

L-VAE: Variational Auto-Encoder with Learnable Beta for Disentangled Representation

Hazal Mogultay Ozcan^{1*}, Sinan Kalkan¹ and Fatos T. Yarman-Vural¹

¹Computer Engineering, Middle East Technical University, Ankara, 06800, Turkey.

*Corresponding author(s). E-mail(s): mogultay@metu.edu.tr;
Contributing authors: skalkan@metu.edu.tr; yarman@metu.edu.tr;

Abstract

In this paper, we propose a novel model called Learnable VAE (L-VAE), which learns a disentangled representation together with the hyperparameters of the cost function. L-VAE can be considered as an extension of β -VAE, wherein the hyperparameter, β , is empirically adjusted. L-VAE mitigates the limitations of β -VAE by learning the relative weights of the terms in the loss function to control the dynamic trade-off between disentanglement and reconstruction losses. In the proposed model, the weight of the loss terms and the parameters of the model architecture are learned concurrently. An additional regularization term is added to the loss function to prevent bias towards either reconstruction or disentanglement losses. Experimental analyses show that the proposed L-VAE finds an effective balance between reconstruction fidelity and disentangling the latent dimensions. Comparisons of the proposed L-VAE against β -VAE, VAE, ControlVAE, DynamicVAE, and σ -VAE on datasets, such as dSprites, MPI3D-complex, Falcor3D, and Isaac3D reveals that L-VAE consistently provides the best or the second best performances measured by a set of disentanglement metrics. Moreover, qualitative experiments on CelebA dataset, confirm the success of the L-VAE model for disentangling the facial attributes.

Keywords: Variational Auto-Encoders, β -VAE, Learnable β , Disentangled Representation Learning.

1 Introduction

Deep learning architectures inherently possess limitations concerning their capacity for generalization, explainability, and interpretability [2, 14, 33, 43]. A promising approach to reduce these limitations is to identify the independent factors of the data generation process that can represent the implicit properties of the data, such as rotation, translation, shape, or shadow [2, 14]. This approach, coined as disentanglement, facilitates generalization, explainability, and interpretability in terms of the identified essential properties [8, 15, 28].

Variational Auto-Encoders (VAEs) [22] are one of the pioneering models to learn disentangled representations. Rather than learning a representation with entangled properties, zero mean unit variance Gaussian priors are enforced on each dimension of the latent properties. For this purpose, the Kullback-Leibler divergence between the priors and the learnt distribution (\mathcal{D}_{KL}) is used alongside a reconstruction loss (\mathcal{L}_R) to enforce disentanglement in the learnt representation,

$$\mathcal{L} = \mathcal{L}_R + \mathcal{D}_{KL}, \quad (1)$$

which joins the two incompatible terms with different orders of magnitude and difficulty, which may result in an imbalance in the optimization of the overall loss function. In order to reduce this imbalance problem, the KL divergence term is weighted by a hyperparameter β in β -VAE [3, 14], which accentuates the importance of the KL divergence with the hope of learning more disentangled representations compared to VAEs.

Despite its promises, β -VAE exhibits certain shortcomings for learning disentangled representations. Firstly, the performances of β -VAE's are not consistently robust against different values of β [27]. Furthermore, finding an optimal β value via empirical methods requires computationally expensive and exhaustive search methods [35]. Secondly, even when a near-optimal β parameter is empirically caught, minimizing the loss function and the KL divergence with a fixed β parameter does not necessarily result in a minimum reconstruction loss with a maximum degree of disentanglement. In most cases, it is observed that β -VAE increases the reconstruction loss for the sake of better disentanglement [5], which may result in poor representation of the data. In other words, a fixed β parameter makes it difficult to discover the disentangled factors of variations providing minimum reconstruction loss. These challenges indicate the need for better approaches to improve the disentanglement capabilities of β -VAE.

In this study, we propose a model called Learnable Variational Auto-Encoder (L-VAE), which learns the parameter β that minimizes the reconstruction loss (\mathcal{L}_R) and maximizes disentanglement (through \mathcal{D}_{KL}). In this sense, L-VAE can be considered an extended version of β -VAE. L-VAE, with the learnable β , learns the trade-off between the reconstruction loss (\mathcal{L}_R) and the amount of disentanglement through the KL divergence term (\mathcal{D}_{KL}). This eliminates the burden of optimizing the hyperparameter β .

Our main contributions can be summarized as follows:

- We conduct an analysis on β -VAE and highlight several critical observations regarding the β hyperparameter. For example, we observe that β -VAE is highly sensitive to the β hyperparameter and that $\beta < 1$ can surprisingly provide better disentanglement.

- Motivated by the challenges and the issues associated with the β hyperparameter, we propose an auto-differentiation compatible method for learning the β hyperparameter in a hassle-free manner.
- We perform comprehensive evaluations on five different datasets (dSprites, MPI3D-complex, Falcor3D, Isaac3D, and CelebA) and compare L-VAE with VAE and β -VAE as well as three state-of-the-art methods (ControlVAE [37, 38], DynamicVAE [39], and σ -VAE [35]) which also aim to dynamically tune the β parameter in β -VAE. We show that L-VAE consistently provides the best or second-best performance in terms of widely used disentanglement measures without empirically tuning the β hyperparameter.

2 Disentanglement in Machine Learning

Over the last decade, disentangled representation learning received significant attention in the machine learning community. It has been utilized in a wide range of problems, including e.g., facial image analysis [25], image dehazing [9], face hallucination [10], video frame generation [7, 16, 44], identity learning [25, 32], image-to-image translation [24], and face forgery detection [12]. In addition to Variational Auto-Encoder-based approaches (to be covered in the next section), researchers have proposed several approaches, such as the ones based on Generative Adversarial Networks [6, 17, 18, 30], and causality [36, 42]. Disentanglement is generally studied in unsupervised learning problems, without using any labels for the factors of variations. However, it is possible to address disentanglement in a supervised framework (see, e.g., [41]).

2.1 Variational Auto-Encoders and its Variations

Due to its relative simplicity and disentangling capability, Variational Auto-Encoders [22] are widely used for disentangled representation learning. It has been shown that VAE models disentangle the representation space to a certain degree, making them an appealing approach for improving the generalization capacity and interpretability

of the model in an unsupervised setting [14]. β -VAE encourages disentanglement by introducing the weight β on the KL term. This hyperparameter emphasizes the importance of disentanglement relative to the reconstruction loss. However, this approach introduces a new hyperparameter to be empirically optimized in a large search space. Furthermore, optimization of the overall loss function may result in an increased reconstruction loss.

Many studies explored the generalization properties of VAEs and proposed extensions. For example, in order to regain the model’s reconstruction abilities, Burgess et al. proposed a control mechanism on the capacity of the model [3]. They argue that, by gradually increasing the capacity of the model from zero, they can increase the disentanglement ability, while conserving the reconstruction loss.

Kim et al. [20] also argued that minimizing the loss function of the β -VAE decreases the reconstruction quality. In order to balance the reconstruction loss and KL divergence, they proposed a new model called Factor-VAE, in which they introduced a discriminator to the same architecture to elaborate the trade-off between disentanglement and reconstruction. The overall loss function increases the independence among latent dimensions and does not affect the mutual information.

Chen et al. [5] investigated the source of success in β -VAE and decomposed the ELBO loss term into three parts, namely, index code mutual information, total correlation, and dimension-wise KL divergence. These terms correspond to the mutual information between data and latent code, dependence among variables, and Kullback-Leibler divergence of the terms from the priors, respectively. They argue that the success of β -VAE in terms of disentanglement stems from the total correlation. Then, they proposed the model β -TCVAE, an extension to β -VAE, where they weighted each of the three terms. The optimized function is the same as [20]; however, they estimated the TC term with a different method.

A good resource that compares different VAE models is the study by Locatello et al. [27]. They analyzed the performances of the aforementioned explicit models in a compelling scenario. They executed 12K models and provided comparative results on six different metrics. Their findings emphasize that hyperparameter selection is crucial in terms of disentanglement.

2.2 Automatically Tuning β in β -VAE

Several studies explored automatic hyperparameter tuning in VAE models, specifically balancing the reconstruction loss and KL divergence terms in the loss function [37–39]. Shao et al. proposed ControlVAE algorithm for this specific purpose [37, 38]. They designed a Proportional–Integral Controller (PI Controller), a variation of PID controller [1], to automatically adjust the β of β -VAE algorithm. They first set a desired KL value, and at each iteration, they compute the error between the desired KL and its current value and apply a correction to reduce the error. The correction is conducted via the output of the PI controller, i.e., the β parameter. They further improved this model and proposed DynamicVAE [39]. DynamicVAE uses a similar PI Controller, however, it reduces β iteratively from a large value rather than increasing it, providing a smoother change on the KL term. Although ControlVAE and DynamicVAE can efficiently learn β , they introduce several new parameters to optimize, such as the desired value of the KL divergence, the initial value of β , and the constants of the PID controller. Moreover, Rybkin et al. proposed σ -VAE, which learns the variance of the decoder with network parameters [35]. Although they did not analyze σ -VAE as a disentanglement model and focused on the reconstruction of the model, their work is similar to ours since they learn the relative weight of the reconstruction term.

2.3 Comparative Summary

Most of the aforementioned studies rely on tuning the β hyperparameter empirically through extensive experiments, with the exception of the recent ControlVAE [37, 38], DynamicVAE [39] and σ -VAE [35] models. Although they show promising results, ControlVAE and DynamicVAE introduce additional hyperparameters, such as, the desired divergence value, or parameters of the PID control algorithm. On the other hand, σ -VAE requires soft clipping on the learned weights and implicitly sets a hyperparameter for the learned weight. These additional hyperparameters further complicate a learning disentangled representation. In contrast, we propose a relatively simple and hassle-free

solution that does not require tuning the hyperparameters of the cost function.

In the following sections, we start by defining the concept of disentanglement. Subsequently, we provide a brief description and analysis of our observations pertaining to β -VAE. Next, we introduce our Learnable Variational Autoencoder (L-VAE) model. Finally, we report our experiments for analyzing the L-VAE model, comparing it with the state of the art disentangled learning models.

3 Definition(s) of Disentanglement

There are various definitions for disentanglement in representation learning [2, 3, 5, 14, 15, 27]. A mathematically tractable approach to define disentanglement is based on the assumption that the data consists of a set of hidden properties shared across the categories. These hidden properties are one of the major causes of the variations among the objects of the same category. Then, disentanglement is defined as the separation of these properties embedded in the observations while learning a representation function. In this study, we adopt the latter definition and concentrate on the computational aspect of disentanglement. In this definition, observations are assumed to be generated by a set of independent Gaussian sources. The goal of disentanglement is to learn a representation, where each Gaussian distribution, generated by a source is modeled compactly in a separate subspace of the learnt representation. This approach allows an interpretation of the distribution of the representation in terms of the distribution of the hidden properties.

Formally, consider a dataset $\mathcal{D} = \{\mathcal{X}, \mathcal{W}, \mathcal{Y}\}$, where $\mathbf{x}_i \in \mathcal{X} \in \mathbb{R}^{M \times N}$ denotes an $M \times N$ -dimensional image sample i , generated by a mixture of K *conditionally independent* and *unobserved* factors of variation $\mathbf{w}_i \in \mathcal{W} \in \mathbb{R}^K$. Hence, in an unsupervised setting, \mathbf{x}_i can be simulated using its source factors of variation, expressed as,

$$\mathbf{x}_i \sim Sim(\mathbf{w}_i).$$

In the supervised setting, the dataset can also include the labels of \mathbf{w}_i as $\mathbf{y}_i \in \mathcal{Y} \in \mathbb{R}^K$.

Disentanglement is then defined as the problem of learning an L -dimensional representation $\mathbf{z}_i \in \mathbb{R}^L$ for image sample \mathbf{x}_i , with

$$p(\mathbf{x}_i | \mathbf{z}_i) \sim Sim(\mathbf{w}_i).$$

In this representation, for each dimension j of the hidden variable vector,

$$\mathbf{z}_i = [z_i^j],$$

the algorithm learns a *single* source of variation w_i^j . Note that in theory, L should be selected as $L \geq K$ in order to capture the discriminative information about all of the hidden properties.

4 A Critical Look at β Variational Auto-Encoders (β -VAEs)

β -VAEs are one of the pioneering extensions of vanilla VAE [3, 14], that generate a representation for the data. Employing an encoder-decoder architecture, β -VAEs estimate a Gaussian distribution of each source of variation, in the latent space.

Formally, given a dataset $\mathcal{D} = \{\mathcal{X}, \mathcal{W}, \mathcal{Y}\}$, β -VAE obtains a representation (denoted by \mathbf{z}_i), which is sampled from learned Gaussian distributions, $\mathcal{N}(\mu_i, \mathbf{v}_i)$, for $i = 1, \dots, L$, where L is the number of hidden variables.

An encoder-decoder architecture is used to estimate $\mathcal{N}(\mu_i, \mathbf{v}_i)$, for the latent representation \mathbf{z}_i and reconstructed data, $\bar{\mathbf{x}}_i$. At training step, the encoder learns the parameters of the Gaussian distribution; μ_i and \mathbf{v}_i . The latent representation \mathbf{z}_i of sample \mathbf{x}_i is then sampled from the learned distribution.

β -VAE is trained to minimize the Evidence Lower BOund (ELBO):

$$\mathcal{L}_{\beta\text{-VAE}} = -E_{q(\mathbf{z}|\mathbf{x})} \left[\log \underbrace{p(\mathbf{x}|\mathbf{z})}_{\text{Decoder}} \right] \quad (2)$$

$$+ \beta \cdot D_{KL} \left(\underbrace{q(\mathbf{z}|\mathbf{x})}_{\text{Encoder}} \parallel p(\mathbf{z}) \right), \quad (3)$$

where the first term enhances the reconstruction quality of the images, compelling the generated images to match the original image, whereas the second term enforces the learned distributions to

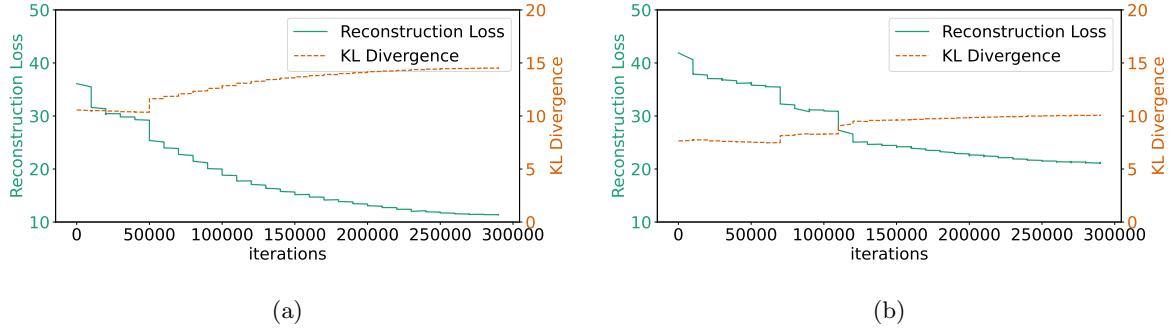


Fig. 1: Training curves for reconstruction loss and KL-Divergence for (a) VAE and (b) β -VAE with $\beta = 2$. (a) KL divergence increases during training due to the difference between the two loss terms. (b) β -VAE controls the difference between the two terms and the increase in the second term. Although the KL-divergence term does not increase as much as we observe in (a), it still does not decrease as expected.

approach a predefined distribution, which is generally selected to be a Gaussian distribution with zero mean and unit variance ($p(\mathbf{z}) = \mathcal{N}(0, 1)$). The distributions of the encoder $q(\mathbf{z}|\mathbf{x})$ and the decoder $p(\mathbf{x}|\mathbf{z})$ are parameterized by neural networks.

Notice that, for $\beta = 1$, the model reduces to the original VAE. In this case, the terms of the loss function, namely, mean squared error and KL divergence, becomes incompatible in terms of their order of magnitudes. Additionally, they pose different learning patterns with regards to the number of epochs and the learning rates for convergence. In order to compensate for these incompatibilities the KL-divergence term is scaled by a hyperparameter $\beta > 1$. This adjustment only harmonizes the numerical ranges between the terms of the loss function, ignoring the implicit learning difficulties, particularly in terms of training speeds, associated with each term.

4.1 Analysis on the Effect of the Hyperparameter β on the Learning Dynamics.

We start our research with an analysis on the effect of the hyper parameter β on the learning dynamics of β -VAE and provide a basis for our study. First, we investigate the learning dynamics between reconstruction and KL-divergence terms of Equation 2. We analyze the relative changes in these terms during the training phase, for different values of the hyperparameter β . Our observations are summarized below.

Observation 1: The hyperparameter β helps balancing the ranges of the reconstruction loss and the KL divergence. Figure 1a compares the reconstruction loss and KL divergence values during the training of a VAE, i.e., β -VAE with $\beta = 1$. We observe that the values of the reconstruction loss are relatively higher than the KL divergence values. As a consequence, the reconstruction loss dominates the loss function, and the contribution of the KL divergence term to the overall loss remains relatively small. Since the reconstruction loss dominates the overall loss, minimizing the KL divergence in parallel to the reconstruction loss becomes a challenge. This imbalance between the values of reconstruction loss and KL divergence term results in an increase in the KL term, as the overall loss function converges to an optimal value. We believe that the leading cause of VAE's failure to disentangle the learned representation is the discrepancy between the values of the reconstruction loss and KL divergence term. Figure 1b, on the other hand, shows the behaviour of the same losses during the training phase for $\beta = 2$. The figure suggests that increasing the hyperparameter β compensates for the discrepancy between the values of the two terms and dampens the increase of the KL term.

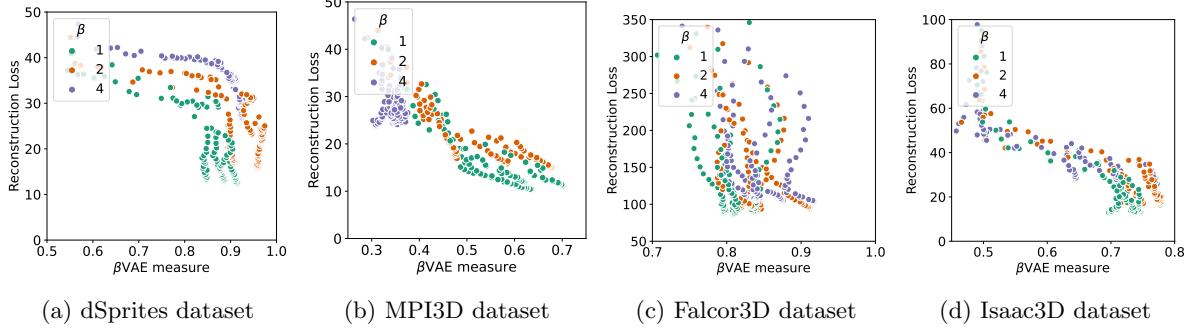


Fig. 2: Reconstruction loss versus β -VAE disentanglement metric across different β hyperparameters for (a) dSprites, (b) MPI3D, (c) Falcor3D, and (d) Isaac3D datasets. Each point represents a model trained with a different hyperparameter setup (altered learning rate, number of iterations, and batch size). Points are color-coded based on the value of β .

amount of disentanglement is inversely proportional to the amount of reconstruction loss.

We further investigate the effect of β on the relationship between disentanglement and reconstruction. We compare disentanglement properties on four datasets; dSprites [29], MPI3D [13], Falcor3D [31] and Isaac3D [31] (details in Section 6.1). We analyze the results of several runs covering a large hyperparameter space consisting of varying learning rates, batch sizes, and iteration counts (details in Section 6.3). In this experiment, we have thoroughly investigated the relationship between the reconstruction and disentanglement terms by systematically varying the value of the hyperparameter β across the hyperparameters of the network. Figure 2a shows the results of this analysis. The x and y -axes show the β -VAE disentanglement measure, and reconstruction loss, respectively. Each point represents a model, trained with a different hyperparameter setup, with learning rate, batch size, and number of iterations (see Section 6.3 for details). Points are color-coded based on the value of β . In order to visually observe the effect of different β values on the reconstruction loss and KL-divergence term, the values of other parameters are not specified in this Figure.

Observation 3: Disentanglement also depends on hyperparameters other than β .

Figure 2a shows that for the dSprites dataset, disentanglement vs. reconstruction dynamic highly depends on the selection of the hyperparameters, such as the learning rate or batch size. As we increase the β parameter, selecting the rest

of the hyperparameters requires a larger search space. Thus, finding the optimal hyperparameter set becomes more difficult. We also observe that increasing the β parameter from 1 to 2 gives better disentanglement values with the cost of higher reconstruction loss. These observations support the arguments of [5], who claimed β -VAE increases the reconstruction loss. We observe similar behavior for the Falcor3D dataset (See Figure 2c).

Observation 4: $\beta < 1$ can surprisingly increase disentanglement. Further analysis on the MPI3D dataset reveals more insight into the effect of β . We have conducted the same experiment on this dataset (see Figure 2b). The results on this dataset complement the dSprites experiments. Higgins et al. [14] argued increasing the importance of KL Divergence yields better disentanglement and hence selected $\beta > 1$. However, our results show that lower values of β tend to provide better results in terms of lower reconstruction loss and higher disentanglement scores in this specific dataset. The flow of Figure 2b reveals that $\beta < 1$ values should also be investigated. These findings support the results in [11]. The behavior of the Isaac3D dataset is similar to the MPI3D dataset (See Figure 2d). We observe a decrease in the disentanglement measure as we increase the β parameter.

4.2 A Critique on β -VAE

β -VAE has three major drawbacks: First, it introduces a new parameter β to hyperparameter

space, which is to be tuned together with other hyperparameters of the model using expensive empirical methods in a large search space. Second, selecting a hyperparameter $\beta > 1$ to improve the amount of disentanglement may result in a relatively poor representation [5] (Observation 2). Third, the ratio between reconstruction loss and the amount of disentanglement heavily depends on the network hyperparameters other than β (Observation 3). The optimal value of β can also be less than 1 for some datasets, further increasing the very large search space (Observation 4).

The above drawbacks highlight the importance of an effective learning model for the adjustment of the loss function weights. We believe that an optimal balance between the reconstruction loss and the degree of disentanglement requires a dynamic learning model. Our model L-VAE has a self-learning mechanism that can simultaneously optimize the model parameters and the weights of the loss function without any restriction on the range of these weights, as will be described in the next section.

5 Learnable VAE (L-VAE)

In this section, we describe the proposed Learnable VAE (L-VAE), which circumvents some of the drawbacks associated with the hyperparameter selection problem of β -VAE. The proposed L-VAE can achieve lower reconstruction loss values than β -VAE, while producing better disentanglement scores.

Our method is based on the multi-task learning method of Kendall et al. [19], where they learn the relative weights of different tasks in a loss term by augmenting them to the optimizer. Without loss of generality, let us assume that the overall loss function consists of two terms, $\mathcal{L}_0(\mathbf{x})$ and $\mathcal{L}_1(\mathbf{x})$, each of which are the functions of the input vector, \mathbf{x} . Then, the trade off between the terms of the loss function can be dynamically learned from the following analytical form:

$$\mathcal{L}(\mathbf{x}) = \frac{1}{\sigma_0^2} \mathcal{L}_0(\mathbf{x}) + \frac{1}{\sigma_1^2} \mathcal{L}_1(\mathbf{x}) + \log(\sigma_0) + \log(\sigma_1), \quad (4)$$

where $1/\sigma_i^2$, for $i = 0, 1$, are the balancing weights. The parameters σ_i are optimized and updated

with other learnable weights of the neural network. The last two terms of Equation 4 regularize the learned weights, σ_0 and σ_1 .

Building upon the same framework of Kendall et al. [19], we develop L-VAE such that it learns the relative weights of the terms of the VAE loss function by updating Equation 2, as follows:

$$\begin{aligned} \mathcal{L}_{\text{L-VAE}} = & -\frac{1}{\sigma_0^2} E_{q(\mathbf{z}|\mathbf{x})} [\log p(\mathbf{x}|\mathbf{z})] \\ & + \frac{1}{\sigma_1^2} \mathcal{D}_{KL}(q(\mathbf{z}|\mathbf{x}) \parallel \mathbf{z}) + \sum_{i=0,1} \sigma_i^2. \end{aligned} \quad (5)$$

In order to regulate the upper limit of the weight parameter σ_i in Equation 6, we introduce a regularization term $\sum \sigma_i^2$. The σ_i terms are added to the parameter set of the optimizer and are estimated through simultaneous learning with the network parameters (refer to Section 7.1 for the learning curves of σ_i).

Except the relative weights in the loss function, the rest of the network architecture is constructed exactly the same as β -VAE. Therefore, a straightforward and fair comparison between β -VAE and L-VAE is possible through a simple conversion of the σ_i parameters to the β parameter. In Section 7, we show the correspondence of β and σ_i parameters. During the experiments, we observe that, after the training phase, the ratio of the learned parameters, $\frac{\sigma_0^2}{\sigma_1^2}$ in L-VAE is aligned with the empirically tuned value of β parameter in β -VAE.

6 Experimental Setup

In this section, we describe the datasets, the encoder-decoder architectures, and the performance measures employed in our experiments.

6.1 Datasets

We conduct our experiments on four popularly used disentanglement data sets:

1. dSprites dataset [29], which is a 2D-shapes dataset with 700K 64x64 images containing white 2D shape images (heart, square, ellipse) on black background. There are five factors of variation to disentangle: shape, scale, orientation, and X, and Y positions of the object.
2. MPI3D-complex dataset [13], which consists of four real-world objects moving on a robotic arm

- leading to 460K 64×64 colored images. There are seven factors of variation to disentangle: color shape, size, camera height, background color, and horizontal and vertical axes. We will refer to this dataset as MPI3D.
3. Falcor3D dataset [31], which contains the images of a living room containing with different lighting conditions containing 233K 64×64 images. There are seven factors of variation to disentangle: lighting intensity, directions x, y, and z of lighting, and x, y, and z coordinates of the camera position.
 4. Isaac3D dataset [31], which contains a robotic arm holding an object in a kitchen. Lighting conditions, camera position, and the position of the arm are altered. The dataset contains 737K 64×64 images. There are nine factors of variation to disentangle: Objects shape, scale, and color, wall color, camera height, robotic arms' x and y positions, lighting intensity and direction.

We randomly split all datasets into training, validation, and test sets to conduct our experiments. The training set covers around 85% of the dataset and the rest is equally split into test and validation sets.

6.2 The Compared Methods

We compare the proposed L-VAE with five well-known disentanglement methods:

1. Variational Autoencoder (VAE) [22], which optimizes reconstruction loss and KL divergence with equal weights.
2. β Variational Autoencoder (β -VAE) [14], which utilizes an empirically tuned hyperparameter β to weight KL divergence.
3. ControlVAE [37, 38], which learns the β parameter, based on a PID control algorithm at each iteration.
4. DynamicVAE [39], which is a slightly modified version of ControlVAE.
5. σ -VAE [35], which can be considered as an instantiation of our model L-VAE. It learns the weight of the reconstruction loss, whereas our model L-VAE learns the weights of both reconstruction loss and KL divergence.

6.3 The Details of L-VAE Model Architecture(s)

We perform our experiments with an MLP Encoder-Decoder architecture for the dSprites dataset and a CNN Encoder-Decoder architecture for the MPI3D, Falcor3D, and Isaac3D datasets (see Table 1 for the architecture details). All of the architecture uses the ReLU activation function on all hidden layers and the Sigmoid activation function at the decoder output. We use the Adam optimizer [21] with $\beta_1, \beta_2 = (0.9, 0.999)$, and $\epsilon = 1e - 08$. Mean Square Error (MSE) is used as the reconstruction loss.

We empirically tune the hyperparameters (namely, batch size, learning rate, and iteration count) for both the compared methods and L-VAE. Batch size is selected from {32, 64, 128, 256}, and the number of iterations from $\{1, 2, \dots, 30\} \times 10^4$. We used OneCycleLR optimization for learning rate where we initialized learning rate with 1e-5 and increased it to 1e-4 for the first 150000 iterations then we decrease the learning rate to 1e-6 using Cosine annealing strategy. For ControlVAE, we set the desired KL value to 18, K_p to 0.0.01, K_i to -0.001 and initialized β with 0 (following [37]). For DynamicVAE, we set the desired KL value to 18, K_p to 0.0.01, K_i to -0.005, and initialized β with 150 (following [39]).

Configuring experiments with this set of hyperparameters, VAE and L-VAE experiments are carried out independently for 120 models. For β -VAE, β is selected from {2,4} (higher β results in higher reconstruction losses for the datasets we have experimented on – see Section 4.1 for details), which leads to 240 independent models trained for β -VAE. We select the best set of hyperparameters for all models based on validation scores on the disentanglement measure with the β -VAE measure [14].

We set the latent dimension size to five for the dSprites dataset, seven for the MPI3D and Falcor3D datasets, and nine for Isaac3D dataset following the number of labeled attributes provided with the datasets [13, 29, 31]. All the σ_i values are initialized with one (1) for the L-VAE experiments.

Table 1: The encoder and decoder architecture we used in our experiments. For fully connected (FC) layers, the number of output features is given in parentheses. For convolutional layers (Conv) and transposed convolutional layers (ConvT), the number of input and output features is also given in parentheses.

Model	Model Details
MLP $E(\cdot)$	FC(1200), FC(1200), FC(2xLatent_dim)
MLP $D(\cdot)$	FC(1200), FC(1200), FC(HxWxC)
CNN $E(\cdot)$	Conv(32x4x4, stride=2), Conv(32x4x4, stride=2), Conv(64x4x4, stride=2), Conv(64x4x4, stride=2), Conv(32x4x4, stride=1), FC(2xLatent_dim)
CNN $D(\cdot)$	FC(256), ConvT(64x4x4, stride=2), ConvT(64x4x4, stride=2), ConvT(32x4x4, stride=2), ConvT(32x4x4, stride=2), ConvT(3x4x4, stride=2), ConvT(3x4x4, stride=2)

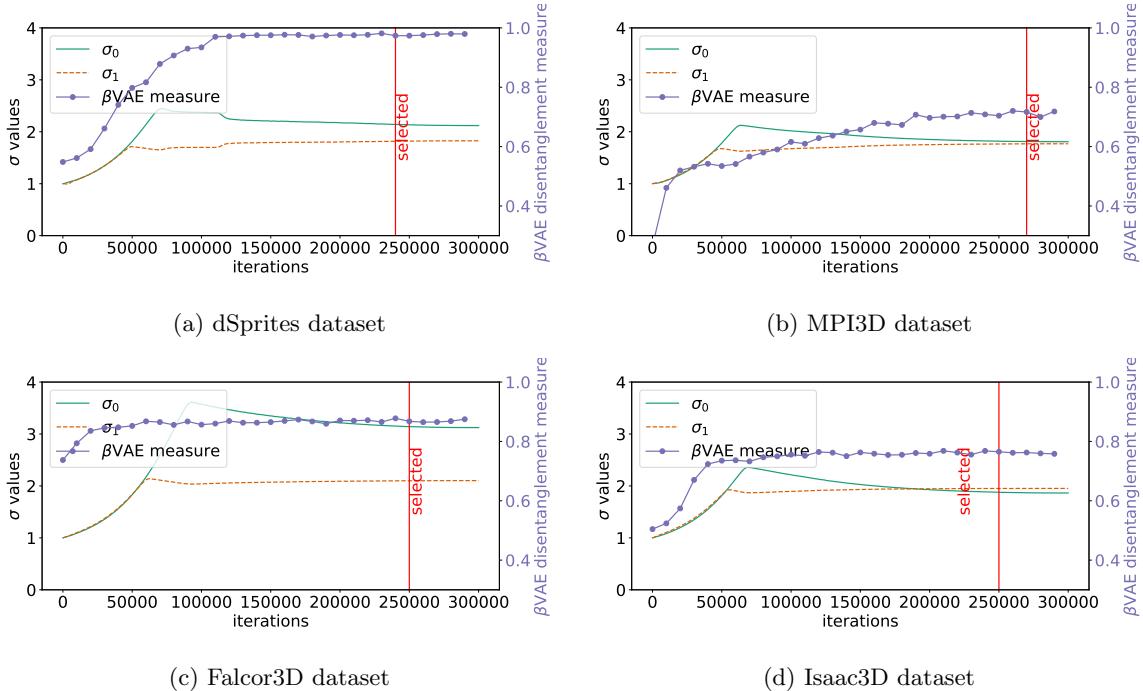


Fig. 3: Training curves of the learned relative weights (σ_0 and σ_1) of LVAE model for (a) the dSprites, (b) the MPI3D (c) the Falcor3D, and (d) the Isaac3D datasets. The green line corresponds to the β -VAE disentanglement metric computed at every 10,000 iterations. We note that σ_i reaches a peak around 50,000 iterations, after which the change decelerates. We determined the number of iterations (along with other hyper-parameters, learning rate, and batch size) based on the β -VAE disentanglement measure on validation dataset (see Section 6.5 for details). Red vertical lines show this selection.

6.4 Disentanglement Measures

In their review on disentanglement measures, Carboneau et al. [4] argue that a disentangled model should be evaluated with respect to three major properties: Explicitness, compactness, and modularity. For evaluating the performances in

our experiments, we selected the following disentanglement measures so that the three major properties are covered:

- **Explicitness** evaluates a model’s ability to recover the input from the representation. We

have selected the *explicitness score* [34] to measure explicitness.

- **Compactness** signifies that a small portion of the representation (ideally one dimension) represents a single attribute. We have selected the *Separated Attribute Predictability (SAP) score* [23], and *Mutual Information Gap (MIG) score* [5] to measure compactness.
- **Modularity** pertains to whether or not a change of a factor affects only a single dimension of the representation. We have selected the β -VAE [14] and the *Factor VAE* [20] measures to quantify modularity.

Some disentanglement measures encompass two or more of the above mentioned properties; in such instances, the measure is referred to as holistic [4]. We have selected *Interventional Robustness Score (IRS)* [40], which measures modularity and explicitness.

Overall, we have selected six disentanglement measures for which we use the implementation provided by Carbonneau et al. [4]. Although these measures are commonly used in the literature, they define disentanglement differently or quantify different aspects of disentanglement. Thus, the ranking of each method may change across the datasets.

6.5 Model Selection

As stated in Section 6.3, we have conducted our experiments to cover a relatively large hyperparameter set (i.e., selection of batch size, learning rate, and iteration count). There are two significant problems in determining the best set of hyperparameters of the models in the validation set: Firstly, finding a balance between the reconstruction loss and KL divergence in the overall loss function (Equation 6) is still an open research issue. Secondly, choosing a suitable measure for quantifying the amount of disentanglement brings a strong bias toward the selected measure. Therefore, empirically finding an "optimal" hyperparameter set is not a well-defined problem in disentangled representation learning. For the sake of being fair in comparing the proposed L-VAE with the other VAE models, we selected the hyperparameters based on the highest β -VAE score, which is proposed along with β -VAE in [14]. We use the best scores achieved on the validation set for the hyperparameter selection for all methods.

7 Quantitative Experiments

In this section, first, we analyse the learned weights of the proposed L-VAE. Then, we evaluate the performance of our L-VAE model in comparison with the baseline VAE models. We measure the performances by employing the reconstruction loss and disentanglement measures, which are selected from the literature, as explained in the previous section. Finally, we carry out an ablation study to explore the effects of a dynamic learning strategy proposed in this paper.

7.1 Experiment 1: Convergence of the Weights Learned by L-VAE

In this set of experiments, we analyze how the parameters σ_0 and σ_1 (Equation 6) change over time during the learning phase. In all experiments, we select the σ_i values that maximize the β -VAE score.

During the derivations of L-VAE, we mentioned that there is a correspondence between the empirically tuned β parameter of β -VAE model and the optimal ratio of the learned parameter of L-VAE model,

$$\hat{\beta} = \frac{\sigma_0^2}{\sigma_1^2}.$$

Hence, we shall also investigate the behavior of the optimal ratio $\hat{\beta}$ for each dataset.

Figure 3 shows the learning curves of σ_i parameters. We notice that, in the dSprites dataset, σ_i exhibits a steep ascent until 50K iterations, reaching a peak value, after which the change attenuates. However, the β -VAE score slightly decreases as σ_0 decreases. During the cross-validation step, the σ_i values, obtained at 50Kth iteration is selected. These values corresponds to the maximum β -VAE score.

The learning curves of σ_i parameters for the rest of the datasets show similar behaviors. However, the β -VAE scores keep increasing during the training phase. As β -VAE scores increase, we observe that the ratio of σ_0^2/σ_1^2 also changes: For the dSprites, and Falcor3D datasets, σ_0 surpasses σ_1 , yielding the generally practiced estimation for

$$\hat{\beta} = \frac{\sigma_0}{\sigma_1} > 1.$$

On the other hand, in the MPI3D, and Isaac3D dataset, σ_1 overtakes σ_0 , yielding an unusual estimation for

Table 2: Quantitative comparison of VAE, β -VAE, ControlVAE, DynamicVAE, σ -VAE, and L-VAE based on reconstruction loss and six disentanglement measures across four datasets.

Dataset	Model	Parameters	Reconstruction ↓	β -VAE ↑	FactorVAE ↑	Explicitness ↑	IRS ↑	MIG ↑	SAP ↑
dSprites	VAE [21]	$\beta = 1$	12.46	0.91	0.62	0.50	0.34	0.11	0.09
	β -VAE [14]	$\beta = 2$	25.04	0.97	<u>0.74</u>	0.58	<u>0.59</u>	0.30	0.27
	ControlVAE [37]	-	28.00	<u>0.96</u>	0.71	0.57	0.57	0.34	0.34
	DynamicVAE [39]	-	31.57	0.93	0.71	0.55	0.53	<u>0.31</u>	0.28
	σ -VAE [35]	-	29.21	0.85	0.61	0.51	0.38	0.10	0.05
	L-VAE	$\hat{\beta} = \frac{\sigma_0^2}{\sigma_1^2} = 1.39$	<u>21.14</u>	0.97	0.77	0.59	0.63	0.30	0.29
MPI3D	VAE	$\beta = 1$	11.21	<u>0.70</u>	0.32	<u>0.41</u>	<u>0.32</u>	0.16	<u>0.18</u>
	β -VAE	$\beta = 2$	15.31	0.67	<u>0.44</u>	0.36	0.31	0.16	0.17
	ControlVAE	-	14.34	0.67	0.42	0.35	0.32	<u>0.13</u>	0.12
	DynamicVAE	-	15.81	0.54	0.35	0.30	0.30	0.12	0.13
	σ -VAE	-	5.42	0.66	0.38	0.48	0.44	0.03	0.02
	L-VAE	$\hat{\beta} = \frac{\sigma_0^2}{\sigma_1^2} = 1.05$	<u>10.79</u>	0.71	0.46	0.39	<u>0.32</u>	0.16	0.20
Falcon3D	VAE	$\beta = 1$	215.03	0.87	0.42	0.58	<u>0.32</u>	0.04	0.03
	β -VAE	$\beta = 4$	105.17	0.92	0.61	0.66	0.33	0.07	0.05
	ControlVAE	-	187.41	0.78	0.40	0.40	0.20	<u>0.06</u>	0.07
	DynamicVAE	-	216.87	0.72	0.30	0.38	0.16	0.06	<u>0.06</u>
	σ -VAE	-	78.82	<u>0.89</u>	0.39	<u>0.61</u>	0.30	0.04	0.03
	L-VAE	$\hat{\beta} = \frac{\sigma_0^2}{\sigma_1^2} = 2.34$	<u>97.97</u>	0.88	<u>0.47</u>	0.66	0.30	0.05	0.05
Isaac3D	VAE	$\beta = 1$	<u>13.37</u>	0.75	0.48	<u>0.56</u>	0.24	0.07	0.07
	β -VAE	$\beta = 2$	17.08	0.78	<u>0.52</u>	0.54	0.31	0.22	0.19
	ControlVAE	-	27.45	0.60	0.34	0.42	0.24	0.13	0.14
	DynamicVAE	-	35.55	0.49	0.27	0.37	0.21	0.13	0.13
	σ -VAE	-	23.09	0.67	0.49	0.55	0.33	0.05	0.04
	L-VAE	$\hat{\beta} = \frac{\sigma_0^2}{\sigma_1^2} = 0.95$	12.97	<u>0.76</u>	0.61	0.57	<u>0.32</u>	0.17	<u>0.15</u>

$$\hat{\beta} = \frac{\sigma_0}{\sigma_1} \leq 1.$$

The above result is rather counter-intuitive, considering the fact that disentanglement is accentuated for $\hat{\beta} > 1$.

7.2 Experiment 2: Comparison with Baseline Methods

In this set of experiments, we compare VAE, β -VAE, ControlVAE, DynamicVAE, σ -VAE, (see Section 2.2 for a brief description of the methods) and L-VAE based on the MSE reconstruction

loss and the six disentanglement measures [4] mentioned in the previous subsection. Table 2 shows the results for all four datasets. For β -VAE, the β value is determined through a hyperparameter search process described in Section 6.5, whereas, for L-VAE, the values of σ_i represent the learned values.

First, we compare the learned value of $\hat{\beta} = \sigma_0^2/\sigma_1^2$ with the empirically tuned β parameter. The learned value of $\hat{\beta}$ is 1.39, 1.05, 2.34, and 0.95 for dSprites, MPI3D, Falcor3D, and Isaac3D datasets, respectively. These values are consistent with our findings in Figure 2, which suggests that the optimal value of β might be lower than 1 for the Isaac3D dataset. These results are aligned with the ratio of the weights $\hat{\beta} = \sigma_0^2/\sigma_1^2$ learned by L-VAE.

To assess the disentanglement performances of the different models, we examine their disentanglement properties (modularity, explicitness, and compactness) separately. Regarding modularity (looking at the β -VAE and FactorVAE measures), L-VAE achieves better scores in the dSprites and MPI3D datasets. However, it performs on par in other datasets. Similarly, with respect to the explicitness score, L-VAE achieves the best performance for the dSprites, Falcor3D, and Isaac3D datasets. Regarding the compactness of representations (the SAP score), L-VAE achieves better compactness in the MPI3D dataset. Finally, we compare the models with the holistic IRS measure, which combines modularity and explicitness properties. The best score is achieved with the L-VAE model in dSprites dataset.

Overall, the results suggest that L-VAE can learn weights (σ_i) consistent with our preliminary observations where we show that increasing β can lead to higher reconstruction loss and $\beta < 1$ can produce higher disentanglement scores (See Observations 2 and 4 in Section 4.1, and Figure 2). Moreover, L-VAE generally produces better or on par reconstructions. Although the six disentanglement measures do not suggest a consistent ordering among the methods, we see that L-VAE achieves superior or on par performance with respect to many measures on all datasets.

7.3 Experiment 3: Ablation Study

In order to further investigate the effect of learning the hyperparameters of L-VAE on the reconstruction loss and the disentanglement term, we train a standalone β -VAE model with the learned weights as $\beta = \sigma_0^2/\sigma_1^2$ as an ablation study. The results of these experiments are shown in Table 3 for all datasets.

Notice that the reconstruction loss of the β -VAE trained by the learned parameter $\hat{\beta}$ is close to that of the L-VAE, except for the dSprites dataset. Moreover, training β -VAE with the $\hat{\beta}$ weights learned with L-VAE exhibits similar or inferior performance than L-VAE in terms of the β -VAE measure in the dSprites dataset, which signifies the importance of a dynamic weighting strategy.

8 Qualitative Experiments

Following the common practice in the literature [5, 14, 39], we provide qualitative results for L-VAE on the CelebA dataset [26]. CelebA consists of 202K images of celebrity faces labeled with facial attributes. There are 40 facial attribute labels, such as baldness, wearing eyeglasses, and pale skin. We cropped the background from the images and downsampled the dataset to size 128×128 following [5]. Furthermore, we have used 12% of the dataset to train the L-VAE model for simplicity. We have trained the CNN Encoder-Decoder architecture in Table 1, with the batch size of 128 and the learning rate of 1e-4 for 1M iterations.

In order to assess the disentanglement ability of L-VAE qualitatively, we analyze the latent-space traversals through reconstructions as suggested in the literature [5, 14, 39]. After encoding sample images, we acquire latent representations for images. We select a specific latent dimension and alter its value while keeping other dimensions unchanged.

Figure 4 shows the results of latent traversals for nine different dimensions. As shown in the Figure, altering a single latent dimension consequently changes a single attribute in facial images; for example, Figure 4i shows that people are aged as we alter the value of a specific dimension.



Fig. 4: Latent traversals of the L-VAE model for the CelebA dataset. We select a single dimension and acquire reconstructions while altering its value. Note that other dimensions are kept unchanged in these experiments. Each subfigure corresponds to the traversal of a single latent dimension. The ranges of the alteration are shown in brackets.

9 Conclusion

In this study, we propose an extension to the classical β -VAE. The proposed model, called, L-VAE, dynamically learns the relative weights of reconstruction loss and KL divergence term. Our study is inspired from the findings of Locatello et al. [27], which argue that hyperparameter selection has more impact on the disentanglement properties compared to the model selection itself. Hence, we suggest a straightforward and efficient algorithm for learning the hyperparameters of the loss

function. The proposed optimization methodologies are also applicable to similar deep learning models.

The foundation of our study rests upon the power of β -VAE in learning a disentangled representation with two major challenges:

First of all, β -VAE increases the disentanglement abilities of the VAE model, at the cost of decreased reconstruction quality. Secondly, the introduction of the hyperparameter β expands the search space of hyperparameters. The proposed L-VAE provides partial remedies to these problems.

Table 3: An ablation study, where we train a standalone β -VAE with $\beta = \hat{\beta} = \sigma_0^2/\sigma_1^2$. The weights σ_i are learned from the suggested L-VAE model. The value of $\hat{\beta} = \sigma_0^2/\sigma_1^2$ are given in the second column. This setup results in a higher reconstruction loss and lower β -VAE measure than we have obtained with L-VAE, indicating the importance of a dynamic learning strategy.

Dataset	Model	Parameters	Reconstruction ↓	β -VAE ↑	FactorVAE ↑	Explicitness ↑	IRS ↑	MIG ↑	SAP ↑
dSprites	β VAE	$\beta = \hat{\beta}$	18.13	0.96	0.74	0.49	0.50	0.16	0.07
	L-VAE	$\hat{\beta} = \frac{\sigma_0^2}{\sigma_1^2} = 1.39$	21.14	0.97	0.77	0.59	0.63	0.30	0.29
MPI3D	β VAE	$\beta = \hat{\beta}$	10.63	0.74	0.46	0.39	0.32	0.17	0.21
	L-VAE	$\hat{\beta} = \frac{\sigma_0^2}{\sigma_1^2} = 1.05$	10.79	0.71	0.46	0.39	0.32	0.16	0.20
Falcor3D	β VAE	$\beta = \hat{\beta}$	98.36	0.93	0.65	0.61	0.35	0.12	0.09
	L-VAE	$\hat{\beta} = \frac{\sigma_0^2}{\sigma_1^2} = 2.34$	97.97	0.88	0.47	0.66	0.30	0.05	0.05
Isaac3D	β VAE	$\beta = \hat{\beta}$	12.35	0.77	0.52	0.58	0.25	0.07	0.06
	L-VAE	$\hat{\beta} = \frac{\sigma_0^2}{\sigma_1^2} = 0.95$	12.97	0.76	0.61	0.57	0.32	0.17	0.15

L-VAE can estimate the optimal ratio of weights concerning the trade-off between reconstruction loss and the disentanglement of the learned representation without introducing additional hyperparameters to the model. Our experiments demonstrate that the L-VAE outperforms or match the state of the art methods in disentanglement. Moreover, L-VAE learns remarkable disentangled representations, while yielding substantially low reconstruction losses.

L-VAE can learn the weights of the losses without any assumptions on the dynamic range of the hyperparameters. A common assumption made in the literature is to select a higher weight for the KL divergence to ensure better disentanglement[14]. However, based on our results, we showed that selecting a smaller weight for KL divergence may result in higher disentanglement scores in some datasets, such as Isaac3D dataset. L-VAE learns the optimal weights, which establish a very sensitive balance between the reconstruction loss and KL divergence term without modifications to the model or the hyperparameter space.

We made an interesting comparison between the β VAE and our L-VAE: In our experimentation, we train the classical β -VAE model with the weights learned at the output of the proposed L-VAE model. β -VAE model achieves better disentanglement scores with the learned β values, with the cost of an increased reconstruction loss. This observation demonstrates the significance of the dynamic learning process suggested in L-VAE.

10 Acknowledgments

We also gratefully acknowledge the computational resources kindly provided METU ImageLab and METU Robotics and Artificial Intelligence Technologies Application and Research Center (ROMER).

References

- [1] Åström KJ, Hägglund T (2006) Advanced PID control. ISA-The Instrumentation, Systems and Automation Society

- [2] Bengio Y, Courville A, Vincent P (2013) Representation learning: A review and new perspectives. *IEEE transactions on pattern analysis and machine intelligence* 35(8):1798–1828
- [3] Burgess CP, Higgins I, Pal A, et al (2018) Understanding disentangling in β -vae. arXiv preprint arXiv:180403599
- [4] Carboneau MA, Zaidi J, Boilard J, et al (2022) Measuring disentanglement: A review of metrics. *IEEE Transactions on Neural Networks and Learning Systems*
- [5] Chen RT, Li X, Grosse RB, et al (2018) Isolating sources of disentanglement in variational autoencoders. In: *Advances in Neural Information Processing Systems*, pp 2610–2620
- [6] Chen X, Duan Y, Houthooft R, et al (2016) Infogan: Interpretable representation learning by information maximizing generative adversarial nets. In: *Advances in neural information processing systems*, pp 2172–2180
- [7] Denton EL, et al (2017) Unsupervised learning of disentangled representations from video. In: *Advances in neural information processing systems*, pp 4414–4423
- [8] Do K, Tran T (2019) Theory and evaluation metrics for learning disentangled representations. arXiv preprint arXiv:190809961
- [9] Dong Y, Liu Y, Zhang H, et al (2020) Fd-gan: Generative adversarial networks with fusion-discriminator for single image dehazing. In: *Proceedings of the AAAI Conference on Artificial Intelligence*, pp 10729–10736
- [10] Duan B, Fu C, Li Y, et al (2020) Cross-spectral face hallucination via disentangling independent factors. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp 7930–7938
- [11] Fil M, Mesinovic M, Morris M, et al (2021) beta-vae reproducibility: Challenges and extensions. arXiv preprint arXiv:211214278
- [12] Fu Z, Chen X, Liu D, et al (2023) Multi-level feature disentanglement network for cross-dataset face forgery detection. *Image and Vision Computing* 135:104686
- [13] Gondal MW, Wuthrich M, Miladinovic D, et al (2019) On the transfer of inductive bias from simulation to the real world: a new disentanglement dataset. *Advances in Neural Information Processing Systems* 32
- [14] Higgins I, Matthey L, Pal A, et al (2016) beta-vae: Learning basic visual concepts with a constrained variational framework. In: *International conference on learning representations*
- [15] Higgins I, Amos D, Pfau D, et al (2018) Towards a definition of disentangled representations. arXiv preprint arXiv:181202230
- [16] Hsieh JT, Liu B, Huang DA, et al (2018) Learning to decompose and disentangle representations for video prediction. In: *Advances in Neural Information Processing Systems*, pp 517–526
- [17] Karras T, Laine S, Aila T (2019) 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
- [18] Karras T, Laine S, Aittala M, et al (2020) Analyzing and improving the image quality of stylegan. In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp 8110–8119
- [19] Kendall A, Gal Y, Cipolla R (2018) Multi-task learning using uncertainty to weigh losses for scene geometry and semantics. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp 7482–7491
- [20] Kim H, Mnih A (2018) Disentangling by factorising. arXiv preprint arXiv:180205983
- [21] Kingma DP, Ba J (2014) Adam: A method for stochastic optimization. arXiv preprint arXiv:14126980

- [22] Kingma DP, Welling M (2013) Auto-encoding variational bayes. arXiv preprint arXiv:13126114
- [23] Kumar A, Sattigeri P, Balakrishnan A (2017) Variational inference of disentangled latent concepts from unlabeled observations. arXiv preprint arXiv:171100848
- [24] Lee HY, Tseng HY, Mao Q, et al (2020) Drit++: Diverse image-to-image translation via disentangled representations. International Journal of Computer Vision 128(10):2402–2417
- [25] Liu Y, Wei F, Shao J, et al (2018) Exploring disentangled feature representation beyond face identification. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp 2080–2089
- [26] Liu Z, Luo P, Wang X, et al (2015) Deep learning face attributes in the wild. In: Proceedings of International Conference on Computer Vision (ICCV)
- [27] Locatello F, Bauer S, Lucic M, et al (2019) Challenging common assumptions in the unsupervised learning of disentangled representations. In: international conference on machine learning, pp 4114–4124
- [28] Locatello F, Poole B, Rätsch G, et al (2020) Weakly-supervised disentanglement without compromises. In: International Conference on Machine Learning, PMLR, pp 6348–6359
- [29] Matthey L, Higgins I, Hassabis D, et al (2017) dsprites: Disentanglement testing sprites dataset. <https://github.com/deepmind/dsprites-dataset/>
- [30] Nguyen-Phuoc T, Li C, Theis L, et al (2019) HoloGAN: Unsupervised learning of 3d representations from natural images. arXiv
- [31] Nie W, Karras T, Garg A, et al (2020) Semi-supervised stylegan for disentanglement learning. In: Proceedings of the 37th International Conference on Machine Learning, pp 7360–7369
- [32] Nitzan Y, Bermano A, Li Y, et al (2020) Face identity disentanglement via latent space mapping. ACM Transactions on Graphics (TOG) 39(6):1–14
- [33] Pouyanfar S, Sadiq S, Yan Y, et al (2018) A survey on deep learning: Algorithms, techniques, and applications. ACM Computing Surveys (CSUR) 51(5):1–36
- [34] Ridgeway K, Mozer MC (2018) Learning deep disentangled embeddings with the f-statistic loss. Advances in neural information processing systems 31
- [35] Rybkin O, Daniilidis K, Levine S (2021) Simple and effective vae training with calibrated decoders. In: International Conference on Machine Learning, PMLR, pp 9179–9189
- [36] Schölkopf B, Locatello F, Bauer S, et al (2021) Toward causal representation learning. Proceedings of the IEEE 109(5):612–634
- [37] Shao H, Yao S, Sun D, et al (2020) Controlvae: Controllable variational autoencoder. In: International Conference on Machine Learning, PMLR, pp 8655–8664
- [38] Shao H, Xiao Z, Yao S, et al (2021) Controlvae: Tuning, analytical properties, and performance analysis. IEEE transactions on pattern analysis and machine intelligence 44(12):9285–9297
- [39] Shao H, Yang Y, Lin H, et al (2022) Rethinking controllable variational autoencoders. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp 19250–19259
- [40] Suter R, Miladinovic D, Schölkopf B, et al (2019) Robustly disentangled causal mechanisms: Validating deep representations for interventional robustness. In: International Conference on Machine Learning, PMLR, pp 6056–6065
- [41] Tran L, Yin X, Liu X (2017) Disentangled representation learning gan for pose-invariant face recognition. In: Proceedings of the IEEE conference on computer vision and pattern

recognition, pp 1415–1424

- [42] Yang M, Liu F, Chen Z, et al (2021) Causal-vae: disentangled representation learning via neural structural causal models. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp 9593–9602
- [43] Zhang C, Bengio S, Hardt M, et al (2021) Understanding deep learning (still) requires rethinking generalization. Communications of the ACM 64(3):107–115
- [44] Zhu Y, Min MR, Kadav A, et al (2020) S3vae: Self-supervised sequential vae for representation disentanglement and data generation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp 6538–6547