This CVPR 2020 paper is the Open Access version, provided by the Computer Vision Foundation.
Except for this watermark, it is identical to the accepted version;
the final published version of the proceedings is available on IEEE Xplore.

Guided Variational Autoencoder for Disentanglement Learning

Zheng Ding[*,1,2], Yifan Xu[*,2], Weijian Xu[2], Gaurav Parmar[2], Yang Yang[3], Max Welling[3,4], Zhuowen Tu[2]
[1]Tsinghua University [2]UC San Diego [3]Qualcomm, Inc. [4]University of Amsterdam

Abstract

We propose an algorithm, guided variational autoencoder (Guided-VAE), that is able to learn a controllable generative model by performing latent representation disentanglement learning. The learning objective is achieved by providing signals to the latent encoding/embedding in VAE without changing its main backbone architecture, hence retaining the desirable properties of the VAE. We design an unsupervised strategy and a supervised strategy in Guided-VAE and observe enhanced modeling and controlling capability over the vanilla VAE. In the unsupervised strategy, we guide the VAE learning by introducing a lightweight decoder that learns latent geometric transformation and principal components; in the supervised strategy, we use an adversarial excitation and inhibition mechanism to encourage the disentanglement of the latent variables. Guided-VAE enjoys its transparency and simplicity for the general representation learning task, as well as disentanglement learning. On a number of experiments for representation learning, improved synthesis/sampling, better disentanglement for classification, and reduced classification errors in meta learning have been observed.

1. Introduction

The resurgence of autoencoders (AE) [34, 6, 21] is an important component in the rapid development of modern deep learning [17]. Autoencoders have been widely adopted for modeling signals and images [46, 50]. Its statistical counterpart, the variational autoencoder (VAE) [29], has led to a recent wave of development in generative modeling due to its two-in-one capability, both representation and statistical learning in a single framework. Another exploding direction in generative modeling includes generative adversarial networks (GAN) [18], but GANs focus on the generation process and are not aimed at representation learning (without an encoder at least in its vanilla version).

Compared with classical dimensionality reduction methods like principal component analysis (PCA) [22, 27] and Laplacian eigenmaps [4], VAEs have demonstrated their unprecedented power in modeling high dimensional data of real-world complexity. However, there is still a large room to improve for VAEs to achieve a high quality reconstruction/synthesis. Additionally, it is desirable to make the VAE representation learning more transparent, interpretable, and controllable.

In this paper, we attempt to learn a transparent representation by introducing guidance to the latent variables in a VAE. We design two strategies for our Guided-VAE, an unsupervised version (Fig. 1.a) and a supervised version (Fig. 1.b). The main motivation behind Guided-VAE is to encourage the latent representation to be semantically interpretable, while maintaining the integrity of the basic VAE architecture. Guided-VAE is learned in a multi-task learning fashion. The objective is achieved by taking advantage of the modeling flexibility and the large solution space of the VAE under a lightweight target. Thus the two tasks, learning a good VAE and making the latent variables controllable, become companions rather than conflicts.

In **unsupervised Guided-VAE**, in addition to the standard VAE backbone, we also explicitly force the latent variables to go through a lightweight encoder that learns a deformable PCA. As seen in Fig. 1.a, two decoders exist, both trying to reconstruct the input data $\mathbf{x}$: The main decoder, denoted as Dec_{main}, functions regularly as in the standard VAE [29]; the secondary decoder, denoted as Dec_{sub}, explicitly learns a geometric deformation together with a linear subspace. In **supervised Guided-VAE**, we introduce a subtask for the VAE by forcing one latent variable to be discriminative (minimizing the classification error) while making the rest of the latent variable to be adversarially discriminative (maximizing the minimal classification error). This subtask is achieved using an adversarial excitation and inhibition formulation. Similar to the unsupervised Guided-VAE, the training process is carried out in an end-to-end multi-task learning manner. The result is a regular generative model that keeps the original VAE properties intact, while having the specified latent variable semantically meaningful and capable of controlling/synthesizing a specific attribute. We apply Guided-VAE to the data modeling and few-shot learning problems and show favorable results

[*]Authors contributed equally.

on the MNIST, CelebA, CIFAR10 and Omniglot datasets.

The contributions of our work can be summarized as follows:

- We propose a new generative model disentanglement learning method by introducing latent variable guidance to variational autoencoders (VAE). Both unsupervised and supervised versions of Guided-VAE have been developed.
- In unsupervised Guided-VAE, we introduce deformable PCA as a subtask to guide the general VAE learning process, making the latent variables interpretable and controllable.
- In supervised Guided-VAE, we use an adversarial excitation and inhibition mechanism to encourage the disentanglement, informativeness, and controllability of the latent variables.

Guided-VAE can be trained in an end-to-end fashion. It is able to keep the attractive properties of the VAE while significantly improving the controllability of the vanilla VAE. It is applicable to a range of problems for generative modeling and representation learning.

2. Related Work

Related work can be discussed along several directions.

Generative model families such as generative adversarial networks (GAN) [18, 2] and variational autoencoder (VAE) [29] have received a tremendous amount of attention lately. Although GAN produces higher quality synthesis than VAE, GAN is missing the encoder part and hence is not directly suited for representation learning. Here, we focus on disentanglement learning by making VAE more controllable and transparent.

Disentanglement learning [41, 48, 23, 1, 16, 26] recently becomes a popular topic in representation learning. Adversarial training has been adopted in approaches such as [41, 48]. Various methods [44, 28, 37] have imposed constraints/regularizations/supervisions to the latent variables, but these existing approaches often involve an architectural change to the VAE backbone and the additional components in these approaches are not provided as secondary decoder for guiding the main encoder. A closely related work is the β-VAE [20] approach in which a balancing term β is introduced to control the capacity and the independence prior. β-TCVAE [8] further extends β-VAE by introducing a total correlation term.

From a different angle, principal component analysis (PCA) family [22, 27, 7] can also be viewed as representation learning. Connections between robust PCA [7] and VAE [29] have been observed [10]. Although being a widely adopted method, PCA nevertheless has limited modeling capability due to its linear subspace assumption. To alleviate the strong requirement for the input data being pre-aligned, RASL [45] deals with unaligned data by estimating a hidden transformation to each input. Here, we take advantage of the transparency of PCA and the modeling power of VAE by developing a sub-encoder (see Fig. 1.a), deformable PCA, that guides the VAE training process in an integrated end-to-end manner. After training, the sub-encoder can be removed by keeping the main VAE backbone only.

To achieve disentanglement learning in supervised Guided-VAE, we encourage one latent variable to directly correspond to an attribute while making the rest of the variables uncorrelated. This is analogous to the excitation-inhibition mechanism [43, 53] or the explaining-away [52] phenomena. Existing approaches [38, 37] impose supervision as a conditional model for an image translation task, whereas our supervised Guided-VAE model targets the generic generative modeling task by using an adversarial excitation and inhibition formulation. This is achieved by minimizing the discriminative loss for the desired latent variable while maximizing the minimal classification error for the rest of the variables. Our formulation has a connection to the domain-adversarial neural networks (DANN) [15], but the two methods differ in purpose and classification formulation. Supervised Guided-VAE is also related to the adversarial autoencoder approach [40], but the two methods differ in the objective, formulation, network structure, and task domain. In [24], the domain invariant variational autoencoders method (DIVA) differs from ours by enforcing disjoint sectors to explain certain attributes.

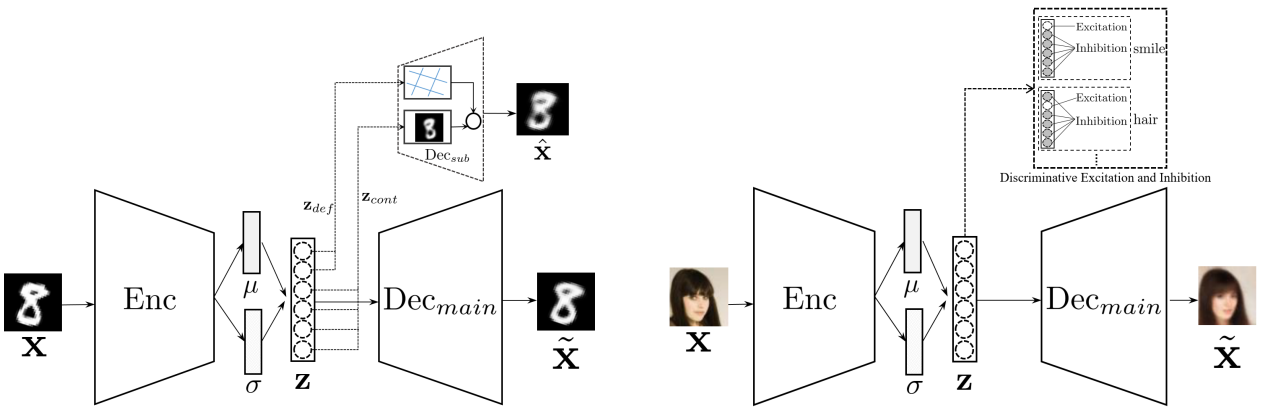
Our model also has connections to the deeply-supervised nets (DSN) [36], where intermediate supervision is added to a standard CNN classifier. There are also approaches [14, 5] in which latent variables constraints are added, but they have different formulations and objectives than Guided-VAE. Recent efforts in fairness disentanglement learning [9, 47] also bear some similarity, but there is still a large difference in formulation.

3. Guided-VAE Model

In this section, we present the main formulations of our Guided-VAE models. The unsupervised Guided-VAE version is presented first, followed by introduction of the supervised version.

3.1. VAE

Following the standard definition in variational autoencoder (VAE) [29], a set of input data is denoted as $\mathrm{X} = (\mathbf{x}_1, ..., \mathbf{x}_n)$ where n denotes the number of total input samples. The latent variables are denoted by vector $\mathbf{z}$. The encoder network includes network and variational parameters $\boldsymbol{\phi}$ that produces variational probability model $q_{\boldsymbol{\phi}}(\mathbf{z}|\mathbf{x})$. The decoder network is parameterized by $\boldsymbol{\theta}$ to reconstruct



(a) Unsupervised Guided-VAE (b) Supervised Guided-VAE

Figure 1. Model architecture for the proposed Guided-VAE algorithms.

sample $\tilde{\mathbf{x}} = f_{\boldsymbol{\theta}}(\mathbf{z})$. The log likelihood $\log p(\mathbf{x})$ estimation is achieved by maximizing the Evidence Lower BOund (ELBO) [29]:

$$\begin{aligned} ELBO(\boldsymbol{\theta}, \boldsymbol{\phi}; \mathbf{x}) = \mathbb{E}_{q_{\boldsymbol{\phi}}(\mathbf{z}|\mathbf{x})}[\log(p_{\boldsymbol{\theta}}(\mathbf{x}|\mathbf{z}))] \\ - D_{\mathrm{KL}}(q_{\boldsymbol{\phi}}(\mathbf{z}|\mathbf{x})||p(\mathbf{z})). \end{aligned} \tag{1}$$

The first term in Eq. (1) corresponds to a reconstruction loss $\int q_{\boldsymbol{\phi}}(\mathbf{z}|\mathbf{x}) \times ||\mathbf{x} - f_{\boldsymbol{\theta}}(\mathbf{z})||^2 d\mathbf{z}$ (the first term is the *negative* of reconstruction loss between input $\mathbf{x}$ and reconstruction $\tilde{\mathbf{x}}$) under Gaussian parameterization of the output. The second term in Eq. (1) refers to the KL divergence between the variational distribution $q_{\boldsymbol{\phi}}(\mathbf{z}|\mathbf{x})$ and the prior distribution $p(\mathbf{z})$. The training process thus tries to optimize:

$$\max_{\boldsymbol{\theta}, \boldsymbol{\phi}} \left\{ \sum_{i=1}^{n} ELBO(\boldsymbol{\theta}, \boldsymbol{\phi}; \mathbf{x}_i) \right\}. \tag{2}$$

3.2. Unsupervised Guided-VAE

In our unsupervised Guided-VAE, we introduce a deformable PCA as a secondary decoder to guide the VAE training. An illustration can be seen in Fig. 1.a. This secondary decoder is called Dec_{sub}. Without loss of generality, we let $\mathbf{z} = (\mathbf{z}_{def}, \mathbf{z}_{cont})$. $\mathbf{z}_{def}$ decides a deformation/transformation field, e.g. an affine transformation denoted as $\tau(\mathbf{z}_{def})$. $\mathbf{z}_{cont}$ determines the content of a sample image for transformation. The PCA model consists of K basis $B = (\mathbf{b}_1, ..., \mathbf{b}_K)$. We define a deformable PCA loss as:

$$\begin{aligned} &\mathcal{L}_{DPCA}(\boldsymbol{\phi}, B) \\ &= \sum_{i=1}^{n} \mathbb{E}_{q_{\boldsymbol{\phi}}(\mathbf{z}_{def}, \mathbf{z}_{cont}|\mathbf{x}_i)} \left[||\mathbf{x}_i - \tau(\mathbf{z}_{def}) \circ (\mathbf{z}_{cont} B^T)||^2 \right] \\ &+ \sum_{k, j \neq k} (\mathbf{b}_k^T \mathbf{b}_j)^2, \end{aligned} \tag{3}$$

where $\circ$ defines a transformation (affine in our experiments) operator decided by $\tau(\mathbf{z}_{def})$ and $\sum_{k,j\neq k}(\mathbf{b}_k^T \mathbf{b}_j)^2$ is regarded as the orthogonal loss. A normalization term $\sum_k (\mathbf{b}_k^T \mathbf{b}_k - 1)^2$ can be optionally added to force the basis to be unit vectors. We follow the spirit of the PCA optimization and a general formulation for learning PCA can be found in [7]. To keep the simplicity of the method we learn a fixed basis B and one can also adopt a probabilistic PCA model [49]. Thus, learning unsupervised Guided-VAE becomes:

$$\max_{\boldsymbol{\theta}, \boldsymbol{\phi}, B} \left\{ \sum_{i=1}^{n} ELBO(\boldsymbol{\theta}, \boldsymbol{\phi}; \mathbf{x}_i) - \mathcal{L}_{DPCA}(\boldsymbol{\phi}, B) \right\}. \tag{4}$$

The affine matrix described in our transformation follows implementation in [25]:

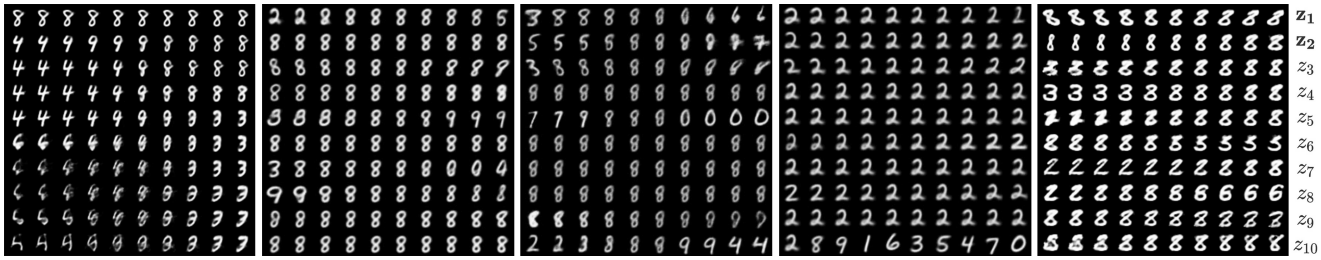
$$A_\theta = \begin{bmatrix} \theta_{11} & \theta_{12} & \theta_{13} \\ \theta_{21} & \theta_{22} & \theta_{23} \end{bmatrix} \tag{5}$$

The affine transformation includes translation, scale, rotation and shear operation. We use different latent variables to calculate different parameters in the affine matrix according to the operations we need.

3.3. Supervised Guided-VAE

For training data $\mathrm{X} = (\mathbf{x}_1, ..., \mathbf{x}_n)$, suppose there exists a total of T attributes with ground-truth labels. Let $\mathbf{z} = (z_t, \mathbf{z}_t^{rst})$ where z_t defines a scalar variable deciding the t-th attribute and $\mathbf{z}_t^{rst}$ represents remaining latent variables. Let $y_t(\mathbf{x}_i)$ be the ground-truth label for the t-th attribute of sample $\mathbf{x}_i$; $y_t(\mathbf{x}_i) \in \{-1, +1\}$. For each attribute, we use an adversarial excitation and inhibition method with term:

$$\begin{aligned} &\mathcal{L}_{Excitation}(\boldsymbol{\phi}, t) \\ &= \max_{w_t} \left\{ \sum_{i=1}^{n} \mathbb{E}_{q_{\boldsymbol{\phi}}(z_t|\mathbf{x}_i)}[\log p_{w_t}(y = y_t(\mathbf{x}_i)|z_t)] \right\}, \end{aligned} \tag{6}$$



(a)VAE (b) β-VAE (c) CCβ-VAE (d) JointVAE (e) Guided-VAE (Ours)

Figure 2. **Latent Variables Traversal on MNIST:** Comparison of traversal results from vanilla VAE [29], β-VAE [20], β-VAE with controlled capacity increase (CCβ-VAE), JointVAE [12] and our Guided-VAE on the MNIST dataset. z_1 and z_2 in Guided-VAE are controlled.

where w_t refers to classifier making a prediction for the t-th attribute using the latent variable z_t.

This is an excitation process since we want latent variable z_t to directly correspond to the attribute label.

Next is an inhibition term.

$$\mathcal{L}_{Inhibition}(\boldsymbol{\phi}, t) = \max_{C_t} \left\{ \sum_{i=1}^{n} \mathbb{E}_{q_{\boldsymbol{\phi}}(\mathbf{z}_t^{rst}|\mathbf{x}_i)}[\log p_{C_t}(y = y_t(\mathbf{x}_i)|\mathbf{z}_t^{rst})] \right\}, \tag{7}$$

where $C_t(\mathbf{z}_t^{rst})$ refers to classifier making a prediction for the t-th attribute using the remaining latent variables $\mathbf{z}_t^{rst}$. $\log p_{C_t}(y = y_t(\mathbf{x})|\mathbf{z}_t^{rst})$ is a cross-entropy term for minimizing the classification error in Eq. (7). This is an inhibition process since we want the remaining variables $\mathbf{z}_t^{rst}$ as independent as possible to the attribute label in Eq. (8) below.

$$\max_{\boldsymbol{\theta},\boldsymbol{\phi}} \left\{ \sum_{i=1}^{n} ELBO(\boldsymbol{\theta}, \boldsymbol{\phi}; \mathbf{x}_i) + \sum_{t=1}^{T} [\mathcal{L}_{Excitation}(\boldsymbol{\phi}, t) - \mathcal{L}_{Inhibition}(\boldsymbol{\phi}, t)] \right\}. \tag{8}$$

Notice in Eq. (8) the minus sign in front of the term $\mathcal{L}_{Inhibition}(\boldsymbol{\phi}, t)$ for maximization which is an adversarial term to make $\mathbf{z}_t^{rst}$ as uninformative to attribute t as possible, by pushing the best possible classifier C_t to be the least discriminative. The formulation of Eq. (8) bears certain similarity to that in domain-adversarial neural networks [15] in which the label classification is minimized with the domain classifier being adversarially maximized. Here, however, we respectively encourage and discourage different parts of the features to make the same type of classification.

4. Experiments

In this section, we first present qualitative and quantitative results demonstrating our proposed unsupervised Guided-VAE (Figure 1a) capable of disentangling latent embedding more favorably than previous disentangle methods [20, 12, 28] on MNIST dataset [35] and 2D shape dataset [42]. We also show that our learned latent representation improves classification performance in a representation learning setting. Next, we extend this idea to a supervised guidance approach in an adversarial excitation and inhibition fashion, where a discriminative objective for certain image properties is given (Figure 1b) on the CelebA dataset [39]. Further, we show that our method is architecture agnostic, applicable in a variety of scenarios such as image interpolation task on CIFAR 10 dataset [31] and a few-shot classification task on Omniglot dataset [33].

4.1. Unsupervised Guided-VAE

4.1.1 Qualitative evaluation

We present qualitative results on the MNIST dataset first by traversing latent variables received affine transformation guiding signal in Figure 2. Here, we applied the Guided-VAE with the bottleneck size of 10 (i.e. the latent variables $\mathbf{z} \in \mathbb{R}^{10}$). The first latent variable z_1 represents the rotation information, and the second latent variable z_2 represents the scaling information. The rest of the latent variables $\mathbf{z}_{3:10}$ represent the content information. Thus, we present the latent variables as $\mathbf{z} = (\mathbf{z}_{def}, \mathbf{z}_{cont}) = (\mathbf{z}_{1:2}, \mathbf{z}_{3:10})$.

We compare traversal results of all latent variables on MNIST dataset for vanilla VAE [29], β-VAE [20], JointVAE [12] and our Guided-VAE (β-VAE, JointVAE results are adopted from [12]). While β-VAE cannot generate meaningful disentangled representations for this dataset, even with controlled capacity increased, JointVAE can disentangle class type from continuous factors. Our Guided-VAE disentangles geometry properties rotation angle at z_1 and stroke thickness at z_2 from the rest content information $\mathbf{z}_{3:10}$.

To assess the disentangling ability of Guided-VAE against various baselines, we create a synthetic 2D shape dataset following [42, 20] as a common way to measure the disentanglement properties of unsupervised disentangling methods. The dataset consists 737,280 images of 2D

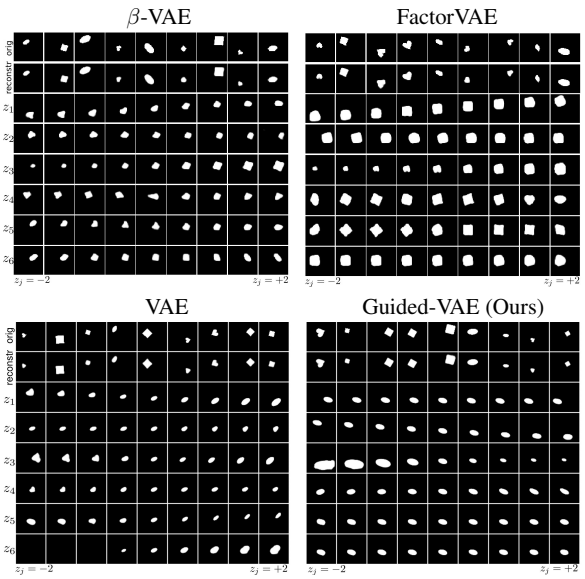


Figure 3. **Comparison of qualitative results on 2D shape.** First row: originals. Second row: reconstructions. Remaining rows: reconstructions of latent traversals across each latent dimension. In our results, z_1 represents the x-position information, z_2 represents the y-position information, z_3 represents the scale information and z_4 represents the rotation information.

shapes (heart, oval and square) generated from four ground truth independent latent factors: x-position information (32 values), y-position information (32 values), scale (6 values) and rotation (40 values). This gives us the ability to compare the disentangling performance of different methods with given ground truth factors. We present the latent space traversal results in Figure 3, where the results of β-VAE and FactorVAE are taken from [28]. Our Guided-VAE learns the four geometry factors with the first four latent variables where the latent variables $\mathbf{z} \in \mathbb{R}^6 = (\mathbf{z}_{def}, \mathbf{z}_{cont}) = (\mathbf{z}_{1:4}, \mathbf{z}_{5:6})$. We observe that although all models are able to capture basic geometry factors, the traversal results from Guided-VAE are more obvious with fewer factors changing except the target one.

4.1.2 Quantitative evaluation

We perform two quantitative experiments with strong baselines for disentanglement and representation learning in Table 1 and 2. We observe significant improvement over existing methods in terms of *disentanglement* measured by Z-Diff score [20], SAP score [32], Factor score [28] in Table 1, and representation *transferability* based on classification error in Table 2.

All models are trained in the same setting as the experiment shown in Figure 3, and are assessed by three disentangle metrics shown in Table 1. An improvement in the Z-Diff score and Factor score represents a lower variance of the inferred latent variable for fixed generative factors, whereas our increased SAP score corresponds with a tighter



Figure 4. **Comparison of Traversal Result learned on CelebA:** Column 1 shows traversed images from male to female. Column 2 shows traversed images from smiling to no-smiling. The first row is from [20] and we follow its figure generation procedure.

Model ($d_{\mathbf{z}} = 6$)	Z-Diff ↑	SAP ↑	Factor ↑
VAE [29]	78.2	0.1696	0.4074
β-VAE (β=2)[20]	98.1	0.1772	0.5786
FACTORVAE (γ=5) [28]	92.4	0.1770	0.6134
FACTORVAE (γ=35) [28]	98.4	0.2717	0.7100
β-TCVAE (α=1,β=5,γ=1) [8]	96.8	0.4287	0.6968
GUIDED-VAE (OURS)	**99.2**	0.4320	0.6660
GUIDED-β-TCVAE (OURS)	96.3	**0.4477**	**0.7294**

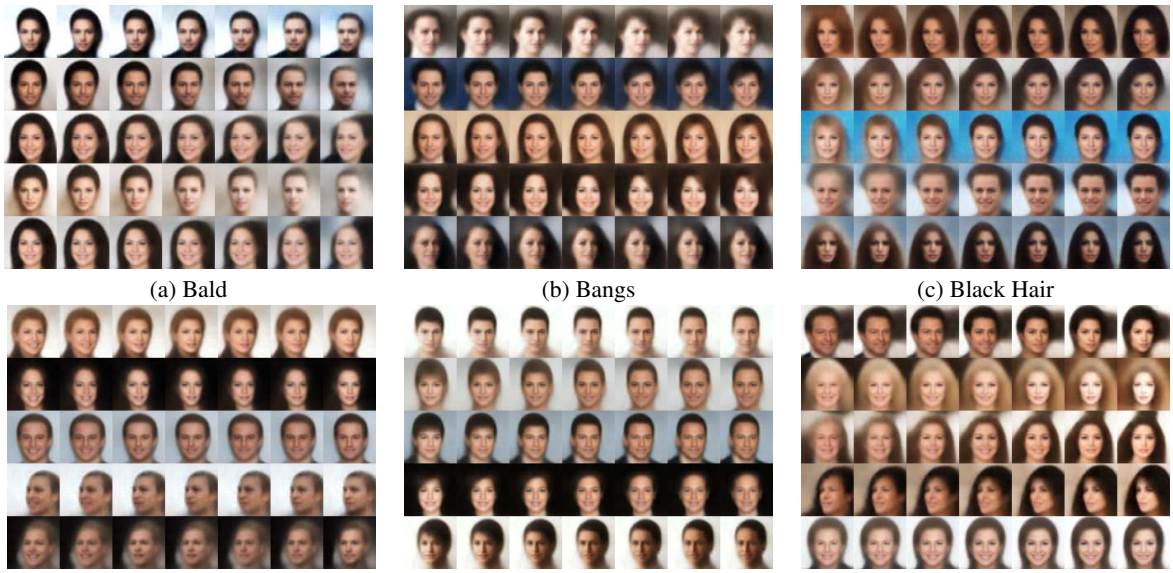
Table 1. **Disentanglement:** Z-Diff score, SAP score, and Factor score over unsupervised disentanglement methods on 2D Shapes dataset. [↑ means higher is better]

Model	$d_{\mathbf{z}} = 16 \downarrow$	$d_{\mathbf{z}} = 32 \downarrow$	$d_{\mathbf{z}} = 64 \downarrow$
VAE [29]	2.92%±0.12	3.05%±0.42	2.98%±0.14
β-VAE(β=2)[20]	4.69%±0.18	5.26%±0.22	5.40%±0.33
FACTORVAE(γ=5) [28]	6.07%±0.05	6.18%±0.20	6.35%±0.48
β-TCVAE (α=1,β=5,γ=1) [8]	1.62%±0.07	1.24%±0.05	1.32%±0.09
GUIDED-VAE (OURS)	1.85%±0.08	1.60%±0.08	1.49%±0.06
GUIDED-β-TCVAE (OURS)	**1.47%±0.12**	**1.10%±0.03**	**1.31%±0.06**

Table 2. **Representation Learning:** Classification error over unsupervised disentanglement methods on MNIST. [↓ means lower is better]† The 95 % confidence intervals from 5 trials are reported.

coupling between a single latent dimension and a generative factor. Compare to previous methods, our method is orthogonal (due to using a side objective) to most existing approaches. β-TCVAE [8] improves β-VAE [20] based on weighted mini-batches to stochastic training. Our Guided-β-TCVAE further improves the results in all three disentangle metrics.

We further study representation transferability by performing classification tasks on the latent embedding of different generative models. Specifically, for each data point $(\mathbf{x}, y)$, we use the pre-trained generative models to obtain the value of latent variable $\mathbf{z}$ given input image $\mathbf{x}$. Here $\mathbf{z}$ is a $d_{\mathbf{z}}$-dim vector. We then train a linear classifier $f(\cdot)$ on the embedding-label pairs $\{(\mathbf{z}, y)\}$ to predict the class of digits. For the Guided-VAE, we disentangle the latent variables $\mathbf{z}$ into deformation variables $\mathbf{z}_{def}$ and content variables $\mathbf{z}_{cont}$



(a) Bald (b) Bangs (c) Black Hair

(d) Mouth Slightly Open (e) Receding Hairlines (f) Young

Figure 5. **Latent factors learned by Guided-VAE on CelebA:** Each image shows the traversal results of Guided-VAE on a single latent variable which is controlled by the lightweight decoder using the corresponding labels as signal.

with same dimensions (i.e. $d_{\mathbf{z}_{def}} = d_{\mathbf{z}_{cont}}$). We compare the classification errors of different models with multiple choices of dimensions of the latent variables in Table 2. In general, VAE [29], β-VAE [20], and FactorVAE [28] do not benefit from the increase of the latent dimensions, and β-TCVAE [8] shows evidence that its discovered representation is more useful for classification task than existing methods. Our Guide-VAE achieves competitive results compare to β-TCVAE, and our Guided-β-TCVAE can further reduce the classification error to 1.1% when $d_{\mathbf{z}} = 32$, which is 1.95% lower than the baseline VAE.

Moreover, we study the effectiveness of $\mathbf{z}_{def}$ and $\mathbf{z}_{cont}$ in Guided-VAE separately to reveal the different properties of the latent subspace. We follow the same classification task procedures described above but use different subsets of latent variables as input features for the classifier $f(\cdot)$. Specifically, we compare results based on the deformation variables $\mathbf{z}_{def}$, the content variables $\mathbf{z}_{cont}$, and the whole latent variables $\mathbf{z}$ as the input feature vector. To conduct a fair comparison, we still keep the same dimensions for the deformation variables $\mathbf{z}_{def}$ and the content variables $\mathbf{z}_{cont}$. Table 3 shows that the classification errors on $\mathbf{z}_{cont}$ are significantly lower than the ones on $\mathbf{z}_{def}$, which indicates the success of disentanglement as the content variables should determine the class of digits. In contrast, the deformation variables should be invariant to the class. Besides, when the dimensions of latent variables $\mathbf{z}$ are higher, the classification errors on $\mathbf{z}_{def}$ increase while the ones on $\mathbf{z}_{cont}$ decrease, indicating a better disentanglement between deformation and content with increased latent dimensions.

Model	$d_{\mathbf{z}_{def}}$	$d_{\mathbf{z}_{cont}}$	$d_{\mathbf{z}}$	$\mathbf{z}_{def}$ *Error* ↑	$\mathbf{z}_{cont}$ *Error* ↓	$\mathbf{z}$ *Error* ↓
GUIDED-VAE	8	8	16	27.1%	3.69%	2.17%
	16	16	32	42.07%	1.79%	1.51%
	32	32	64	62.94%	1.55%	1.42%

Table 3. **Classification on MNIST using different latent variables as features:** Classification error over Guided-VAE with different dimensions of latent variables [↑ means higher is better, ↓ means lower is better]

4.2. Supervised Guided-VAE

4.2.1 Qualitative evaluation

We first present qualitative results on the CelebA dataset [39] by traversing latent variables of attributes shown in Figure 4 and Figure 5. In Figure 4, we compare the traversal results of Guided-VAE with β-VAE for two labeled attributes (gender, smile) in the CelebA dataset. The bottleneck size is set to 16 ($d_{\mathbf{z}} = 16$). We use the first two latent variables z_1, z_2 to represent the attribute information, and the rest $\mathbf{z}_{3:16}$ to represent the content information. During evaluation, we choose $z_t \in \{z_1, z_2\}$ while keeping the remaining latent variables $\mathbf{z}_t^{rst}$ fixed. Then we obtain a set of images through traversing t-th attribute (e.g., smiling to non-smiling) and compare them over β-VAE. In Figure 5, we present traversing results on another six attributes.

β-VAE performs decently for the controlled attribute change, but the individual $\mathbf{z}$ in β-VAE is not fully entangled or disentangled with the attribute. We observe the traversed images contain several attribute changes at the same time. Different from our Guided-VAE, β-VAE cannot specify which latent variables to encode specific attribute information. Guided-VAE, however, is designed to allow defined

latent variables to encode any specific attributes. Guided-VAE outperforms β-VAE by only traversing the intended factors (smile, gender) without changing other factors (hair color, baldness).

4.2.2 Quantitative evaluation

We attempt to interpret whether the disentangled attribute variables can control the generated images from the supervised Guided-VAE. We pre-train an external binary classifier for t-th attribute on the CelebA training set and then use this classifier to test the generated images from Guided-VAE. Each test includes $10,000$ generated images randomly sampled on all latent variables except for the particular latent variable z_t we decide to control. As Figure 6 shows, we can draw the confidence-z curves of the t-th attribute where $z = z_t \in [-3.0, 3.0]$ with 0.1 as the stride length. For the gender and the smile attributes, it can be seen that the corresponding z_t is able to enable ($z_t < -1$) and disable ($z_t > 1$) the attribute of the generated image, which shows the controlling ability of the t-th attribute by tuning the corresponding latent variable z_t.

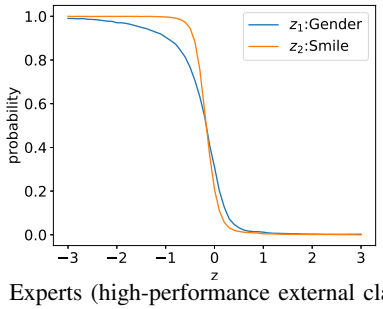


Figure 6. Experts (high-performance external classifiers for attribute classification) prediction for being negatives on the generated images. We traverse z_1 (gender) and z_2 (smile) separately to generate images for the classification test.

4.2.3 Image Interpolation

We further show the disentanglement properties of using supervised Guided-VAE on the CIFAR10 dataset. ALI-VAE borrows the architecture that is defined in [11], where we treat G_z as the encoder and G_x as the decoder. This enables us to optimize an additional reconstruction loss. Based on ALI-VAE, we implement Guided-ALI-VAE (Ours), which adds supervised guidance through excitation and inhibition shown in Figure 1. ALI-VAE and AC-GAN [3] serve as a baseline for this experiment.

To analyze the disentanglement of the latent space, we train each of these models on a subset of the CIFAR10 dataset [31] (Automobile, Truck, Horses) where the class label corresponds to the attribute to be controlled. We use a bottleneck size of 10 for each of these models. We follow the training procedure mentioned in [3] for training the AC-GAN model and the optimization parameters reported in [11] for ALI-VAE and our model. For our Guided-ALI-VAE model, we add supervision through inhibition and excitation on $z_{1:3}$.

Model	Automobile-Horse ↓	Truck-Automobile ↓
AC-GAN [3]	88.27	81.13
ALI-VAE †	91.96	78.92
GUIDED-ALI-VAE (OURS)	**85.43**	**72.31**

Table 4. **Image Interpolation:** FID score measured for a subset of CIFAR10 [31] with two classes each. [↓ means lower is better] † ALI-VAE is a modification of the architecture defined in [11]

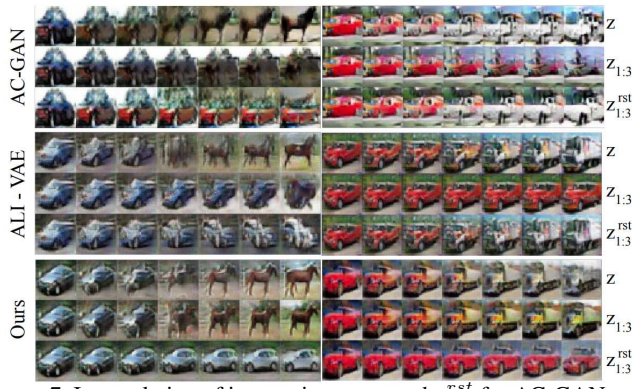


Figure 7. Interpolation of images in z, $z_{1:3}$ and $z^{rst}_{1:3}$ for AC-GAN, ALI-VAE and Guided-ALI-VAE (Ours).

To visualize the disentanglement in our model, we interpolate the corresponding z, z_t and z_t^{rst} of two images sampled from different classes. The interpolation here is computed as a uniformly spaced linear combination of the corresponding vectors. The results in Figure 7 qualitatively show that our model is successfully able to capture complementary features in $z_{1:3}$ and $z^{rst}_{1:3}$. Interpolation in $z_{1:3}$ corresponds to changing the object type. Whereas, the interpolation in $z^{rst}_{1:3}$ corresponds to complementary features such as color and pose of the object.

The right column in Figure 7 shows that our model can traverse in $z_{1:3}$ to change the object in the image from an automobile to a truck. Whereas a traversal in $z^{rst}_{1:3}$ changes other features such as background and the orientation of the automobile. We replicate the procedure on ALI-VAE and AC-GAN and show that these models are not able to consistently traverse in $z_{1:3}$ and $z^{rst}_{1:3}$ in a similar manner. Our model also produces interpolated images in higher quality as shown through the FID scores [19] in Table 4.

4.3. Few-Shot Learning

Previously, we have shown that Guided-VAE can perform images synthesis and interpolation and form better representation for the classification task. Similarly, we can apply our supervised method to VAE-like models in the few-shot classification. Specifically, we apply our adversarial excitation and inhibition formulation to the Neural Statistician [13] by adding a supervised guidance network after the statistic network. The supervised guidance sig-

nal is the label of each input. We also apply the Mixup method [54] in the supervised guidance network. However, we could not reproduce exact reported results in the Neural Statistician, which is also indicated in [30]. For comparison, we mainly consider results from Matching Nets [51] and Bruno [30] shown in Table 5. Yet it cannot outperform Matching Nets, our proposed Guided Neural Statistician reaches comparable performance as Bruno (discriminative), where a discriminative objective is fine-tuned to maximize the likelihood of correct labels.

Model	5-way		20-way	
Omniglot	**1-shot**	**5-shot**	**1-shot**	**5-shot**
PIXELS [51]	41.7%	63.2%	26.7%	42.6%
BASELINE CLASSIFIER [51]	80.0%	95.0%	69.5%	89.1%
MATCHING NETS [51]	98.1%	98.9%	93.8%	98.5%
BRUNO [30]	86.3%	95.6%	69.2%	87.7%
BRUNO (DISCRIMINATIVE) [30]	97.1%	99.4%	91.3%	97.8%
BASELINE	97.7%	99.4%	91.4%	96.4%
OURS (DISCRIMINATIVE)	97.8%	99.4%	92.1%	96.6%

Table 5. **Few-shot classification:** Classification accuracy for a few-shot learning task on the Omniglot dataset.

5. Ablation Study

5.1. Deformable PCA

In Figure 8, we visualize the sampling results from PCA and Dec_{sub}. By applying a deformation layer into the PCA-like layer, we show deformable PCA has a more crispy sampling result than vanilla PCA.



Figure 8. (Top) Sampling Result Obtained from PCA (Bottom) Sampling Result obtained from learned deformable PCA (Ours)

5.2. Guided Autoencoder

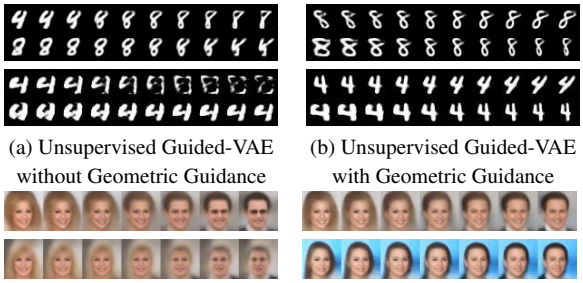
To further validate our concept of "guidance", we introduce our lightweight decoder to the standard autoencoder (AE) framework. We conduct MNIST classification tasks using the same setting in Figure 2. As Table 6 shows, our lightweight decoder improves the representation learned in autoencoder framework. Yet a VAE-like structure is indeed not needed if the purpose is just reconstruction and representation learning. However, VAE is of great importance in building generative models. The modeling of the latent space of $\mathbf{z}$ with e.g., Gaussian distributions is again important if a probabilistic model is needed to perform novel data synthesis (e.g., the images shown in Figure 4 and Figure 5).

Model	$d_{\mathbf{z}} = 16 \downarrow$	$d_{\mathbf{z}} = 32 \downarrow$	$d_{\mathbf{z}} = 64 \downarrow$
AUTO-ENCODER (AE)	**1.37**%±0.05	1.06%±0.04	1.34%±0.04
GUIDED-AE (OURS)	1.46%±0.06	**1.00**%±0.06	**1.10**%±0.08

Table 6. Classification error over AE and Guided-AE on MNIST.

5.3. Geometric Transformations

We conduct an experiment by excluding the geometry-guided part from the unsupervised Guided-VAE. In this way, the lightweight decoder is just a PCA-like decoder but not a deformable PCA. The setting of this experiment is exactly the same as described in Figure 2. The bottleneck size of our model is set to 10 of which the first two latent variables z_1, z_2 represent the rotation and scaling information separately. As a comparison, we drop off the geometric guidance so that all 10 latent variables are controlled by the PCA-like light decoder. As shown in Figure 9 (a) (b), it can be easily seen that geometry information is hardly encoded into the first two latent variables without a geometry-guided part.



(a) Unsupervised Guided-VAE without Geometric Guidance

(b) Unsupervised Guided-VAE with Geometric Guidance

(c) Supervised Guided-VAE without Inhibition

(d) Supervised Guided-VAE with Inhibition

Figure 9. Ablation study on Unsupervised Guided-VAE and Supervised Guided-VAE

5.4. Adversarial Excitation and Inhibition

We study the effectiveness of adversarial inhibition using the exact same setting described in the supervised Guided-VAE part. As shown in Figure 9 (c) and (d), Guided-VAE without inhibition changes the smiling and sunglasses while traversing the latent variable controlling the gender information. This problem is alleviated by introducing the excitation-inhibition mechanism into Guided-VAE.

6. Conclusion

In this paper, we have presented a new representation learning method, guided variational autoencoder (Guided-VAE), for disentanglement learning. Both unsupervised and supervised versions of Guided-VAE utilize lightweight guidance to the latent variables to achieve better controllability and transparency. Improvements in disentanglement, image traversal, and meta-learning over the competing methods are observed. Guided-VAE maintains the backbone of VAE and it can be applied to other generative modeling applications.

Acknowledgment. This work is funded by NSF IIS-1618477, NSF IIS-1717431, and Qualcomm Inc. ZD is supported by the Tsinghua Academic Fund for Undergraduate Overseas Studies. We thank Kwonjoon Lee, Justin Lazarow, and Jilei Hou for valuable feedbacks.

References

[1] Alessandro Achille and Stefano Soatto. Emergence of invariance and disentanglement in deep representations. *The Journal of Machine Learning Research*, 19(1):1947–1980, 2018. 2

[2] Martin Arjovsky, Soumith Chintala, and Léon Bottou. Wasserstein generative adversarial networks. In *ICML*, 2017. 2

[3] Jonathon Shlens Augustus Odena, Christopher Olah. Conditional image synthesis with auxiliary classifier gans. In *ICML*, 2017. 7

[4] Mikhail Belkin and Partha Niyogi. Laplacian eigenmaps for dimensionality reduction and data representation. *Neural computation*, 15(6):1373–1396, 2003. 1

[5] Piotr Bojanowski, Armand Joulin, David Lopez-Paz, and Arthur Szlam. Optimizing the latent space of generative networks. In *ICML*, 2018. 2

[6] Hervé Bourlard and Yves Kamp. Auto-association by multilayer perceptrons and singular value decomposition. *Biological cybernetics*, 59(4-5):291–294, 1988. 1

[7] Emmanuel J Candès, Xiaodong Li, Yi Ma, and John Wright. Robust principal component analysis? *Journal of the ACM (JACM)*, 58(3):11, 2011. 2, 3

[8] Tian Qi Chen, Xuechen Li, Roger B Grosse, and David K Duvenaud. Isolating sources of disentanglement in variational autoencoders. In *Advances in Neural Information Processing Systems*, 2018. 2, 5, 6

[9] Elliot Creager, David Madras, Joern-Henrik Jacobsen, Marissa Weis, Kevin Swersky, Toniann Pitassi, and Richard Zemel. Flexibly fair representation learning by disentanglement. In *ICML*, 2019. 2

[10] Bin Dai, Yu Wang, John Aston, Gang Hua, and David Wipf. Connections with robust pca and the role of emergent sparsity in variational autoencoder models. *The Journal of Machine Learning Research*, 19(1):1573–1614, 2018. 2

[11] Vincent Dumoulin, Ishmael Belghazi, Ben Poole, Olivier Mastropietro, Alex Lamb, Martin Arjovsky, and Aaron Courville. Adversarially learned inference. In *ICLR*, 2017. 7

[12] Emilien Dupont. Learning disentangled joint continuous and discrete representations. In *Advances in Neural Information Processing Systems*, 2018. 4

[13] Harrison Edwards and Amos Storkey. Towards a neural statistician. In *ICLR*, 2017. 7

[14] Jesse Engel, Matthew Hoffman, and Adam Roberts. Latent constraints: Learning to generate conditionally from unconditional generative models. In *ICLR*, 2018. 2

[15] Yaroslav Ganin, Evgeniya Ustinova, Hana Ajakan, Pascal Germain, Hugo Larochelle, François Laviolette, Mario Marchand, and Victor Lempitsky. Domain-adversarial training of neural networks. *The Journal of Machine Learning Research*, 17(1):2096–2030, 2016. 2, 4

[16] Abel Gonzalez-Garcia, Joost van de Weijer, and Yoshua Bengio. Image-to-image translation for cross-domain disentanglement. In *Advances in Neural Information Processing Systems*, 2018. 2

[17] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep learning*, volume 1. MIT Press, 2016. 1

[18] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Advances in neural information processing systems*, 2014. 1, 2

[19] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. In *Advances in neural information processing systems*, 2017. 7

[20] Irina Higgins, Loic Matthey, Arka Pal, Christopher Burgess, Xavier Glorot, Matthew Botvinick, Shakir Mohamed, and Alexander Lerchner. beta-vae: Learning basic visual concepts with a constrained variational framework. In *ICLR*, 2017. 2, 4, 5, 6

[21] Geoffrey E Hinton and Richard S Zemel. Autoencoders, minimum description length and helmholtz free energy. In *Advances in neural information processing systems*, 1994. 1

[22] Harold Hotelling. Analysis of a complex of statistical variables into principal components. *Journal of educational psycholog*, 24, 1933. 1, 2

[23] Qiyang Hu, Attila Szabó, Tiziano Portenier, Paolo Favaro, and Matthias Zwicker. Disentangling factors of variation by mixing them. In *CVPR*, 2018. 2

[24] Maximilian Ilse, Jakub M Tomczak, Christos Louizos, and Max Welling. Diva: Domain invariant variational autoencoders. In *ICLR Worshop Track*, 2019. 2

[25] Max Jaderberg, Karen Simonyan, Andrew Zisserman, et al. Spatial transformer networks. In *Advances in neural information processing systems*, 2015. 3

[26] Ananya Harsh Jha, Saket Anand, Maneesh Singh, and VSR Veeravasarapu. Disentangling factors of variation with cycle-consistent variational auto-encoders. In *ECCV*, 2018. 2

[27] Ian Jolliffe. Principal component analysis. *Springer Berlin Heidelberg*, 2011. 1, 2

[28] Hyunjik Kim and Andriy Mnih. Disentangling by factorising. In *ICML*, 2018. 2, 4, 5, 6

[29] Diederik P Kingma and Max Welling. Auto-encoding variational bayes. In *ICLR*, 2014. 1, 2, 3, 4, 5, 6

[30] Iryna Korshunova, Jonas Degrave, Ferenc Huszár, Yarin Gal, Arthur Gretton, and Joni Dambre. Bruno: A deep recurrent model for exchangeable data. In *Advances in Neural Information Processing Systems*, 2018. 8

[31] Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. Technical report, Citeseer, 2009. 4, 7

[32] Abhishek Kumar, Prasanna Sattigeri, and Avinash Balakrishnan. Variational inference of disentangled latent concepts from unlabeled observations. In *ICLR*, 2018. 5

[33] Brenden M Lake, Ruslan Salakhutdinov, and Joshua B Tenenbaum. Human-level concept learning through probabilistic program induction. *Science*, 350(6266):1332–1338, 2015. 4

[34] Yann LeCun. *Modeles connexionnistes de lapprentissage*. PhD thesis, PhD thesis, These de Doctorat, Universite Paris 6, 1987. 1

[35] Yann LeCun. The mnist database of handwritten digits. *http://yann. lecun. com/exdb/mnist/*, 1998. 4

[36] Chen-Yu Lee, Saining Xie, Patrick Gallagher, Zhengyou Zhang, and Zhuowen Tu. Deeply-supervised nets. In *Artificial intelligence and statistics*, pages 562–570, 2015. 2

[37] Jianxin Lin, Zhibo Chen, Yingce Xia, Sen Liu, Tao Qin, and Jiebo Luo. Exploring explicit domain supervision for latent space disentanglement in unpaired image-to-image translation. *IEEE transactions on pattern analysis and machine intelligence*, 2019. 2

[38] Yen-Cheng Liu, Yu-Ying Yeh, Tzu-Chien Fu, Sheng-De Wang, Wei-Chen Chiu, and Yu-Chiang Frank Wang. Detach and adapt: Learning cross-domain disentangled deep representation. In *CVPR*, 2018. 2

[39] Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Deep learning face attributes in the wild. In *ICCV*, 2015. 4, 6

[40] Alireza Makhzani, Jonathon Shlens, Navdeep Jaitly, Ian Goodfellow, and Brendan Frey. Adversarial autoencoders. In *ICLR Workshop Track*, 2016. 2

[41] Michael F Mathieu, Junbo Jake Zhao, Junbo Zhao, Aditya Ramesh, Pablo Sprechmann, and Yann LeCun. Disentangling factors of variation in deep representation using adversarial training. In *Advances in Neural Information Processing Systems*, 2016. 2

[42] Loic Matthey, Irina Higgins, Demis Hassabis, and Alexander Lerchner. dsprites: Disentanglement testing sprites dataset. https://github.com/deepmind/dsprites-dataset/, 2017. 4

[43] Brendan K Murphy and Kenneth D Miller. Multiplicative gain changes are induced by excitation or inhibition alone. *Journal of Neuroscience*, 23(31):10040–10051, 2003. 2

[44] Xi Peng, Xiang Yu, Kihyuk Sohn, Dimitris N Metaxas, and Manmohan Chandraker. Reconstruction-based disentanglement for pose-invariant face recognition. In *ICCV*, 2017. 2

[45] Yigang Peng, Arvind Ganesh, John Wright, Wenli Xu, and Yi Ma. Rasl: Robust alignment by sparse and low-rank decomposition for linearly correlated images. *IEEE transactions on pattern analysis and machine intelligence*, 34(11):2233–2246, 2012. 2

[46] Christopher Poultney, Sumit Chopra, Yann LeCun, et al. Efficient learning of sparse representations with an energy-based model. In *Advances in neural information processing systems*, 2007. 1

[47] Jiaming Song, Pratyusha Kalluri, Aditya Grover, Shengjia Zhao, and Stefano Ermon. Learning controllable fair representations. In *The 22nd International Conference on Artificial Intelligence and Statistics*, 2019. 2

[48] Attila Szabó, Qiyang Hu, Tiziano Portenier, Matthias Zwicker, and Paolo Favaro. Challenges in disentangling independent factors of variation. In *ICLR Workshop Track*, 2018. 2

[49] Michael E Tipping and Christopher M Bishop. Probabilistic principal component analysis. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 61(3):611–622, 1999. 3

[50] Pascal Vincent, Hugo Larochelle, Isabelle Lajoie, Yoshua Bengio, and Pierre-Antoine Manzagol. Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion. *Journal of machine learning research*, 11(Dec):3371–3408, 2010. 1

[51] Oriol Vinyals, Charles Blundell, Timothy Lillicrap, Daan Wierstra, et al. Matching networks for one shot learning. In *Advances in neural information processing systems*, 2016. 8

[52] Michael P Wellman and Max Henrion. Explaining'explaining away'. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 15(3):287–292, 1993. 2

[53] Ofer Yizhar, Lief E Fenno, Matthias Prigge, Franziska Schneider, Thomas J Davidson, Daniel J O'shea, Vikaas S Sohal, Inbal Goshen, Joel Finkelstein, Jeanne T Paz, et al. Neocortical excitation/inhibition balance in information processing and social dysfunction. *Nature*, 477(7363):171, 2011. 2

[54] Hongyi Zhang, Moustapha Cisse, Yann N Dauphin, and David Lopez-Paz. mixup: Beyond empirical risk minimization. In *ICLR*, 2018. 8