return: $\mathcal{L}_{\mathcal{TD}}$ – topological loss term.
```

4.3. The TopDis Loss

The TopDis loss is calculated using the RTD measure, which quantifies the dissimilarity between two point clouds with one-to-one correspondence. The reconstructed batch of images, $\hat{X}$, is considered as a point cloud in the $\mathbb{R}^{H \times W \times C}$ space[1]; H, W, and C are the height, width, and number of channels of the images respectively. The one-to-one correspondence between the original and shifted samples is realized naturally by the shift in the latent space. Finally, having the original and shifted point clouds:

$$\hat{X}_{\text{original}} \sim p_\theta(x|z_{\text{original}}),\ \hat{X}_{\text{shifted}} \sim p_\theta(x|z_{\text{shifted}}), \qquad (2)$$

we propose the following topological loss term (Algorithm 2):

$$\mathcal{L}_{TD} = \text{RTD}^{(p)}(\hat{X}_{\text{original}}, \hat{X}_{\text{shifted}}), \qquad (3)$$

where the superscript (p) in $\text{RTD}^{(p)}$ stands for using sum of the lengths of intervals in R-Cross-Barcode$_1$ to the $p-$th

[1] For complex images, RTD and the TopDis loss can be calculated in a representation space instead of the pixel space X.

power. The $\mathcal{L}_{TD}$ term imposes a penalty for data point clouds having different topological structures, like the 2nd and the 3rd rows in Figure 3a. Both standard values $p = 1$ and $p = 2$ perform well. In some circumstances, the $p = 2$ value is more appropriate because it penalizes significant variations in topology structures.

In this work, we propose to use the topological loss term $\mathcal{L}_{TD}$, in addition to the VAE-based loss:

$$\mathcal{L} = \mathcal{L}_{VAE-based} + \gamma \mathcal{L}_{TD}. \tag{4}$$

All variants of VAEs (classical VAE, β-VAE, FactorVAE, β-TCVAE, ControlVAE, DAVA) are modified accordingly. The computational complexity of $\mathcal{L}_{TD}$ is discussed in Appendix K. We analyze the sensitivity of the proposed approach on the value of γ in (4) in Appendix O.

4.4. Gradient Orthogonalization

As with all regularization terms, the $\mathcal{L}_{TD}$ minimization may lead to a lack of reconstruction quality. In order to achieve state-of-the-art results while minimizing the topological regularization term $\mathcal{L}_{TD}$, we apply the gradient orthogonalization between $\mathcal{L}_{TD}$ and the reconstruction loss term $\mathcal{L}_{rec}$. Specifically, if the scalar product between $\nabla_{\phi,\theta}\mathcal{L}_{rec}$ and $\nabla_{\phi,\theta}\mathcal{L}_{TD}$ is negative, then we adjust the gradients from our $\mathcal{L}_{TD}$ loss to be orthogonal to those from $\mathcal{L}_{rec}$ by applying the appropriate linear transformation:

$$\nabla^{ort}\mathcal{L}_{TD} = \nabla\mathcal{L}_{TD} - \frac{\langle \nabla\mathcal{L}_{TD}, \nabla\mathcal{L}_{rec}\rangle}{\langle \nabla\mathcal{L}_{rec}, \nabla\mathcal{L}_{rec}\rangle}\nabla\mathcal{L}_{rec}. \tag{5}$$

This technique helps to maintain a balance between the reconstruction quality and the topological regularization, thus resulting in improved overall performance. This follows essentially from the properties of the gradient of a differentiable function: making a step of length δ in a direction orthogonal to the gradient does not change the function value up to higher order $\mathcal{O}(\delta^2)$, while moving in a direction that has negative scalar product with the gradient decreases the function value. We provide an ablation study of gradient orthogonalization technique in Appendix N.

5. Experiments

5.1. Experiments on Standard Benchmarks

In the experimental section of our work, we evaluate the effectiveness of the proposed topology-based loss. Specifically, we conduct a thorough analysis of the ability of our method to learn disentangled latent spaces using various datasets and evaluation metrics. We compare the results obtained by our method with the state-of-the-art models and demonstrate the advantage of our approach in terms of disentanglement and reconstruction quality.

Datasets. We used popular benchmarks: dSprites (Matthey et al., 2017), 3D Shapes (Burgess & Kim, 2018), 3D Faces (Paysan et al., 2009), MPI 3D (Gondal et al., 2019), CelebA (Liu et al., 2015). See the description of the datasets in Appendix A. Although the datasets dSprites, 3D Shapes, 3D Faces are synthetic, the known true factors of variation allow accurate supervised evaluation of disentanglement. Hence, these datasets are commonly used in both classical and most recent works on disentanglement (Burgess et al., 2017; Kim & Mnih, 2018; Estermann & Wattenhofer, 2023; Roth et al., 2023). Finally, we examine the real-life setup with the CelebA (Liu et al., 2015) dataset.

Methods. To demonstrate that the proposed TopDis loss contributes to learning disentangled representations, we combine it with classical VAE (Kingma & Welling, 2013), β-VAE (Higgins et al., 2017), FactorVAE (Kim & Mnih, 2018), β-TCVAE (Chen et al., 2018), ControlVAE (Shao et al., 2020) and DAVA (Estermann & Wattenhofer, 2023). Following the previous work (Kim & Mnih, 2018), we used similar architectures for the encoder, decoder and discriminator (see Appendix D), the same for all models. The hyperparameters and other training details are in Appendix J. We set the latent space dimensionality to 10. Since the quality of disentanglement has high variance w.r.t. network initialization (Locatello et al., 2019), we conducted multiple runs of our experiments using different initialization seeds (see Appendix I) and averaged results.

Evaluation. Not all existing metrics were shown to be equally useful and suitable for disentanglement (Dittadi et al., 2021; Locatello et al., 2019). Due to this, hyperparameter tuning and model selection may become controversial. Moreover, in the work Zaidi et al. (2022), the authors conclude that the most appropriate metric is DCI disentanglement score (Eastwood & Williams, 2018), the conclusion which coincides with another line of research Roth et al. (2023). Based on the existing results about metrics' applicability, we restricted evaluation to measuring the following disentanglement metrics: the Mutual Information Gap (MIG) (Chen et al., 2018), the FactorVAE score (Kim & Mnih, 2018), DCI disentanglement score, and Separated Attribute Predictability (SAP) score (Kumar et al., 2018). Besides its popularity, these metrics cover all main approaches to evaluate the disentanglement of generative models (Zaidi et al., 2022): information-based (MIG), predictor-based (SAP score, DCI disentanglement score), and intervention-based (FactorVAE score).

5.1.1. Quantitative Evaluation

First of all, we study the TopDis loss as a self-sufficient disentanglement objective by adding it to VAE. Table 14 in Appendix T shows that disentanglement metrics are improved for all the datasets. Next, we add the TopDiss loss to state-of-the-art models. As demonstrated in Table 1, the

Table 1. Evaluation on the benchmark datasets. **Bold** denotes the best variant in the pair with vs. without the TopDis loss. Blue denotes the best method for a dataset/metric.

Method	**FactorVAE score**	**MIG**	**SAP**	**DCI, dis.**
dSprites				
β-VAE	0.807 ± 0.037	0.272 ± 0.101	0.065 ± 0.002	0.440 ± 0.102
β-VAE + TopDis (ours)	**0.833 ± 0.016**	**0.348 ± 0.028**	**0.066 ± 0.015**	**0.506 ± 0.050**
FactorVAE	0.819 ± 0.028	0.295 ± 0.049	0.053 ± 0.006	**0.534 ± 0.029**
FactorVAE + TopDis (ours)	**0.824 ± 0.038**	**0.356 ± 0.025**	**0.082 ± 0.001**	0.521 ± 0.044
β-TCVAE	0.810 ± 0.058	0.332 ± 0.029	0.045 ± 0.004	0.543 ± 0.049
β-TCVAE + TopDis (ours)	**0.821 ± 0.034**	**0.341 ± 0.021**	**0.051 ± 0.004**	**0.556 ± 0.042**
ControlVAE	0.806 ± 0.012	0.333 ± 0.037	0.056 ± 0.002	0.557 ± 0.009
ControlVAE + TopDis (ours)	**0.810 ± 0.012**	**0.344 ± 0.029**	**0.059 ± 0.002**	**0.578 ± 0.007**
DAVA	0.746 ± 0.099	0.253 ± 0.058	0.024 ± 0.015	0.395 ± 0.054
DAVA + TopDis (ours)	**0.807 ± 0.010**	**0.344 ± 0.010**	**0.048 ± 0.012**	**0.551 ± 0.019**
3D Shapes				
β-VAE	0.965 ± 0.060	0.740 ± 0.141	0.143 ± 0.071	0.913 ± 0.147
β-VAE + TopDis (ours)	**1.0 ± 0.0**	**0.839 ± 0.077**	**0.195 ± 0.030**	**0.998 ± 0.004**
FactorVAE	0.934 ± 0.058	0.698 ± 0.151	0.099 ± 0.064	0.848 ± 0.129
FactorVAE + TopDis (ours)	**0.975 ± 0.044**	**0.779 ± 0.036**	**0.159 ± 0.032**	**0.940 ± 0.089**
β-TCVAE	0.909 ± 0.079	0.693 ± 0.053	0.113 ± 0.070	0.877 ± 0.018
β-TCVAE + TopDis (ours)	**1.0 ± 0.0**	**0.751 ± 0.051**	**0.147 ± 0.064**	**0.901 ± 0.014**
ControlVAE	0.746 ± 0.094	0.433 ± 0.094	0.091 ± 0.068	0.633 ± 0.093
ControlVAE + TopDis (ours)	**0.806 ± 0.046**	**0.591 ± 0.055**	**0.125 ± 0.02**	**0.795 ± 0.098**
DAVA	0.800 ± 0.095	0.625 ± 0.061	0.099 ± 0.016	0.762 ± 0.088
DAVA + TopDis (ours)	**0.847 ± 0.092**	**0.679 ± 0.112**	**0.101 ± 0.043**	**0.836 ± 0.074**
3D Faces				
β-VAE	**1.0 ± 0.0**	**0.561 ± 0.017**	**0.058 ± 0.008**	**0.873 ± 0.018**
β-VAE + TopDis (ours)	**1.0 ± 0.0**	0.545 ± 0.005	0.052 ± 0.004	0.854 ± 0.013
FactorVAE	**1.0 ± 0.0**	0.593 ± 0.058	0.061 ± 0.014	0.848 ± 0.011
FactorVAE + TopDis (ours)	**1.0 ± 0.0**	**0.626 ± 0.026**	**0.062 ± 0.013**	**0.867 ± 0.037**
β-TCVAE	**1.0 ± 0.0**	0.568 ± 0.063	0.060 ± 0.017	0.822 ± 0.033
β-TCVAE + TopDis (ours)	**1.0 ± 0.0**	**0.591 ± 0.058**	**0.062 ± 0.011**	**0.859 ± 0.031**
ControlVAE	**1.0 ± 0.0**	0.447 ± 0.011	0.058 ± 0.008	0.713 ± 0.007
ControlVAE + TopDis (ours)	**1.0 ± 0.0**	**0.477 ± 0.004**	**0.074 ± 0.007**	**0.760 ± 0.014**
DAVA	**1.0 ± 0.0**	0.527 ± 0.002	0.047 ± 0.009	**0.822 ± 0.006**
DAVA + TopDis (ours)	**1.0 ± 0.0**	**0.536 ± 0.012**	**0.052 ± 0.011**	0.814 ± 0.008
MPI 3D				
β-VAE	0.428 ± 0.054	0.221 ± 0.087	0.092 ± 0.035	0.238 ± 0.049
β-VAE + TopDis (ours)	**0.479 ± 0.040**	**0.335 ± 0.056**	**0.172 ± 0.032**	**0.337 ± 0.036**
FactorVAE	0.589 ± 0.053	0.336 ± 0.056	0.179 ± 0.052	0.391 ± 0.056
FactorVAE + TopDis (ours)	**0.665 ± 0.041**	**0.377 ± 0.053**	**0.238 ± 0.040**	**0.438 ± 0.065**
β-TCVAE	0.377 ± 0.039	0.168 ± 0.021	0.084 ± 0.012	0.233 ± 0.059
β-TCVAE + TopDis (ours)	**0.501 ± 0.023**	**0.287 ± 0.011**	**0.149 ± 0.006**	**0.356 ± 0.045**
ControlVAE	0.391 ± 0.021	0.180 ± 0.048	0.107 ± 0.003	0.178 ± 0.037
ControlVAE + TopDis (ours)	**0.554 ± 0.026**	**0.232 ± 0.016**	**0.154 ± 0.003**	**0.274 ± 0.028**
DAVA	0.404 ± 0.080	0.234 ± 0.075	0.086 ± 0.043	0.268 ± 0.051
DAVA + TopDis (ours)	**0.606 ± 0.036**	**0.337 ± 0.067**	**0.181 ± 0.041**	**0.401 ± 0.049**

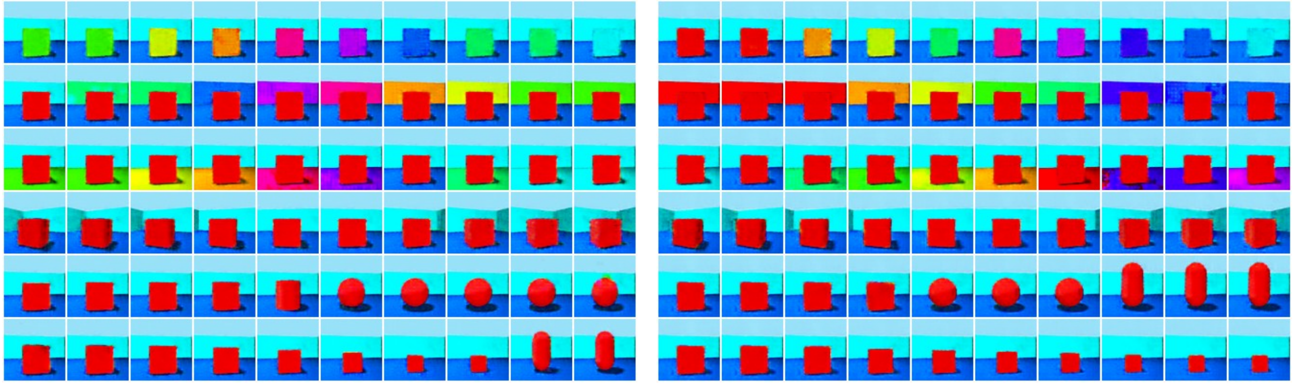


Figure 5. FactorVAE (left) and FactorVAE + TopDis (right) latent traversals on 3D Shapes.

models trained with the auxiliary TopDis loss outperform the original ones for all datasets and almost all quality measures. The addition of the TopDis loss improves the results as evaluated by FactorVAE score, MIG, SAP, DCI: on dSprites up to +8%, +35%, +100%, +39%, on 3D Shapes up to +8%, +36%, +60%, +25%, on 3D Faces up to +6%, +27%, +6% and up to +50%, +70%, +110%, +53% on MPI 3D respectively across all models. The best variant for a dataset/metrics is almost always a variant with the TopDis loss, in 94% cases. In addition, our approach preserves the reconstruction quality, see Table 4, Appendix E.

5.1.2. Qualitative Evaluation

In order to qualitatively evaluate the ability of our proposed TopDis loss to learn disentangled latent representations, we plot the traversals along a subset of latent codes that exhibit the most significant changes in an image. As a measure of disentanglement, it is desirable for each latent code to produce a single factor of variation. We compare traversals from FactorVAE and FactorVAE+TopDis decoders. The corresponding Figures 18, 19 and a detailed discussion are in Appendix X.

For the **dSprites** dataset, the simple FactorVAE model has entangled rotation and shift factors, while in FactorVAE+TopDis these factors are isolated. For the **3D Shapes** (Figure 5), FactorVAE+TopDis learns disentangled shape and scale factors, while classical FactorVAE doesn't. In **3D Faces**, FactorVAE+TopDis better disentangles azimuth, elevation, and lighting. Especially for lighting, facial attributes such as the chin, nose, and eyes are preserved for the "lightning" axis traversal. For **MPI 3D**, FactorVAE+TopDis successfully disentangles size and elevation factors. Finally, for **CelebA**, FactorVAE+TopDis disentangles skin tone and lightning, while in FactorVAE they are entangled with background and hairstyle.

5.2. Learning Disentangled Representations from Correlated Data

Existing methods for disentanglement learning make unrealistic assumptions about statistical independence of factors of variations (Träuble et al., 2021). Synthetic datasets (dSprites, 3D Shapes, 3D Faces, MPI 3D) also share this assumption. However, in the real world, causal factors are typically correlated. We carry out a series of experiments with shared confounders (one factor correlated to all others, (Roth et al., 2023)). The TopDis loss isn't based on assumptions of statistical independence. The addition of the TopDis loss gives a consistent improvement in all quality measures in this setting, see Table 13 in Appendix S.

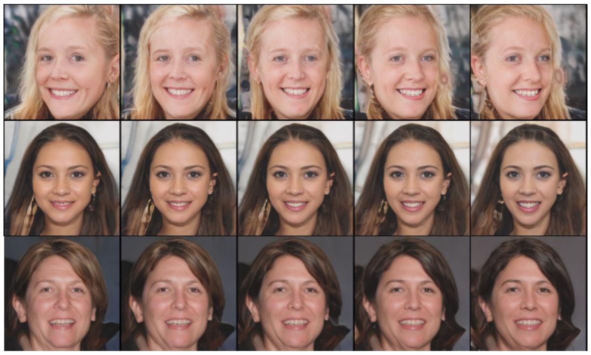


Figure 6. Three disentangled directions discovered by TopDis in StyleGAN: azimuth, smile, and hair color.

5.3. Unsupervised Discovery of Disentangled Directions in StyleGAN

We perform additional experiments to study the ability of the proposed topology-based loss to infer disentangled directions in a pretrained StyleGAN (Karras et al., 2019). We searched for disentangled directions within the space of principal components in latent space by optimizing the multi-scale topological difference after a shift along this axis $\mathrm{RTD}(\hat{X}_{\mathrm{original}}, \hat{X}_{\mathrm{shifted}})$. We were able to find three disentangled directions: azimuth, smile, hair color. See Figure 6 and Appendix H for more details. Comparison of methods dedicated to the unsupervised discovery of disentangled directions in StyleGAN is qualitative since the FFHQ dataset doesn't have labels. We do not claim that our method outperforms alternatives (Härkönen et al., 2020), as our goal is rather to demonstrate the applicability of the TopDis loss for this problem.

6. Conclusion

Our method, the Topological Disentanglement, has demonstrated its effectiveness in learning disentangled representations, in an unsupervised manner. The experiments on the dSprites, 3D Shapes, 3D Faces, and MPI 3D datasets have shown that an addition of the proposed TopDis loss improves VAE, β-VAE, FactorVAE, β-TCVAE, ControlVAE, and DAVA models in terms of disentanglement scores (MIG, FactorVAE, SAP, DCI disentanglement) while preserving the reconstruction quality. Inside our method, there is the idea of applying the topological dissimilarity to optimize disentanglement that can be added to any existing approach or used alone. We proposed to apply group(oid) action shifts preserving the Gaussian distribution in the latent space. To preserve the reconstruction quality, the gradient orthogonalization was used. Our method isn't based on the statistical independence assumption and brings improvement in quality measures even if factors of variation are correlated. In this paper, we limited ourselves to the image domain for easy visualization of disentangled directions. Extension to other domains (robotics, time series, etc.) is an interesting avenue for further research.

Acknowledgements

The work was supported by the Analytical center under the RF Government (subsidy agreement 000000D730321P5Q0002, Grant No. 70-2021-00145 02.11.2021).

Impact Statement

Disentanglement learning will lead to better generative models in various domains like image, video, text, etc. Generative models have a high potential industrial and societal impact since they may lead to creation of multimodal chat bots, AI-assisted image and video production, etc. The danger of deepfakes should be taken into account.

References

Barannikov, S. The framed Morse complex and its invariants. *Advances in Soviet Mathematics*, 21:93–116, 1994.

Barannikov, S., Trofimov, I., Balabin, N., and Burnaev, E. Representation Topology Divergence: A method for comparing neural network representations. In *ICML International Conference on Machine Learning*. PMLR, 2022.

Bengio, Y. From deep learning of disentangled representations to higher-level cognition, 2018. URL https://www.microsoft.com/en-us/research/video/from-deep-learning-of-disentangled-representations-to-higher-level-cognition/.

Bengio, Y., Courville, A., and Vincent, P. Representation learning: A review and new perspectives. *IEEE transactions on pattern analysis and machine intelligence*, 35 (8):1798–1828, 2013.

Bertolini, M., Clevert, D.-A., and Montanari, F. Explaining, evaluating and enhancing neural networks' learned representations. In *International Conference on Artificial Neural Networks*, 2023.

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

Burgess, C. P., Higgins, I., Pal, A., Matthey, L., Watters, N., Desjardins, G., and Lerchner, A. Understanding disentangling in β-vae. *Advances in neural information processing systems*, 2017.

Chazal, F. and Michel, B. An introduction to topological data analysis: fundamental and practical aspects for data scientists. *arXiv preprint arXiv:1710.04019*, 2017.

Chen, R. T., Li, X., Grosse, R. B., and Duvenaud, D. K. Isolating sources of disentanglement in variational autoencoders. *Advances in neural information processing systems*, 31, 2018.

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.

Cohen, T. and Welling, M. Learning the irreducible representations of commutative lie groups. In *International Conference on Machine Learning*, pp. 1755–1763. PMLR, 2014.

Denton, E. L. et al. Unsupervised learning of disentangled representations from video. *Advances in neural information processing systems*, 30, 2017.

Desjardins, G., Courville, A., and Bengio, Y. Disentangling factors of variation via generative entangling. *arXiv preprint arXiv:1210.5474*, 2012.

Dittadi, A., Trauble, F., Locatello, F., Wuthrich, M., Agrawal, V., Winther, O., Bauer, S., and Scholkopf, B. On the transfer of disentangled representations in realistic settings. In *ICLR International Conference on Learning Representations*, 2021.

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

Estermann, B. and Wattenhofer, R. Dava: Disentangling adversarial variational autoencoder. In *ICLR International Conference on Learning Representations*, 2023.

Farajtabar, M., Azizan, N., Mott, A., and Li, A. Orthogonal gradient descent for continual learning. In *International Conference on Artificial Intelligence and Statistics*, pp. 3762–3773. PMLR, 2020.

Feng, Z., Wang, X., Ke, C., Zeng, A., Tao, D., and Song, M. Dual swap disentangling. *Advances in neural information processing systems*, 32, 2018.

Gaujac, B., Feige, I., and Barber, D. Learning disentangled representations with the wasserstein autoencoder. In *Machine Learning and Knowledge Discovery in Databases. Research Track: European Conference, ECML PKDD 2021, Bilbao, Spain, September 13–17, 2021, Proceedings, Part III 21*, pp. 69–84. Springer, 2021.

Gondal, M. W., Wuthrich, M., Miladinovic, D., Locatello, F., Breidt, M., Volchkov, V., Akpo, J., Bachem, O., Schölkopf, B., and Bauer, S. On the transfer of inductive bias from simulation to the real world: a new disentanglement dataset. *Advances in Neural Information Processing Systems*, 32, 2019.

Gonzalez-Garcia, A., van de Weijer, J., and Bengio, Y. Image-to-image translation for cross-domain disentanglement. *Advances in neural information processing systems*, 2018.

Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., and Bengio, Y. Generative adversarial nets. *Advances in neural information processing systems*, 27:2672–2680, 2014.

Goodfellow, I., Bengio, Y., Courville, A., and Bengio, Y. *Deep learning*, volume 1. MIT press Cambridge, 2016.

Härkönen, E., Hertzmann, A., Lehtinen, J., and Paris, S. Ganspace: Discovering interpretable gan controls. *Advances in Neural Information Processing Systems*, 33: 9841–9850, 2020.

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 *ICLR International conference on learning representations*, 2017.

Higgins, I., Amos, D., Pfau, D., Racaniere, S., Matthey, L., Rezende, D., and Lerchner, A. Towards a definition of disentangled representations. *arXiv preprint arXiv:1812.02230*, 2018.

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

Kim, H. and Mnih, A. Disentangling by factorising. In *International Conference on Machine Learning*, pp. 2649–2658. PMLR, 2018.

Kingma, D. P. and Ba, J. Adam: A method for stochastic optimization. In *ICLR International Conference on Learning Representations*, 2015.

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

Kingma, D. P., Mohamed, S., Jimenez Rezende, D., and Welling, M. Semi-supervised learning with deep generative models. *Advances in neural information processing systems*, 27, 2014.

Kulkarni, T. D., Whitney, W. F., Kohli, P., and Tenenbaum, J. Deep convolutional inverse graphics network. *Advances in neural information processing systems*, 28, 2015.

Kumar, A., Sattigeri, P., and Balakrishnan, A. Variational inference of disentangled latent concepts from unlabeled observations. In *ICLR International Conference on Learning Representations*, 2018.

Leygonie, J., Oudot, S., and Tillmann, U. A framework for differential calculus on persistence barcodes. *Foundations of Computational Mathematics*, pp. 1–63, 2021.

Lin, Z., Thekumparampil, K., Fanti, G., and Oh, S. Infogan-cr and model centrality: Self-supervised model training and selection for disentangling gans. In *international conference on machine learning*, pp. 6127–6139. PMLR, 2020.

Liu, Z., Luo, P., Wang, X., and Tang, X. Deep learning face attributes in the wild. In *Proceedings of the IEEE international conference on computer vision*, pp. 3730–3738, 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.

Locatello, F., Bauer, S., Lucic, M., Rätsch, G., Gelly, S., Schölkopf, B., and Bachem, O. A sober look at the unsupervised learning of disentangled representations and their evaluation. *Journal of machine learning research*, 21, 2020.

Mathieu, M. F., Zhao, J. J., Zhao, J., Ramesh, A., Sprechmann, P., and LeCun, Y. Disentangling factors of variation in deep representation using adversarial training.

Advances in neural information processing systems, 29, 2016.

Matthey, L., Higgins, I., Hassabis, D., and Lerchner, A. dsprites: Disentanglement testing sprites dataset. https://github.com/deepmind/dsprites-dataset/, 2017.

Michlo, N., Klein, R., and James, S. Overlooked implications of the reconstruction loss for vae disentanglement. *Proceedings of the Thirty-Second International Joint Conference on Artificial Intelligence*, 2023.

Moor, M., Horn, M., Rieck, B., and Borgwardt, K. Topological autoencoders. In *International conference on machine learning*, pp. 7045–7054. PMLR, 2020.

Paige, B., van de Meent, J.-W., Desmaison, A., Goodman, N., Kohli, P., Wood, F., Torr, P., et al. Learning disentangled representations with semi-supervised deep generative models. *Advances in neural information processing systems*, 30, 2017.

Paysan, P., Knothe, R., Amberg, B., Romdhani, S., and Vetter, T. A 3d face model for pose and illumination invariant face recognition. In *2009 sixth IEEE international conference on advanced video and signal based surveillance*, pp. 296–301. Ieee, 2009.

Peebles, W., Peebles, J., Zhu, J.-Y., Efros, A., and Torralba, A. The hessian penalty: A weak prior for unsupervised disentanglement. In *European Conference on Computer Vision*, pp. 581–597. Springer, 2020.

Ridgeway, K. and Mozer, M. C. Learning deep disentangled embeddings with the f-statistic loss. *Advances in neural information processing systems*, 31, 2018.

Rolinek, M., Zietlow, D., and Martius, G. Variational autoencoders pursue pca directions (by accident). In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 12406–12415, 2019.

Roth, K., Ibrahim, M., Akata, Z., Vincent, P., and Bouchacourt, D. Disentanglement of correlated factors via hausdorff factorized support. In *ICLR International Conference on Learning Representations*, 2023.

Shao, H., Yao, S., Sun, D., Zhang, A., Liu, S., Liu, D., Wang, J., and Abdelzaher, T. Controlvae: Controllable variational autoencoder. In *International Conference on Machine Learning*, pp. 8655–8664. PMLR, 2020.

Shukla, A., Uppal, S., Bhagat, S., Anand, S., and Turaga, P. Geometry of deep generative models for disentangled representations. In *Proceedings of the 11th Indian Conference on Computer Vision, Graphics and Image Processing*, pp. 1–8, 2018.

Sikka, H., Zhong, W., Yin, J., and Pehlevant, C. A closer look at disentangling in β-vae. In *2019 53rd Asilomar Conference on Signals, Systems, and Computers*, pp. 888–895. IEEE, 2019.

Suteu, M. and Guo, Y. Regularizing deep multi-task networks using orthogonal gradients. *arXiv preprint arXiv:1912.06844*, 2019.

Tran, L., Khasahmadi, A. H., Sanghi, A., and Asgari, S. Group-disentangled representation learning with weakly-supervised regularization. *arXiv preprint arXiv:2110.12185v1*, 2021.

Träuble, F., Creager, E., Kilbertus, N., Locatello, F., Dittadi, A., Goyal, A., Schölkopf, B., and Bauer, S. On disentangled representations learned from correlated data. pp. 10401–10412. PMLR, 2021.

Trofimov, I., Cherniavskii, D., Tulchinskii, E., Balabin, N., Burnaev, E., and Barannikov, S. Learning topology-preserving data representations. In *ICLR International Conference on Learning Representations*, 2023.

Wang, B. and Ponce, C. R. The geometry of deep generative image models and its applications. In *ICLR International Conference on Learning Representations*, 2021.

Wei, Y., Shi, Y., Liu, X., Ji, Z., Gao, Y., Wu, Z., and Zuo, W. Orthogonal jacobian regularization for unsupervised disentanglement in image generation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 6721–6730, 2021.

Weinstein, A. Groupoids: unifying internal and external symmetry. *Notices of the AMS*, 43(7):744–752, 1996.

Yang, J., Dvornek, N. C., Zhang, F., Chapiro, J., Lin, M., and Duncan, J. S. Unsupervised domain adaptation via disentangled representations: Application to cross-modality liver segmentation. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pp. 255–263. Springer, 2019.

Yang, T., Ren, X., Wang, Y., Zeng, W., and Zheng, N. Towards building a group-based unsupervised representation disentanglement framework. In *ICLR International Conference on Learning Representations*, 2021.

Yu, T., Kumar, S., Gupta, A., Levine, S., Hausman, K., and Finn, C. Gradient surgery for multi-task learning. *Advances in Neural Information Processing Systems*, 33: 5824–5836, 2020.

Zaidi, J., Boilard, J., Gagnon, G., and Carbonneau, M.-A. Measuring disentanglement: A review of metrics. In *IEEE Transactions on Neural Networks and Learning Systems*, pp. 1–15, 2022.

Zhang, Q., Wu, Y. N., and Zhu, S.-C. Interpretable convolutional neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 8827–8836, 2018.

Zhou, B., Bau, D., Oliva, A., and Torralba, A. Interpreting deep visual representations via network dissection. *IEEE transactions on pattern analysis and machine intelligence*, 41(9):2131–2145, 2018.

Zhou, S., Zelikman, E., Lu, F., Ng, A. Y., Carlsson, G. E., and Ermon, S. Evaluating the disentanglement of deep generative models through manifold topology. In *ICLR International Conference on Learning Representations*, 2021.

Zou, Y., Yang, X., Yu, Z., Kumar, B., and Kautz, J. Joint disentangling and adaptation for cross-domain person reidentification. In *European Conference on Computer Vision*, pp. 87–104. Springer, 2020.

A. Datasets

This section provides a brief overview of the benchmark datasets along with sample images.

dSprites contains 2D shapes generated procedurally from five independent latent factors: shape, scale, rotation, x-coordinate, and y-coordinate of a sprite. See Figure 7 for sample images.

3D Shapes consists of 3D scenes with six generative factors: floor hue, wall hue, orientation, shape, scale, and shape color. See Figure 8 for sample images.

3D Faces The 3D Faces dataset consists of 3D rendered faces with four generative factors: face id, azimuth, elevation, lighting. See Figure 9 for sample images.

MPI 3D contains images of physical 3D objects with seven generative factors: color, shape, size, camera height, background color, horizontal and vertical axes. We used the MPI3D-Complex version which provides samples of complex real-world shapes from the robotic platform. See Figure 11 for sample images.

CelebA provides images of aligned faces of celebrities. This dataset doesn't have any ground truth generative factors because of its real-world nature. Figure 9 demonstrates the sample images.

FFHQ contains high-quality images of human faces at resolution. Similarly to CelebA, due to its real-world origin, this dataset doesn't have any ground truth factors of variation. Figure 12 provides the sample images.

In our work, we used both synthetic and real-world datasets as in recent state-of-the-art research in disentanglement, see Roth et al. (2023); Estermann & Wattenhofer (2023); Shao et al. (2020); Chen et al. (2018); Kim & Mnih (2018). Estermann & Wattenhofer (2023) Shao et al. (2020) Chen et al. (2018) Kim & Mnih (2018) We highlight that we utilize the MPI3D-Real-Complex version of the MPI 3D dataset, which was developed based on robotic platform in real-world setting and contains complex real-world shapes.

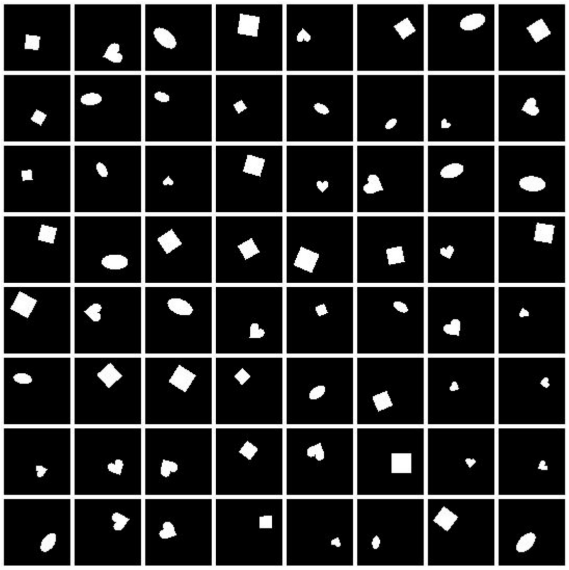


Figure 7. Samples from dSprites.

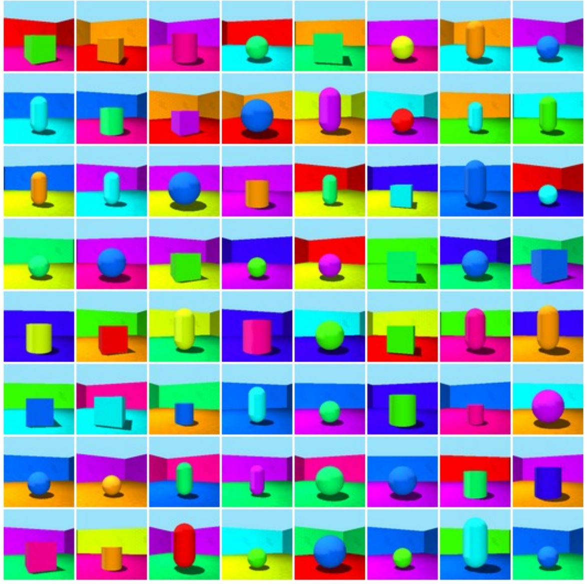


Figure 8. Samples from 3D Shapes.

B. Proof of Proposition 4.1

Proof. a) Two consecutive shifts defined in equation (1) give

$$F^{-1}(F(F^{-1}(F(z \mid \rho,\sigma^2) + C_1 \mid \rho,\sigma^2) + C_2 \mid \rho,\sigma^2) = F^{-1}(F(z \mid \rho,\sigma^2) + C_1 + C_2 \mid \rho,\sigma^2)$$

So the two consecutive shifts with C_1, C_2 is the same as the single shift with $C_1 + C_2$.

Figure 9. Samples from 3D Faces.

Figure 10. Samples from CelebA.

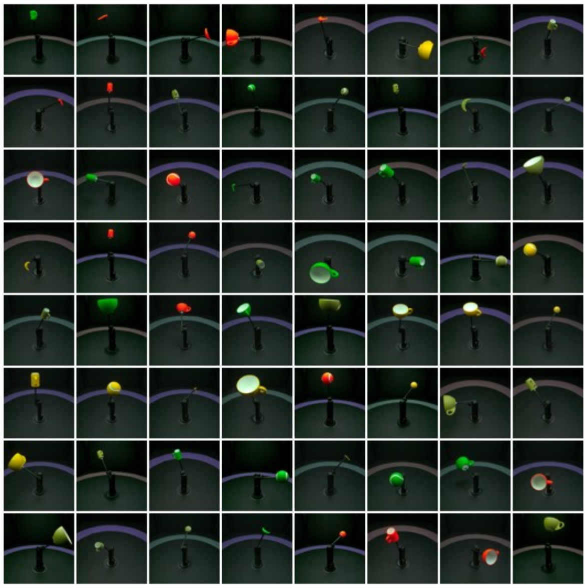


Figure 11. Samples from MPI 3D.

Figure 12. Samples from FFHQ.

b) We have for a given shift with parameter C and any pair of shifted points $z_{\text{shifted}}, \tilde{z}_{\text{shifted}} \in \mathbb{R}$:

$$F(\tilde{z}_{\text{shifted}}) - F(z_{\text{shifted}}) = (F(\tilde{z}) + C) - (F(z) + C) = F(\tilde{z}) - F(z) \tag{6}$$

i.e. if the shift of points $z, \tilde{z} \in \mathbb{R}$ is defined, then the $N(\rho, \sigma^2)$ measure of the line segment $[z, \tilde{z}]$ is preserved under the shift.

c) Conversely, if for $z, \tilde{z} \in \mathbb{R}$ the $N(\rho, \sigma^2)$ measure of the line segment $[z, \tilde{z}]$ is preserved under the shift, i.e. $F(\tilde{z}_{\text{shifted}}) - F(z_{\text{shifted}}) = F(\tilde{z}) - F(z)$, then setting z to any fixed value, for example $z = -\infty$, we get $F(\tilde{z}_{\text{shifted}}) = F(\tilde{z}) + C$. □

Notice also that $F(z_{\text{shifted}}) - F(z) = C$, so the three orange curvilinear rectangles on Figure 4 have the same area $C = 1/8$.

Recall that $F(z \mid \rho, \sigma^2) = \frac{1}{\sigma\sqrt{2\pi}} \int_{-\infty}^{z} \exp\left(-\frac{(t-\rho)^2}{2\sigma^2}\right) dt$ denotes here the cumulative function of the Gaussian distribution $N(\rho, \sigma^2)$.

Remark B.1. *Notice that, during the calculation of the topological term, we do not consider the data points with $F(z) + C > 1$ ($F(z) + C < 0$), i.e. whose latent codes are already at the very right (left) tail of the distribution and which thus cannot be shifted to the right (respectfully, left).*

C. Proof of Proposition 4.2

Proof. a) The shift defined by (1) for the distribution $q(z_i)$ acting on the latent space, preserves also any $q(z_j)$ for $j \neq i$.
b) The result follows from the case of an arbitrary distribution over a pair of random variables z_1, z_2. For two variables, it follows from the Bayes formula that the shifts of z_1 preserve the conditional $q(z_2|z_1)$. Since the group(oid) action is transitive it follows that the conditional does not depend on z_1, and hence $q(z_1, z_2) = q(z_1)q(z_2)$.

D. Architecture Details

Table 2 demonstrates the architecture of VAE model while the discriminator's architecture is presented in Table 3. In the experiments with VAE(+TopDis), β-VAE(+TopDis), FactorVAE(+TopDis), β-TCVAE(+TopDis), ControlVAE(+TopDis), DAVA(+TopDis) (Tables 14, 1, 4), we used the following architecture configurations:

- dSprites: num channels $= 1, m_1 = 2, m_2 = 2, m_3 = 4, m_4 = 4, n = 5$;
- 3D Shapes: num channels $= 3, m_1 = 1, m_2 = 1, m_3 = 1, m_4 = 2, n = 5$;
- 3D Faces: num channels $= 1, m_1 = 1, m_2 = 1, m_3 = 1, m_4 = 2, n = 5$;
- MPI 3D: num channels $= 3, m_1 = 1, m_2 = 1, m_3 = 1, m_4 = 2, n = 6$;
- CelebA: num channels $= 3, m_1 = 1, m_2 = 1, m_3 = 1, m_4 = 2, n = 5$.

Table 2. Encoder and Decoder architecture for the dSprites experiments.

Encoder	Decoder
Input: 64 × 64 × num channels	Input: $\mathbb{R}^{10}$
4 × 4 conv, 32 ReLU, stride 2	1 × 1 conv, 128 × m_4 ReLU, stride 1
4 × 4 conv, 32 · m_1 ReLU, stride 2	4 × 4 upconv, 64 · m_3 ReLU, stride 1
4 × 4 conv, 64 · m_2 ReLU, stride 2	4 × 4 upconv, 64 · m_2 ReLU, stride 2
4 × 4 conv, 64 · m_3 ReLU, stride 2	4 × 4 upconv, 32 · m_1 ReLU, stride 2
4 × 4 conv, 128 · m_4 ReLU, stride 1	4 × 4 upconv, 32 ReLU, stride 2
1 × 1 conv, 2 × 10, stride 1	4 × 4 upconv, 1, stride 2

Table 3. FactorVAE Discriminator architecture.

Discriminator
[FC, 1000 leaky ReLU] × n
FC, 2

E. Reconstruction Error

We provide the reconstruction error for all the evaluated models in Table 4.

Table 4. Reconstruction error.

Method	dSprites	3D Shapes	3D Faces	MPI 3D
VAE	8.67 ± 0.29	3494.10 ± 3.27	1374.42 ± 3.38	3879.75 ± 0.49
VAE + TopDis (ours)	9.54 ± 0.19	3489.53 ± 1.50	1376.22 ± 0.32	3879.89 ± 0.51
β-VAE	12.97 ± 0.50	3500.60 ± 13.59	1379.64 ± 0.19	3888.84 ± 2.45
β-VAE + TopDis (ours)	13.75 ± 0.63	3495.76 ± 6.54	1380.10 ± 0.19	3886.57 ± 0.81
FactorVAE	14.65 ± 0.41	3501.53 ± 13.43	1488.26 ± 4.47	3884.31 ± 0.59
FactorVAE + TopDis (ours)	14.72 ± 0.49	3504.42 ± 9.98	1377.93 ± 3.47	3885.74 ± 0.82
β-TCVAE	17.87 ± 0.56	3492.25 ± 5.79	1375.03 ± 3.41	3891.03 ± 1.41
β-TCVAE + TopDis (ours)	17.32 ± 0.31	3495.13 ± 2.49	1376.21 ± 3.09	3889.34 ± 1.97
ControlVAE	15.32 ± 0.47	3499.61 ± 12.13	1404.42 ± 5.01	3889.81 ± 0.43
ControlVAE + TopDis (ours)	14.91 ± 0.39	3500.28 ± 10.73	1389.42 ± 4.47	3889.24 ± 0.50
DAVA	36.41 ± 2.03	3532.56 ± 14.14	1403.77 ± 0.99	3890.42 ± 2.15
DAVA + TopDis (ours)	26.03 ± 2.51	3537.39 ± 40.52	1403.20 ± 0.49	3893.41 ± 3.48

F. Training Statistics

Figure 13 demonstrates that TopDis loss decreases during training and has good negative correlation with MIG score, as expected. TopDis score was averaged with a sliding window of size 500, MIG was calculated every 50000 iterations.

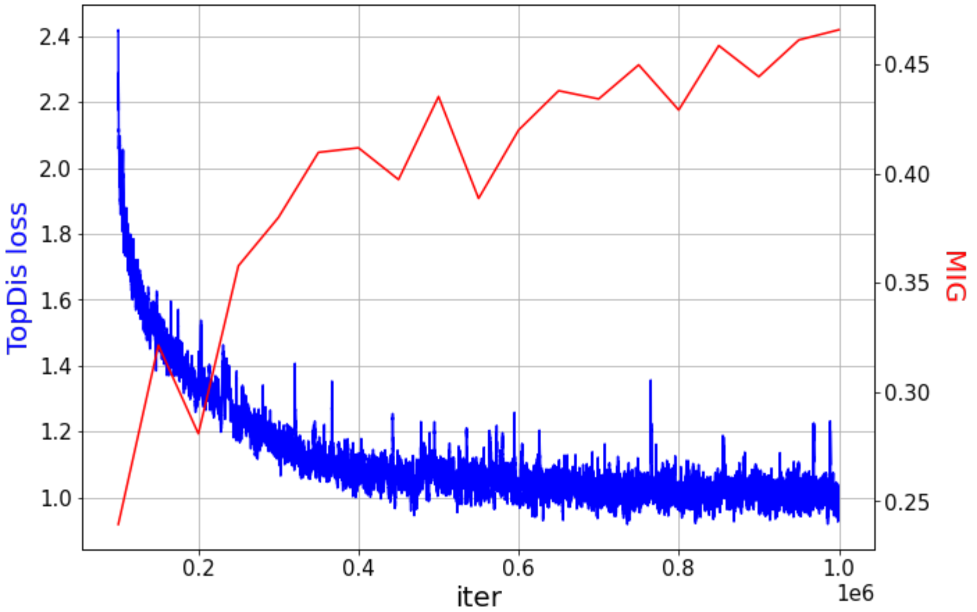


Figure 13. Training curves of TopDis loss and MIG for FactorVAE + TopDis on MPI 3D dataset.

Also, we provide the TopDis scores for different axes in the latent space for the case of BetaVAE + TopDis and 3D Shapes dataset in Table 5. We randomly sample a batch with a fixed factor of variation. Following the definition of TopDis, we shift this batch with our shift in the probability space and compute RTD between the original and shifted batches. We average the result over different batch samples. As it can be seen, the values are in the same range for all factors, and none of the factors incurs much higher value of TopDis than others.

Table 5. TopDis scores for different axes in the latent space for BetaVAE + TopDis on 3D Shapes.

Factor	**Floor hue**	**Wall hue**	**Object hue**	**Object size**	**Object shape**	**Orientation**
TopDis	1.335 ± 0.134	0.926 ± 0.086	1.207 ± 0.144	1.115 ± 0.138	0.678 ± 0.029	0.763 ± 0.067

Table 6. Comparison with the TCWAE method

Method	**FactorVAE score**	**MIG**	**SAP**	**DCI, dis.**
		dSprites		
TCWAE	0.76 ± 0.03	0.32 ± 0.04	0.072 ± 0.004	-
FactorVAE + TopDis (ours)	$\mathbf{0.82 \pm 0.04}$	$\mathbf{0.36 \pm 0.03}$	$\mathbf{0.082 \pm 0.001}$	$\mathbf{0.52 \pm 0.04}$

G. More on Related Work

In Table 6, we compare our results with another recent state-of-the-art method, TCWAE (Gaujac et al., 2021). Since there is no code available to replicate their results, we present the values from the original papers. The architecture and training setup were essentially identical to what is described in this paper.

In Table 7, we provide a comparison with HFS (Roth et al., 2023). As our TopDis, HFS loss is used in addition to standard disentanglement methods. Below we show the DCI disentanglement score for a subset of methods common in our paper and (Roth et al., 2023), numbers are taken from (Roth et al., 2023). TopDis always has a higher score.

H. Unsupervised Discovery of Disentangled Directions in StyleGAN

We perform additional experiments to study the ability of the proposed topology-based loss to infer disentangled directions in a pretrained GAN. In experiments, we used StyleGAN (Karras et al., 2019)[2]. The unsupervised directions were explored in the style space $\mathcal{Z}$. To filter out non-informative directions we followed the approach from Härkönen et al. (2020) and selected top 32 directions by doing PCA for the large batch of data in the style space. Then, we selected the new basis $n_i,\ i = 1, \dots, 32$ in this subspace, starting from a random initialization. Directions n_i were selected sequentially by minimization of RTD along shifts in $\mathcal{Z}$ space:

$$\mathrm{RTD}(Gen_k(Z), Gen_k(Z + cn_i)),$$

where $Gen_k(\cdot)$ is the $k-$layer of the StyleGAN generator (we used $k = 3$). After each iteration the Gram–Schmidt orthonormalization process for n_i was performed. We were able to discover at least 3 disentangled directions: azimuth (Fig. 14), smile (Fig. 15), hair color (Fig. 16).

[2] we used a PyTorch reimplementation from: `https://github.com/rosinality/style-based-gan-pytorch`.

Table 7. Comparison with the HFS loss by a DCI disentanglement score.

Method	**dSprites**	**Shapes3D**	**MPI 3D**
β-VAE + HFS	0.506	0.912	0.328
β-VAE + TopDis	**0.556**	**0.998**	**0.337**
β-TCVAE + HFS	0.499	0.857	0.328
β-TCVAE + TopDis	**0.531**	**0.901**	**0.356**

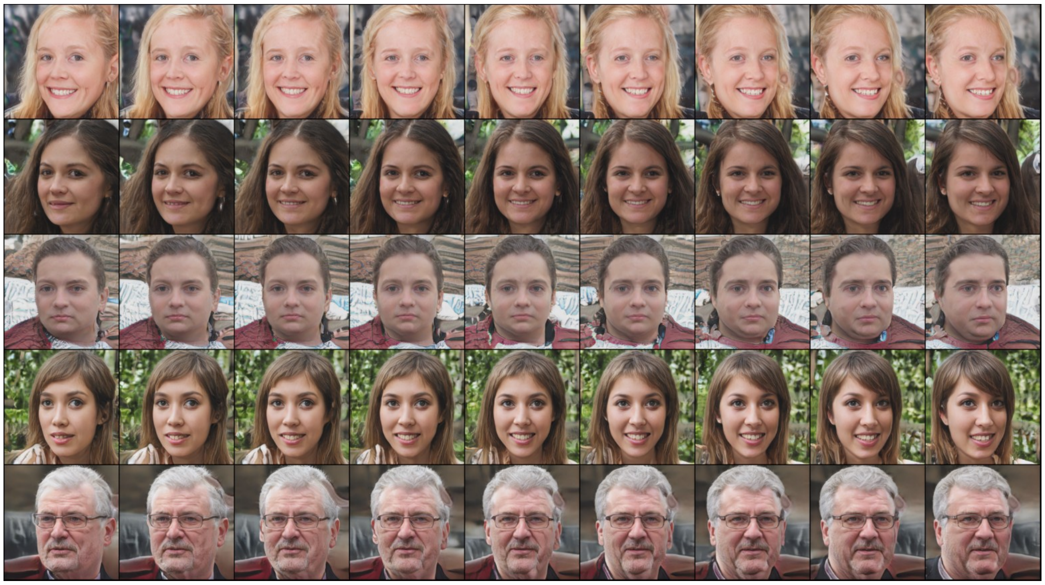


Figure 14. StyleGAN, change of an azimuth.

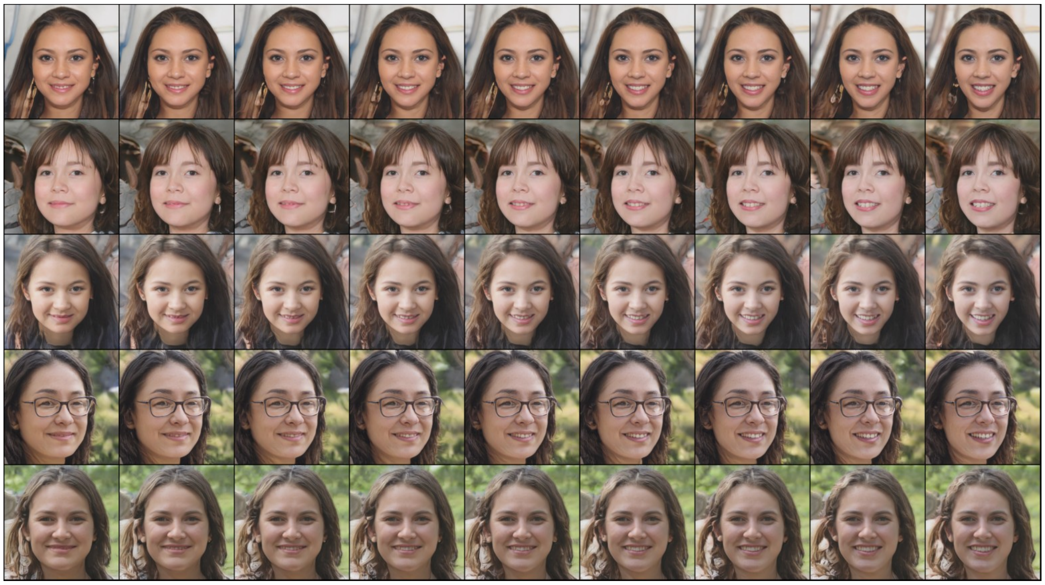


Figure 15. StyleGAN, change of a smile.

I. Extra Details on the Significance of TopDis Effect

In order to accurately assess the impact of the TopDis term, we employed a consistent set of random initializations. This approach was adopted to eliminate potential confounding factors that may arise from disparate initial conditions. This allowed us to attribute any observed improvements in disentanglement quality specifically to the inclusion of the TopDis term in our model. In Table 8 we demonstrate the consistent improvement across multiple runs.

J. Training Details

Following the previous work Kim & Mnih (2018), we used similar architectures for the encoder, decoder and discriminator, the same for all models. We set the latent space dimensionality to 10. We normalized the data to $[0, 1]$ interval and trained 1M iterations with batch size of 64 and Adam (Kingma & Ba, 2015) optimizer. The learning rate for VAE updates was 10^{-4}

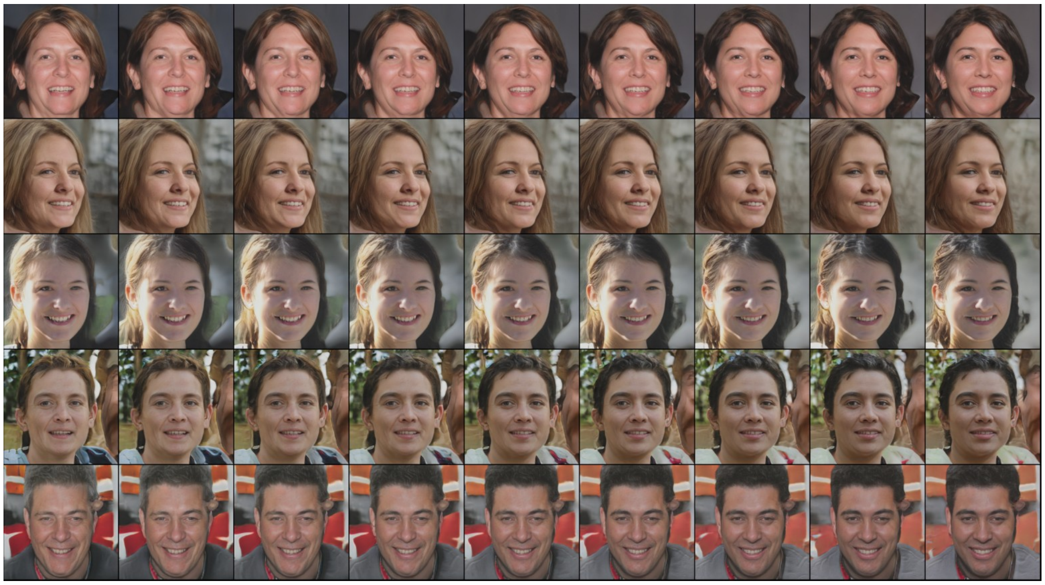


Figure 16. StyleGAN, change of a hair color.

for dSprites and MPI 3D datasets, 10^{-3} for 3D Shapes dataset, and 2×10^{-4} for 3D faces and CelebA datasets, $\beta_1 = 0.9$, $\beta_2 = 0.999$, while the learning rate for discriminator updates was 10^{-4} for dSprites, 3D Faces, MPI 3D and CelebA datasets, 10^{-3} for 3D Shapes dataset, $\beta_1 = 0.5$, $\beta_2 = 0.9$ for discriminator updates. In order to speed up convergence, we first trained the model without TopDis loss for a certain number of iterations and then continued training with TopDis loss. We also fine-tuned the hyperparameter γ over set commonly used in the literature (Kim & Mnih, 2018; Locatello et al., 2019; Ridgeway & Mozer, 2018) to achieve the best performance on the baseline models.

The best performance found hyperparameters are the following:

- dSprites. β-VAE: $\beta = 2$, β-VAE + TopDis: $\beta = 2, \gamma = 4$, FactorVAE: $\gamma = 20$, FactorVAE + TopDis: $\gamma_1 = 5, \gamma_2 = 5$, β-TCVAE: $\beta = 6$, β-TCVAE+ TopDis: $\beta = 6, \gamma = 5$, DAVA + TopDis: $\gamma = 5$;

- 3D Shapes. β-VAE: $\beta = 2$, β-VAE + TopDis: $\beta = 2, \gamma = 1$, FactorVAE: $\gamma = 30$, FactorVAE + TopDis: $\gamma_1 = 5, \gamma_2 = 5$, β-TCVAE: $\beta = 4$, β-TCVAE + TopDis: $\beta = 4, \gamma = 5$, DAVA + TopDis: $\gamma = 3$;

- 3D Faces. β-VAE: $\beta = 2$, β-VAE + TopDis: $\beta = 2, \gamma = 1$, FactorVAE: $\gamma = 5$, FactorVAE + TopDis: $\gamma_1 = 5, \gamma_2 = 5$, β-TCVAE: $\beta = 6$, β-TCVAE + TopDis: $\beta = 6, \gamma = 5$, DAVA + TopDis: $\gamma = 2$;

- MPI 3D. β-VAE: $\beta = 2$, β-VAE + TopDis: $\beta = 2, \gamma = 1$, FactorVAE: $\gamma = 10$, FactorVAE + TopDis: $\gamma_1 = 5, \gamma_2 = 6$, β-TCVAE: $\beta = 6$, β-TCVAE + TopDis: $\beta = 6, \gamma = 5$, DAVA + TopDis: $\gamma = 5$;

- CelebA. FactorVAE: $\gamma = 5$, FactorVAE + TopDis: $\gamma_1 = 5, \gamma_2 = 2$;

For the ControlVAE and ControlVAE+TopDis experiments[3], we utilized the same set of relevant hyperparameters as in the FactorVAE and FactorVAE+TopDis experiments. Additionally, ControlVAE requires an expected KL loss value as a hyperparameter, which was set to KL=18, as in the original paper. It should also be noted that the requirement of an expected KL loss value is counterintuitive for an unsupervised problem, as this value depends on the number of true factors of variation. For the DAVA and DAVA + TopDis experiments[4], we used the original training procedure proposed in (Estermann & Wattenhofer, 2023), adjusting the batch size to 64 and number of iteration to 1M to match our setup. Please, refer to our GitHub repository for further details.

[3] https://github.com/shj1987/ControlVAE-ICML2020.

[4] https://github.com/besterma/dava

Table 8. Evaluation of the proposed FactorVAE + TopDis on the benchmark datasets for separate runs.

Method	FactorVAE score			MIG			SAP			DCI, dis.		
	run 1	run 2	run 3	run 1	run 2	run 3	run 1	run 2	run 3	run 1	run 2	run 3
dSprites												
FactorVAE	**0.856**	0.830	0.786	**0.341**	0.308	0.243	0.054	0.053	0.051	**0.565**	0.526	**0.509**
FactorVAE + TopDis (ours)	0.779	**0.845**	**0.847**	0.331	**0.382**	**0.360**	**0.082**	**0.081**	**0.092**	0.489	**0.571**	0.503
3D Shapes												
FactorVAE	0.901	0.893	**1.000**	0.678	0.573	**0.867**	0.055	0.067	0.175	0.780	0.772	**0.996**
FactorVAE + TopDis (ours)	**0.924**	**1.000**	**1.000**	**0.810**	**0.787**	0.739	**0.123**	**0.172**	**0.184**	**0.837**	**0.991**	0.991
3D Faces												
FactorVAE	**1.000**	**1.000**	**1.000**	0.597	0.533	0.649	**0.059**	0.048	0.076	0.843	0.840	0.861
FactorVAE + TopDis (ours)	**1.000**	**1.000**	**1.000**	**0.631**	**0.596**	**0.651**	0.058	**0.051**	**0.077**	**0.859**	**0.835**	**0.907**
MPI 3D												
FactorVAE	0.582	0.651	0.650	0.323	0.388	0.341	0.160	0.230	**0.239**	0.379	0.435	0.430
FactorVAE + TopDis (ours)	**0.696**	**0.662**	**0.674**	**0.455**	**0.416**	**0.389**	**0.283**	**0.259**	0.221	**0.505**	**0.464**	**0.437**

K. Computational Complexity

The complexity of the $\mathcal{L}_{TD}$ is formed by the calculation of RTD. For the batch size N, object dimensionality $C \times H \times W$ and latent dimensionality d, the complexity is $O(N^2(CHW + d))$, because all the pairwise distances in a batch should be calculated. The calculation of the RTD itself is often quite fast for batch sizes ≤ 256 since the boundary matrix is typically sparse for real datasets (Barannikov et al., 2022). Operations required to RTD differentiation do not take extra time. For RTD calculation and differentiation, we used GPU-optimized software.

L. Formal Definition of Representation Topology Divergence (RTD)

Data points in a high-dimensional space are often concentrated near a low-dimensional manifold (Goodfellow et al., 2016). The manifold's topological features can be represented via Vietoris-Rips simplicial complex, a union of simplices whose vertices are points at a distance smaller than a threshold α.

We define the weighted graph $\mathcal{G}$ with data points as vertices and the distances between data points $d(A_iA_j)$ as edge weights. The Vietoris-Rips complex at the threshold α is then:

$$\text{VR}_\alpha(\mathcal{G}) = \{\{A_{i_0}, \ldots, A_{i_k}\}, A_i \in \text{Vert}(\mathcal{G}) \mid d(A_iA_j) \leq \alpha\},$$

The vector space C_k consists of all formal linear combinations of the k-dimensional simplices from $\text{VR}_\alpha(\mathcal{G})$ with modulo 2 arithmetic. The boundary operators $\partial_k : C_k \to C_{k-1}$ maps each simplex to the sum of its facets. The k-th homology group $H_k = ker(\partial_k)/im(\partial_{k+1})$ represents $k-$dimensional topological features.

Choosing α is challenging, so we analyze all $\alpha > 0$. This creates a filtration of nested Vietoris-Rips complexes. We track the "birth" and "death" scales, α_b, α_d, of each topological feature, defining its persistence as $\alpha_d - \alpha_b$. The sequence of the intervals $[\alpha_b, \alpha_d]$ for basic features forms the persistence barcode (Barannikov, 1994; Chazal & Michel, 2017).

The standard persistence barcode analyzes a single point cloud X. The Representation Topology Divergence (RTD) (Barannikov et al., 2022) was introduced to measure the multi-scale topological dissimilarity between two point clouds $X, \tilde{X}$. This is done by constructing an auxilary graph $\hat{\mathcal{G}}^{w,\tilde{w}}$ whose Vietoris-Rips complex measures the difference between

Vietoris-Rips complexes $\mathrm{VR}_\alpha(\mathcal{G}^w)$ and $\mathrm{VR}_\alpha(\mathcal{G}^{\tilde{w}})$, where $w, \tilde{w}$ are the distance matrices of $X, \tilde{X}$. The auxiliary graph $\hat{\mathcal{G}}^{w,\tilde{w}}$ has the double set of vertices and the edge weights matrix $\begin{pmatrix} 0 & (w_+)^\intercal \\ w_+ & \min(w, \tilde{w}) \end{pmatrix}$, where w_+ is the w matrix with lower-triangular part replaced by $+\infty$.

The *R-Cross-Barcode*$_k(X, \tilde{X})$ is the persistence barcode of the filtered simplicial complex $\mathrm{VR}(\hat{\mathcal{G}}^{w,\tilde{w}})$. $\mathrm{RTD}_k(X, \tilde{X})$ equals the sum of intervals' lengths in *R-Cross-Barcode*$_k(X, \tilde{X})$ and measures its closeness to an empty set, with longer lifespans indicating essential features. $\mathrm{RTD}(X, \tilde{X})$ is the half-sum $\mathrm{RTD}(X, \tilde{X}) = 1/2(\mathrm{RTD}_1(X, \tilde{X}) + \mathrm{RTD}_1(\tilde{X}, X))$.

M. Symmetry Group(oid) Action

A groupoid is a mathematical structure that generalizes the concept of a group. It consists of a set G along with a partially defined binary operation. Unlike groups, the binary operation in a groupoid is not required to be defined for all pairs of elements. More formally, a groupoid is a set G together with a binary operation $\cdot : G \times G \to G$ that satisfies the following conditions for all a, b, c in G where the operations are defined: 1) Associativity: $(a \cdot b) \cdot c = a \cdot (b \cdot c)$; 2) Identity: there is an element e in G such that $a \cdot e = e \cdot a = a$ for each a in G; 3) Inverses: for each a in G, there is an element a^{-1} in G such that $a \cdot a^{-1} = a^{-1} \cdot a = e$.

A Lie groupoid is a groupoid that has additional structure of a manifold, together with smooth structure maps. These maps are required to satisfy certain properties analogous to those of a groupoid, but in a smooth category. See Weinstein (1996) for details.

N. Gradient Orthogonalization Ablation Study

Gradient orthogonalization is a technique to optimize a sum of two losses, which ensures that decreasing the second loss doesn't conflict with the decrease of the first loss. Considering our TopDis loss term L_{TD} and the reconstruction loss term L_{rec}, we take the projection of ∇L_{TD} on orthogonal space w.r.t the gradient ∇L_{rec} if their scalar product is negative. Moving within this direction allows the model parameters to get closer to the low error region for L_{TD} while preserving the reconstruction quality at the same time.

We have performed the experiments concerning the ablation study of gradient orthogonalization technique. First, we evaluate the effect of gradient orthogonalization when integrating TopDis into the classical VAE model on dSprites, see Figure 17 and Table 9. We conduct this experiment to verify the gradient orthogonalization technique in the basic setup when additional terms promoting disentanglement are absent. Second, we evaluate the effect of gradient orthogonalization when integrating TopDis to FactorVAE on the MPI3D dataset. This experiment verifies how gradient orthogonalization works for more complex data in the case of a more complicated objective. We highlight that adding the gradient orthogonalization results in lower reconstruction loss throughout the training. In particular, this may be relevant when the reconstruction quality is of high importance. Similar technique was applied for continual and multi-task learning Farajtabar et al. (2020); Suteu & Guo (2019); Yu et al. (2020).

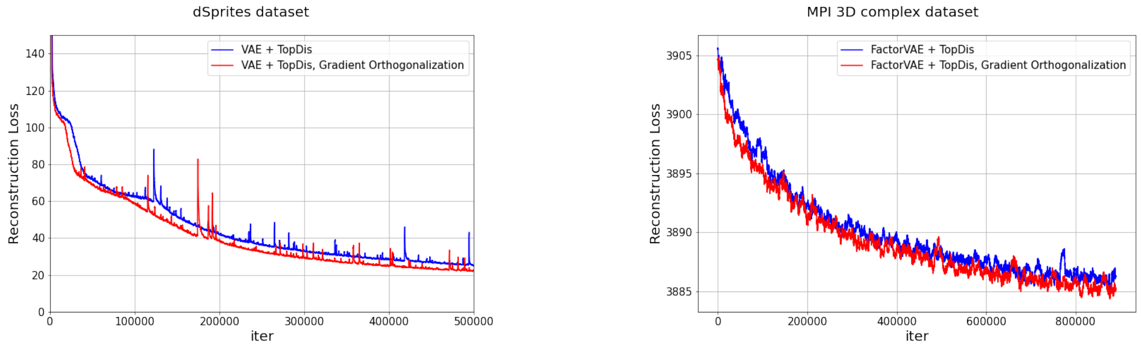


Figure 17. Effect of gradient orthogonalization on reconstruction loss. Left: VAE+TopDis, dSprites. Right: FactorVAE+TopDis, MPI 3D.

Table 9. Effect of gradient orthogonalization on disentanglement.

Method	FactorVAE	MIG	SAP	DCI, dis.
dSprites				
VAE + TopDis, no gradient orthogonalization	0.736	0.098	0.041	0.202
VAE + TopDis, gradient orthogonalization	0.723	0.121	0.031	0.229
MPI 3D				
FactorVAE + TopDis, no gradient orthogonalization	0.696	0.455	0.283	0.505
FactorVAE + TopDis, gradient orthogonalization	0.707	0.466	0.288	0.508

O. Sensitivity Analysis

The proposed TopDis approach has two hyperparameters - the weight of loss term γ (see equation 4 for details) and the value of the shift C (see equation 1 for details). We provide the sensitivity analysis w.r.t. γ for FactorVAE+TopDis on MPI3D-Real ($3 \cdot 10^5$ training iterations), please see Table 10. In Table 10, γ_{TD} denotes the weight γ for the TopDis loss from equation 4 while γ_{TC} denotes the weight for the Total Correlation loss from the FactorVAE model (see (Kim & Mnih, 2018) for details). In particular, $\gamma_{TC} = 5, \gamma_{TD} = 0$ corresponds to plain FactorVAE model. In practice, when integrating the TopDis loss, we first search for the set of the hyperparameters for the base model (VAE, FactorVAE, ControlVAE, etc.), and then tune the weight for the TopDis loss only. This strategy demonstrates the outperforming results in most cases.

Table 10. Sensitivity analysis. γ_{TD} denotes the weight for the TopDis loss (see equation 4 for details) while γ_{TC} denotes the weight for the Total Correlation loss from the FactorVAE model (see (Kim & Mnih, 2018) for details).

Method	FactorVAE	MIG	SAP	DCI, dis.	Reconstruction
FactorVAE + TopDis, MPI 3D					
$\gamma_{TC} = 5, \gamma_{TD} = 0$	0.586 ± 0.038	0.300 ± 0.020	0.184 ± 0.028	0.357 ± 0.001	3888.96 ± 0.94
$\gamma_{TC} = 5, \gamma_{TD} = 3$	0.607 ± 0.047	0.320 ± 0.003	0.193 ± 0.015	0.401 ± 0.025	3891.28 ± 0.91
$\gamma_{TC} = 5, \gamma_{TD} = 5$	0.605 ± 0.048	0.332 ± 0.033	0.205 ± 0.025	0.397 ± 0.038	3892.51 ± 1.19
$\gamma_{TC} = 5, \gamma_{TD} = 6$	0.605 ± 0.051	0.340 ± 0.035	0.207 ± 0.036	0.412 ± 0.026	3892.17 ± 0.54
$\gamma_{TC} = 5, \gamma_{TD} = 7$	0.594 ± 0.041	0.297 ± 0.042	0.183 ± 0.029	0.362 ± 0.050	3892.98 ± 0.70

Further, Table 11 provides analysis of performance for different values of C from equation 1 for FactorVAE+TopDis model on dSprites dataset (1M training iterations). In practice, we choose the value of C to be the same across all datasets, and we found the choice $C = 1/8$ demonstrates the best performance for all our experiments.

Table 11. Sensitivity analysis. C denotes the value of shift for the TopDis loss (see equation 1 for details).

Method	FactorVAE	MIG	SAP	DCI, dis.	Reconstruction
FactorVAE + TopDis, dSprites					
FactorVAE	0.819 ± 0.028	0.295 ± 0.049	0.053 ± 0.006	0.534 ± 0.029	14.65 ± 0.41
FactorVAE + TopDis, $C = 1/16$	0.779 ± 0.021	0.344 ± 0.029	0.058 ± 0.004	0.528 ± 0.032	14.71 ± 0.47
FactorVAE + TopDis, $C = 1/8$	0.824 ± 0.038	0.356 ± 0.025	0.082 ± 0.001	0.521 ± 0.044	14.72 ± 0.49
FactorVAE + TopDis, $C = 1/4$	0.820 ± 0.041	0.340 ± 0.033	0.058 ± 0.002	0.525 ± 0.062	14.85 ± 0.51

P. Group(oid) Action Versus Constant Shift Ablation

To demonstrate the relevance of the proposed shift in latent codes (see Section 4.2 for details), we perform the ablation experiment for the FactorVAE + TopDis and the MPI 3D dataset. In this ablation, we replace the proposed shift (i.e. equation 1) in the latent space with the shift which is the same for all objects in the batch. To keep the reasonable magnitude of this shift, we take the shift to be proportional to the standard deviation of the batch in a chosen latent dimension. Table 12 reveals that although the constant shift has positive effect on model's performance in comparison with FactorVAE, it results in worse performance than FactorVAE + TopDis with the proposed shift. This example illustrates the effectiveness of the proposed shift procedure.

Table 12. Ablation experiment for the proposed shift in probability space (see equation 1 for details). *Const shift* denotes the shift of a reasonable magnitude that is the same for all the objects in the batch.

Method	**FactorVAE**	**MIG**	**SAP**	**DCI, dis.**
FactorVAE + TopDis, MPI 3D				
FactorVAE	0.589 ± 0.053	0.336 ± 0.056	0.179 ± 0.052	0.391 ± 0.056
FactorVAE + TopDis	0.665 ± 0.041	0.377 ± 0.053	0.238 ± 0.040	0.438 ± 0.065
FactorVAE + TopDis const shift	0.628 ± 0.042	0.342 ± 0.059	0.209 ± 0.033	0.418 ± 0.038

Q. RTD Differentiation

Here we gather details on RTD differentiation in order to use RTD as a loss in neural networks.

Define Σ as the set of all simplices in the filtration of the graph $VR(\hat{\mathcal{G}}^{w,\tilde{w}})$, and $\mathcal{T}_k$ as the set of all segments in *R-Cross-Barcode*$_k(X, \hat{X})$. Fix (an arbitrary) strict order on $\mathcal{T}_k$.

There exists a function $f_k : \{b_i, d_i\}_{(b_i,d_i)\in\mathcal{T}_k} \to \Sigma$ that maps b_i (or d_i) to simplices σ (or τ) whose addition leads to "birth" (or "death") of the corresponding homological class.

Thus, we may obtain the following equation for subgradient

$$\frac{\partial \,\mathrm{RTD}(X,\hat{X})}{\partial\sigma} = \sum_{i\in\mathcal{T}_k} \frac{\partial \mathrm{RTD}(X,\hat{X})}{\partial b_i}\mathbb{I}\{f_k(b_i)=\sigma\} + \sum_{i\in\mathcal{T}_k} \frac{\partial \mathrm{RTD}(X,\hat{X})}{\partial d_i}\mathbb{I}\{f_k(d_i)=\sigma\}$$

Here, for any σ no more than one term has non-zero indicator.

b_i and d_i are just the filtration values at which simplices $f_k(b_i)$ and $f_k(d_i)$ join the filtration. They depend on weights of graph edges as

$$g_k(\sigma) = \max_{i,j\in\sigma} m_{i,j}$$

This function is differentiable (Leygonie et al., 2021) and so is $f_k \circ g_k$. Thus we obtain the subgradient:

$$\frac{\partial \,\mathrm{RTD}(X,\hat{X})}{\partial m_{i,j}} = \sum_{\sigma\in\Sigma} \frac{\partial \,\mathrm{RTD}(X,\hat{X})}{\partial\sigma} \frac{\partial\sigma}{\partial m_{i,j}}.$$

The only thing that is left is to obtain subgradients of $\mathrm{RTD}(X,\hat{X})$ by points from X and $\hat{X}$. Consider (an arbitrary) element $m_{i,j}$ of matrix m. There are 4 possible scenarios:

1. $i, j \leq N$, in other words $m_{i,j}$ is from the upper-left quadrant of m. Its length is constant and thus $\forall l : \frac{\partial m_{i,j}}{\partial X_l} = \frac{\partial m_{i,j}}{\partial \hat{X}_l} = 0$.
2. $i \leq N < j$, in other words $m_{i,j}$ is from the upper-right quadrant of m. Its length is computed as Euclidean distance and thus $\frac{\partial m_{i,j}}{\partial X_i} = \frac{X_i - X_{j-N}}{\|X_i - X_{j-N}\|_2}$ (similar for X_{N-j}).

3. $j \leq N < i$, similar to the previous case.

4. $N < i, j$, in other words $m_{i,j}$ is from the bottom-right quadrant of m. Here we have subgradients like

$$\frac{\partial m_{i,j}}{\partial X_{i-N}} = \frac{X_{i-N} - X_{j-N}}{\|X_{i-N} - X_{j-N}\|_2} \mathbb{I}\{w_{i-N,j-N} < \hat{w}_{i-N,j-N}\}$$

   Similar for $X_{j-N}, \hat{X}_{i-N}$ and $\hat{X}_{j-N}$.

Subgradients $\frac{\partial \text{ RTD}(X,\hat{X})}{\partial X_i}$ and $\frac{\partial \text{ RTD}(X,\hat{X})}{\partial \hat{X}_i}$ can be derived from the before mentioned using the chain rule and the formula of full (sub)gradient. Now we are able to minimize RTD$(X, \hat{X})$ by methods of (sub)gradient optimization.

R. Discussing the Definition of Disentangled Representation.

Let $X \subset \mathbb{R}^{N_x \times N_y}$ denotes the dataset consisting of $N_x \times N_y$ pixels pictures containing a disk of various color with fixed disk radius r and the center of the disks situated at an arbitrary point x, y. Denote ρ_X the uniform distribution over the coordinates of centers of the disks and the colors. Let $G_x \times G_y \times G_c$ be the commutative group of symmetries of this data distribution, $G_x \times G_y$ is the position change acting (locally) via

$$(a, b) : (x, y, c) \mapsto (x + a, y + b, c)$$

and G_z is changing the colour along the colour circle $\theta : (x, y, c) \mapsto (x, y, c + \theta \mod 2\pi)$. Contrary to Higgins et al. (2018), section 3, we do not assume the gluing of the opposite sides of our pictures, which is closer to real world situations. Notice that, as a consequence of this, each group element from $G_x \times G_y$ can act only on a subset of X, so that the result is still situated inside $N_x \times N_y$ pixels picture. This mathematical structure when each group element has its own set of points on which it acts, is called groupoid, we discuss this notion in more details in Appendix M.

The outcome of disentangled learning in such case are the encoder $h : X \to Z$ and the decoder $f : Z \to X$ maps with $Z = \mathbb{R}^3$, $f \circ h = \text{Id}$, together with symmetry group(oid) G actions on X and Z, such that a) the encoder-decoder maps preserve the distributions, which are the distribution ρ_X describing the dataset X and the standard in VAE learning $N(0, 1)$ distribution in latent space Z; b) the decoder and the encoder maps are equivariant with respect to the symmetry group(oid) action, where the action on the latent space is defined as shifts of latent variables; the group action preserves the dataset distribution X therefore the group(oid) action shifts on the latent space must preserve the standard $N(0, 1)$ distribution on latent coordinates, i.e. they must act via the formula 1.

Connection with disentangled representations in which the symmetry group latent space action is *linear*. The normal distribution arises naturally as the projection to an axis of the uniform distribution on a very high dimensional sphere $S^N \subset \mathbb{R}^{N+1}$. Let a general symmetry compact Lie group $\hat{G}$ acts linearly on $\mathbb{R}^{N+1}$ and preserves the sphere S^N. Let G^{ab} be a maximal commutative subgroup in G. Then the ambient space $\mathbb{R}^{N+1}$ decomposes into direct sum of subspaces $\mathbb{R}^{N+1} = \oplus_\alpha Z_\alpha$, on which $G^{ab} = \Pi_i G_i$, acts via rotations in two-dimensional space, and the orbit of this action is a circle $S^1 \subset S^N$. If one chooses an axis in each such two-dimensional space then the projection to this axis gives a coordinate on the sphere S^N. And the group action of G^{ab} decomposes into independent actions along these axes. In such a way, the disentangled representation in the sense of Section 4.1 can be obtained from the data representation with uniform distribution on the sphere/disk on which the symmetry group action is linear, and vice versa.

S. Experiments with Correlated Factors

Table 13 shows experimental results for disentanglement learning with confounders - one factor correlated with all others. The addition of the TopDis loss results in a consistent improvement of all quality measures. For experiments, we used the implementation of the "shared confounders" distribution from (Roth et al., 2023)[5] and the same hyperparameters as for the rest of experiments.

[5] https://github.com/facebookresearch/disentangling-correlated-factors

Table 13. Evaluation on the benchmark datasets with correlated factors

Method	**FactorVAE score**	**MIG**	**SAP**	**DCI, dis.**
		dSprites		
FactorVAE	0.803 ± 0.055	0.086 ± 0.026	0.030 ± 0.010	0.216 ± 0.044
FactorVAE + TopDis (ours)	$\mathbf{0.840 \pm 0.011}$	$\mathbf{0.103 \pm 0.019}$	$\mathbf{0.044 \pm 0.014}$	$\mathbf{0.270 \pm 0.002}$
		3D Shapes		
FactorVAE	0.949 ± 0.67	0.363 ± 0.100	0.083 ± 0.004	0.477 ± 0.116
FactorVAE + TopDis (ours)	$\mathbf{0.998 \pm 0.001}$	$\mathbf{0.403 \pm 0.091}$	$\mathbf{0.112 \pm 0.013}$	$\mathbf{0.623 \pm 0.026}$

Table 14. Evaluation on the benchmark datasets for VAE + TopDis

Method	**FactorVAE score**	**MIG**	**SAP**	**DCI, dis.**
		dSprites		
VAE	0.781 ± 0.016	0.170 ± 0.072	0.057 ± 0.039	0.314 ± 0.072
VAE + TopDis (ours)	$\mathbf{0.833 \pm 0.068}$	$\mathbf{0.200 \pm 0.119}$	$\mathbf{0.065 \pm 0.009}$	$\mathbf{0.394 \pm 0.132}$
		3D Shapes		
VAE	$\mathbf{1.0 \pm 0.0}$	0.729 ± 0.070	0.160 ± 0.050	0.952 ± 0.023
VAE + TopDis (ours)	$\mathbf{1.0 \pm 0.0}$	$\mathbf{0.835 \pm 0.012}$	$\mathbf{0.216 \pm 0.020}$	$\mathbf{0.977 \pm 0.023}$
		3D Faces		
VAE	0.96 ± 0.03	0.525 ± 0.051	0.059 ± 0.013	0.813 ± 0.063
VAE + TopDis (ours)	$\mathbf{1.0 \pm 0.0}$	$\mathbf{0.539 \pm 0.037}$	$\mathbf{0.063 \pm 0.011}$	$\mathbf{0.831 \pm 0.023}$
		MPI 3D		
VAE	0.556 ± 0.081	0.280 ± 0.059	0.167 ± 0.064	0.346 ± 0.029
VAE + TopDis (ours)	$\mathbf{0.595 \pm 0.055}$	$\mathbf{0.358 \pm 0.022}$	$\mathbf{0.229 \pm 0.022}$	$\mathbf{0.407 \pm 0.025}$

T. Experiments with VAE + TopDis

In order to verify the proposed TopDis as a self-sufficient loss contributing to disentanglement, we add TopDis to the classical VAE (Kingma & Welling, 2013) objective as an additional loss term. As demonstrated by quantitative evaluation on the benchmark datasets in Table 14, the addition of TopDis loss improves the quality of disentanglement as measured by FactorVAE score, MIG, SAP, DCI: on dSprites up to $+6\%$, $+17\%$, $+14\%$, $+25\%$, on 3D Shapes up to $+14\%$, $+35\%$, $+2\%$, on 3D Faces up to $+4\%$, $+2\%$, $+6\%$, $+2\%$, on MPI 3D up to $+7\%$, $+27\%$, $+37\%$, $+17\%$. For all the datasets and metrics, VAE+TopDis outperforms the classical VAE model. Besides, VAE+TopDis preserves the reconstruction quality as revealed by Table 4.

U. Motivation for Topological Feature Distance in Learning Disentangled Representations

To connect topological features with disentangled representations, the continuity and invertibility of the Lie transformations on the model distribution support are the key properties. Such transformations, known as homeomorphisms, induce homology isomorphisms and hence preserve topological features. Minimizing TopDis ensures maximal preservation of these features by the Lie action. The following proposition further strengthens this relationship.

Let $W(RLT(d,k), RLT(d,k'))$ denotes the Wasserstein distance between the RLT's of two data submanifolds conditioned on two values k and k' of a generative factor z_d (Zhou et al., 2021). The proposition below proves that minimizing TopDis ensures a small topological distance between conditioned submanifolds, which is a requirement for disentanglement according to (Zhou et al., 2021).

Proposition U.1. *If $TopDis_i(z,a) < \epsilon$, for $a \leq \frac{1}{8}$, $i = 0, 1$, where $TopDis_i(z,a)$ is the $TopDis_i$ loss with the shift*

$z \to z'$ along a generative factor z_d with parameter $C = a$, then the Wasserstein distance between the RLT's for two data submanifolds conditioned on two values k, k' of z_d satisfies $W(RLT(d,k), RLT(d,k')) < \frac{16}{\alpha_{max}}\epsilon$, where α_{max} is the constant from the definition of RLT.

Proof. Given a sample z from the data submanifold with the generative factor z_d conditioned on $z_d = k$, a shift along z_d with some C' produces from z a sample with the generative factor z_d conditioned on $z_d = k'$. For a small shift with $C' \leq \frac{1}{8}$, the Wasserstein distance between RLT's of z and z' is bounded by $\frac{1}{\alpha_{max}}(TopDis_0(z,C') + TopDis_1(z,C')) < \frac{2}{\alpha_{max}}\epsilon$. An arbitrary shift can be decomposed into no more than 8 smaller shifts with $C' \leq \frac{1}{8}$, and then the Wasserstein distance between RLT's of z and z' is bounded similarly by $\frac{16}{\alpha_{max}}\epsilon$.

V. Comparison with Previous Works on RTD

The Representation Topology Divergence (RTD) was introduced in the paper (Barannikov et al., 2022), as a tool for comparing two neural representations of the same set of objects. An application of RTD to evaluation of interpretable directions in a simple synthetic dataset is described briefly in loc.cit. The experiment in loc.cit. involves comparing the topological dissimilarity in data submanifolds corresponding to slices in the latent space. While in the current paper, we use axis-aligned traversals and samples from the whole data manifold. One of the crucial difference with loc.cit. is that, in the current paper, we are able to apply our topological loss TopDis (equation 3) directly on the outcome of the decoder during the VAE optimization. This permits to propagate the gradients from the topological loss to the variational autoencoder weights. In Barannikov et al. (2022), RTD was used as a metric between two submanifolds conditioned on two values of a generative factor, in a static situation of an already disentangled model. During learning the outcome of the decoder never lies on such conditioned submanifold. This has led us further to the Lie group(oid) action and the Proposition 4.2 establishing equivalence of two approaches to disentanglement, via factorized distributions and via the Lie symmetry actions. We show in addition how to preserve the reconstruction quality via gradient orthogonalization. The effectiveness of gradient orthogonalization to improve the reconstruction (see Figure 17) and the disentanglement (Table 9) is validated via ablation study. In the paper Trofimov et al. (2023), the RTD is used as an additional loss for dimensionality reduction, in order to preserve topology of original data in a latent space.

W. A Remark on Inductive Bias

In Locatello et al. (2020), the authors have shown that unsupervised disentanglement learning without any inductive bias is impossible. Next, we explain in what sense the proposed TopDis approach can be seen as an inductive bias.

Each factor in a disentangled model has a symmetry Lie action that fixes other factors, and therefore each factor encodes variations that must cause only topologically mild changes in the pixel space. It's important to keep in mind that 'topologically' refers here to the topology of the clouds of data points in pixel space, rather than the topology of an object depicted on a specific dataset image. Regarding a categorical factor whose change produces objects in the pictures with objects' topology that might be dramatically different, such factors are expected to correspond to discrete symmetries on data manifolds. However, in our approach, these discrete symmetries are included into continuous families that correspond to continuous interpolation between the objects categories. These symmetry actions guarantee that such interpolations are the same for objects, for example, positioned at varying locations within an image. So a useful inductive bias is the fact that all disentangled factors cause continuous changes. For categorical factors, which might inherently possess only discrete symmetries, these discrete symmetries can be integrated into continuous interpolating families of symmetries. It's possible that these factors may exhibit slightly larger TopDis because data points corresponding to interpolations between dramatically different objects are absent from the dataset, potentially requiring more iterations to learn interpolations between object categories in a symmetrically uniform way.

X. Visualization of Latent Traversals

Images obtained from selected latent traversal exhibiting the most differences are presented in Figure 18 (FactorVAE, FactorVAE+TopDis trained on dSprites, 3D shapes, MPI 3D, 3D Faces) and Figure 19 (FactorVAE, FactorVAE+TopDis trained on CelebA).

Figures 20, 21, 22, 23, 24 shows latent traversals along all axes.

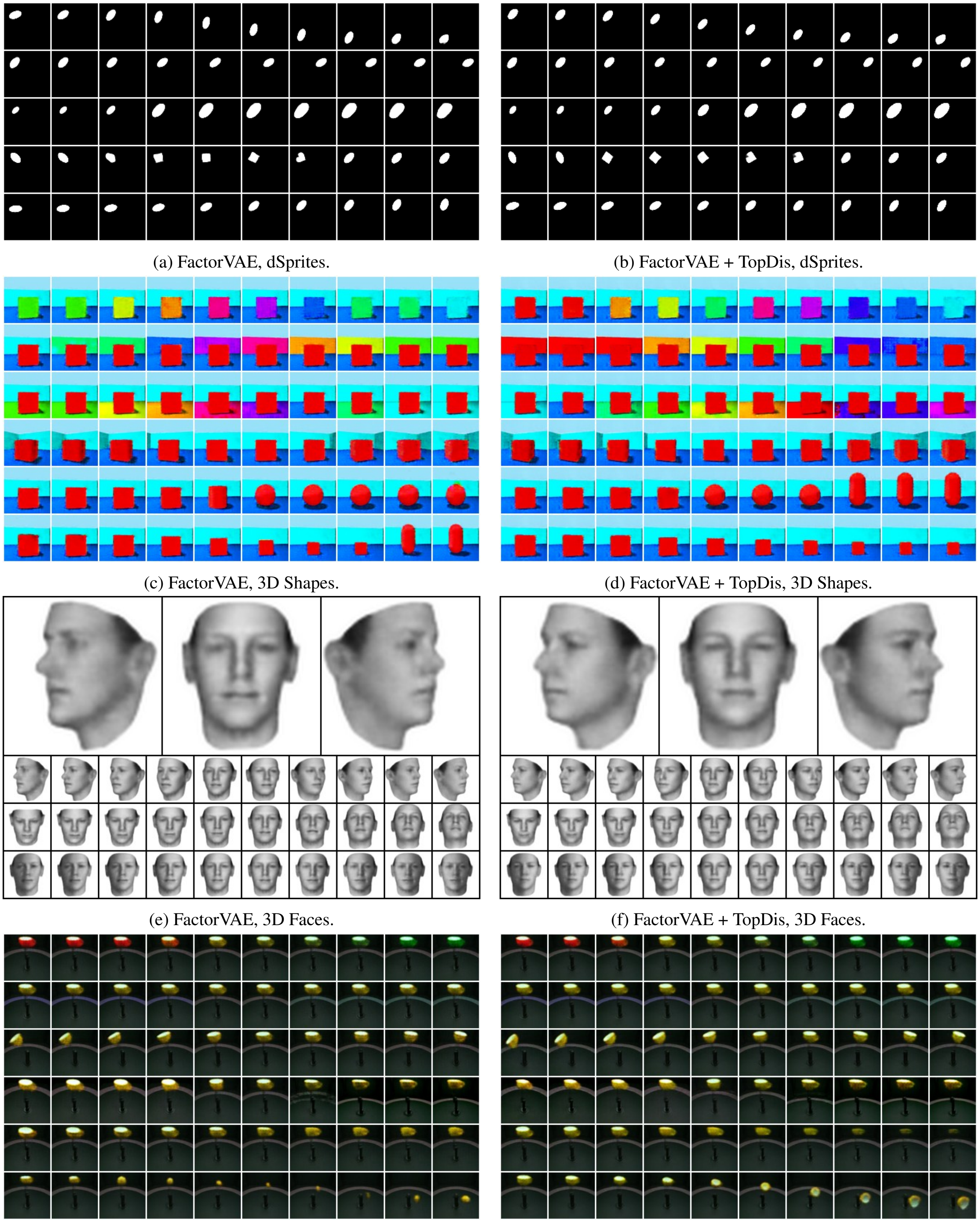
dSprites. Figures 18a and 18b show that the TopDis loss helps to outperform simple FactorVAE in terms of visual perception. The simple FactorVAE model has entangled rotation and shift along axes (rows 1,2,5 in Figure 18a), even though the Total Correlation in both models is minimal, which demonstrates the impact of the proposed topological objective.

3D Shapes. Figures 18c and 18d show that proposed TopDis loss leads to the superior disentanglement of the factors as compared to the simple FactorVAE model, where the shape and the scale factors remain entangled in the last row.

3D Faces. FactorVAE+TopDis (Figure 18f) outperforms FactorVAE (Figure 18e) in terms of disentangling the main factors such as azimuth, elevation, and lighting from facial identity. On top of these figures we highlight the azimuth traversal. The advantage of TopDis is seen from the observed preservations in facial attributes such as the chin, nose, and eyes.

MPI 3D. Here, the entanglement between the size and elevation factors is particularly evident when comparing the bottom two rows of Figures 18g and 18h. In contrast to the base FactorVAE, which left these factors entangled, our TopDis method successfully disentangles them.

CelebA. For this dataset, we show the most significant improvements obtained by adding the TopDis loss in Figure 19. The TopDis loss improves disentanglement of skin tone and lightning compared to basic FactorVAE, where these factor are entangled with other factors - background and hairstyle.



(a) FactorVAE, dSprites.

(b) FactorVAE + TopDis, dSprites.

(c) FactorVAE, 3D Shapes.

(d) FactorVAE + TopDis, 3D Shapes.

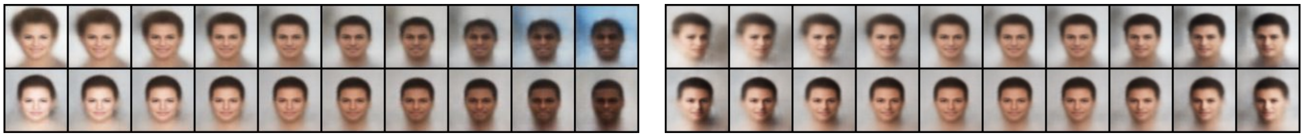
(e) FactorVAE, 3D Faces.

(f) FactorVAE + TopDis, 3D Faces.

(g) FactorVAE, MPI 3D.

(h) FactorVAE + TopDis, MPI 3D.

Figure 18. FactorVAE and FactorVAE + TopDis latent traversals.



(a) Skin tone

(b) Lightning

Figure 19. Visual improvement from addition of TopDis, CelebA. Top: FactorVAE, bottom: FactorVAE + TopDis.

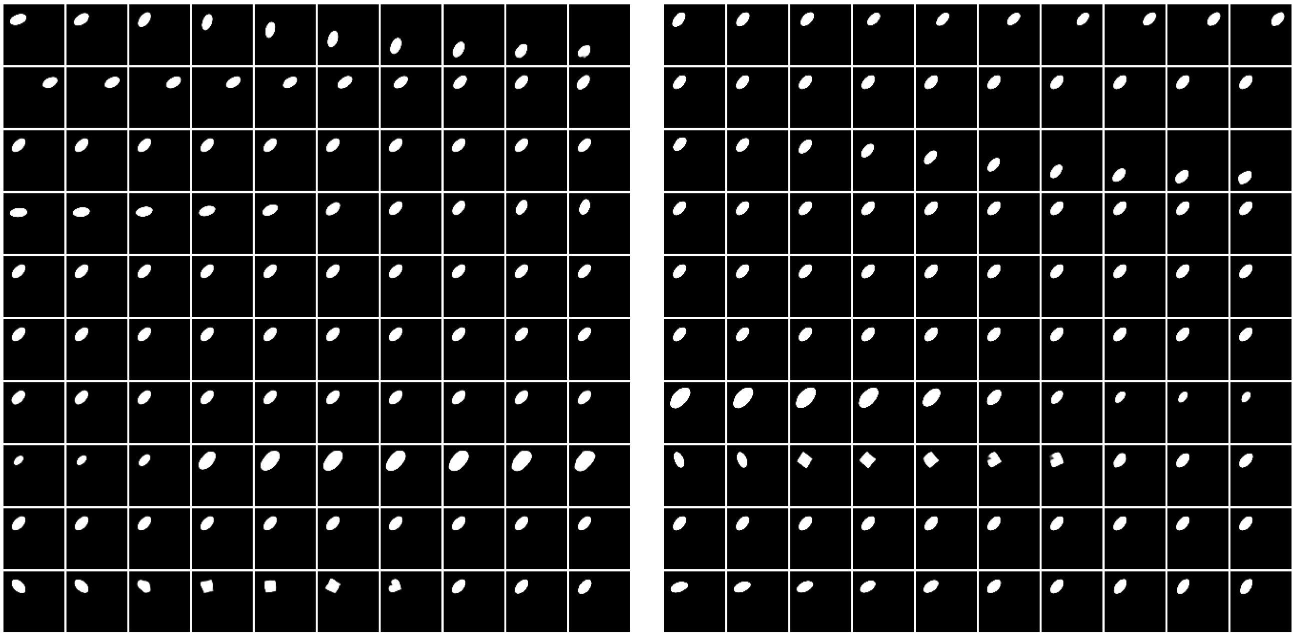


Figure 20. FactorVAE (left) and FactorVAE + TopDis (right) latent traversals, dSprites.

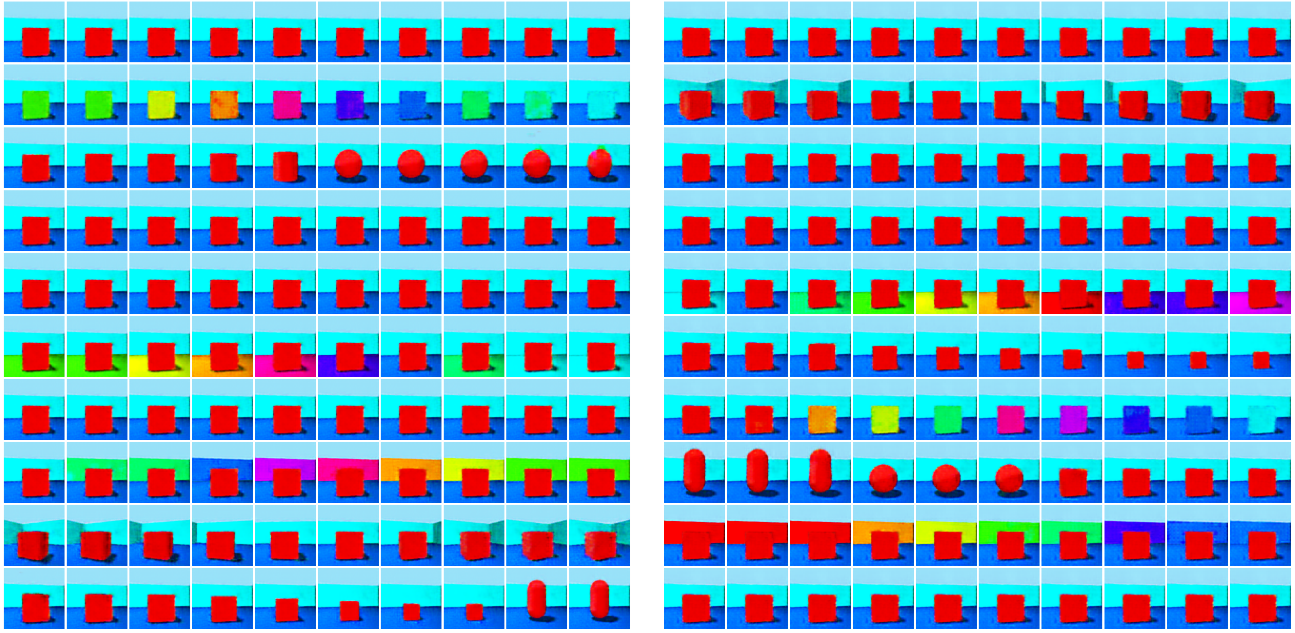


Figure 21. FactorVAE (left) and FactorVAE + TopDis (right) latent traversals, 3D Shapes.

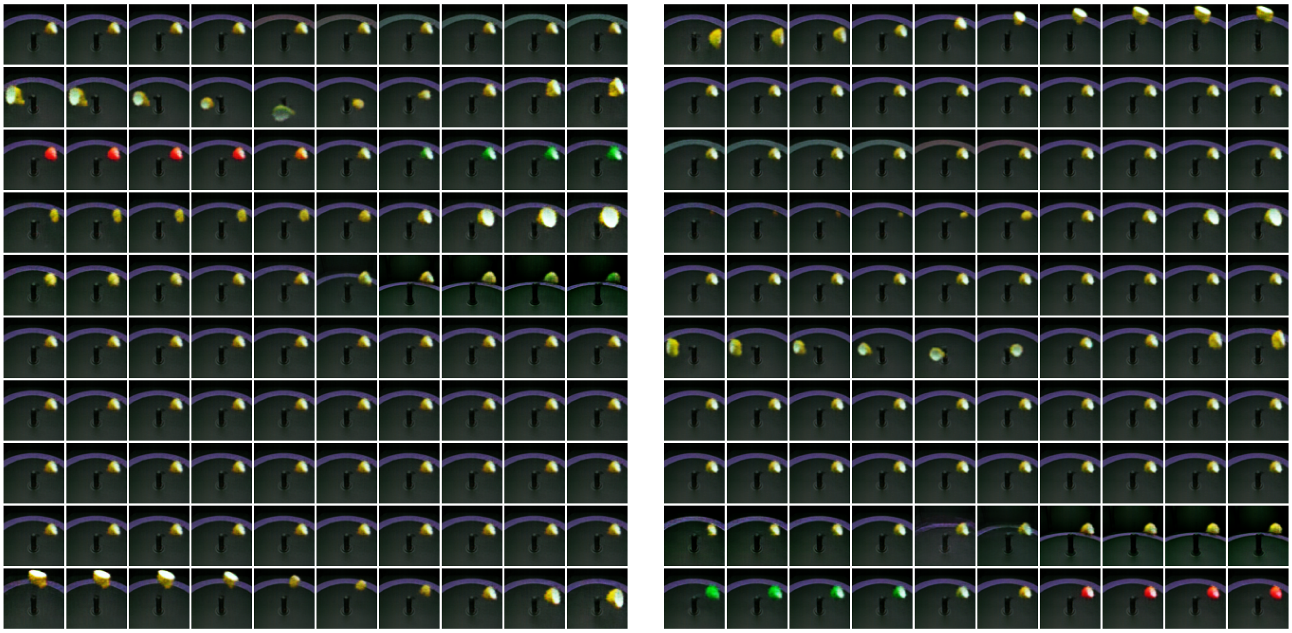


Figure 22. FactorVAE (left) and FactorVAE + TopDis (right) latent traversals, MPI 3D.

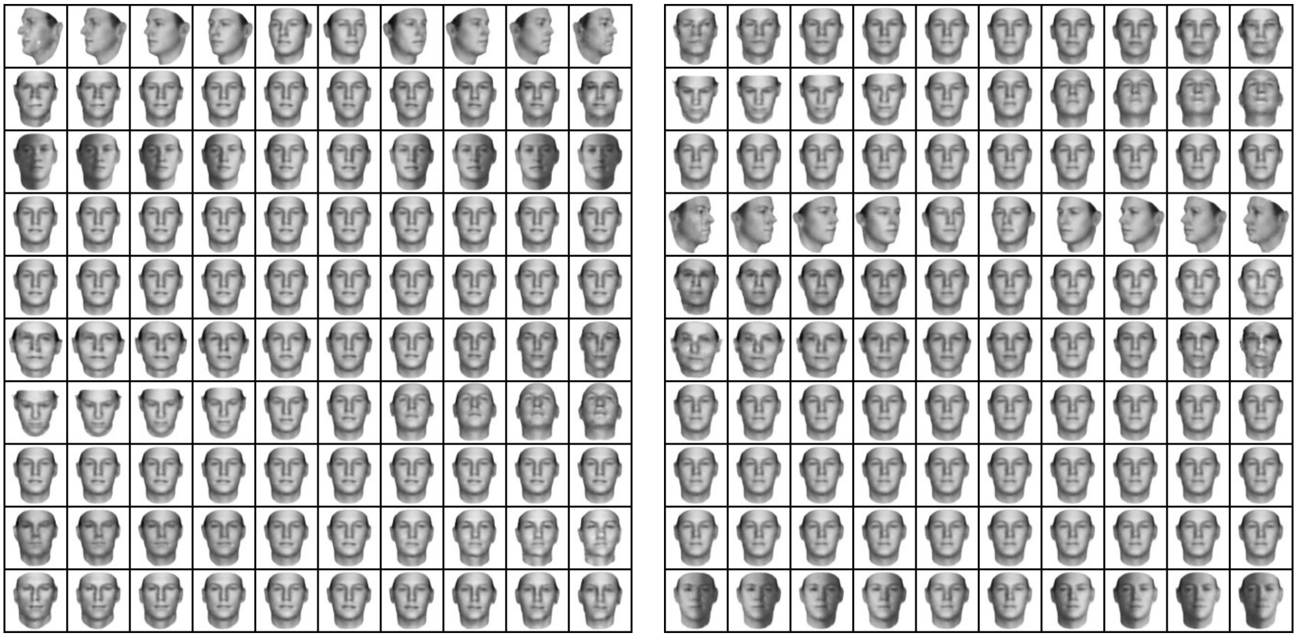


Figure 23. FactorVAE (left) and FactorVAE + TopDis (right) latent traversals, 3D Faces.

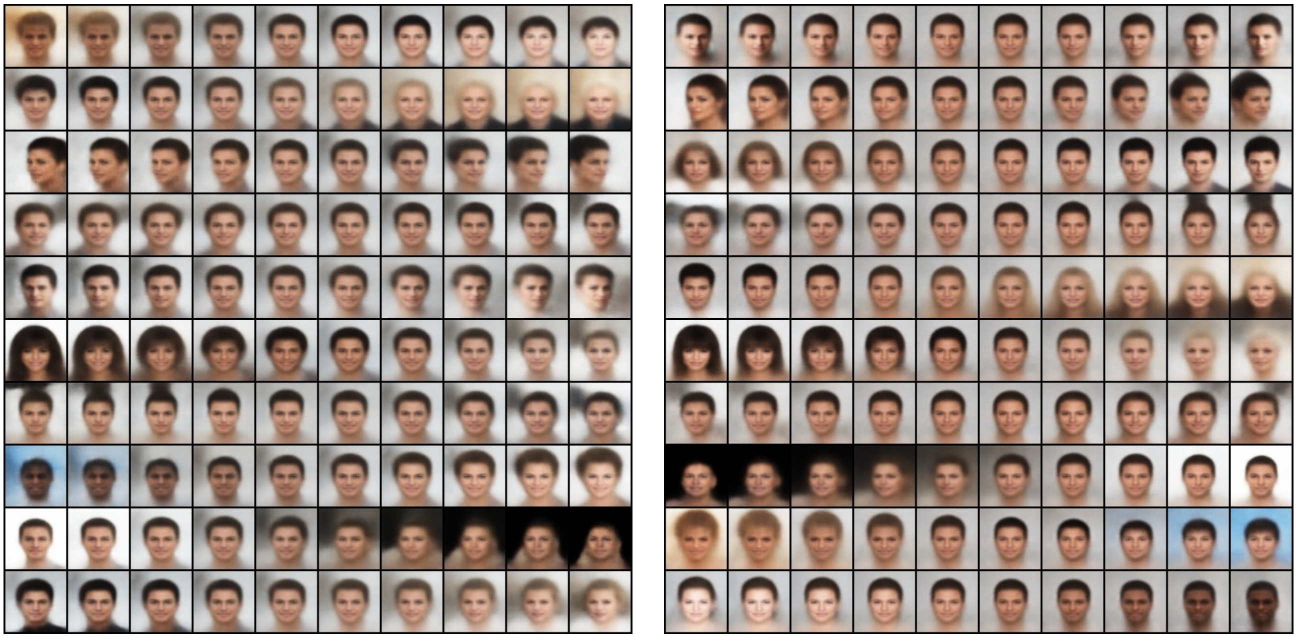


Figure 24. FactorVAE (left) and FactorVAE + TopDis (right) latent traversals, CelebA.