ORIGINAL ARTICLE

HGMVAE: hierarchical disentanglement in Gaussian mixture variational autoencoder

Jiashuang Zhou[1] · Yongqi Liu[1] · Xiaoqin Du[1]

Abstract

Recent advancements in deep neural networks have shown great potential in generating realistic data and performing clustering tasks. This is due to their ability to capture intricate patterns. However, current generative models face challenges such as poor performance and computational complexity caused by the issue of dimension disaster. The variational autoencoder (VAE), a commonly used method, also encounters problems such as posterior collapse and poor performance in multiclass classification when using the latent variables of VAE. Our goal in this study is to tackle the issue of effective disentanglement in image generation, classification and clustering tasks. We develop a generative network based on VAE incorporating a Gaussian mixture distribution as the prior. This enhancement improves the representation of latent variables and helps to overcome the challenges of matching the ground truth posterior. To further improve clustering performance, we introduce the total correlation as a kernel for computing latent features between embedding points and cluster centers. This technique is particularly useful in cases with complex latent variables and can also be applied for hierarchical disentanglement. Moreover, we employ the Fisher discriminant as a regularization term to minimize the within-class distance and maximize the between-class distance for samples, which has an important effect on the performance of our model viewed from the experimental results. We evaluate our proposed network on four datasets, and the experimental results demonstrate its effectiveness across multiple metrics.

Keywords Variational autoencoder · Disentanglement · Representation learning · Gaussian mixture distribution

1 Introduction

One of the fundamental goals in machine learning research is to construct models that possess a comprehensive understanding of the world. In the realm of supervised machine learning, two prominent methodologies have emerged: generative approaches and discriminative approaches, which give rise to generative models and discriminative models, respectively. Generative models, relying on joint distributions, capture a broader spectrum of data information and exhibit greater universality, whereas discriminative models focus on conditional distributions. Over the past few decades, significant efforts have been directed toward the exploration of generative models for image generation. Notable approaches include the utilization of generative adversarial networks (GANs) [1–4], variational autoencoder (VAE) [5–7], PixelCNN [8, 9], or diffusion models [10–12]. The encoder and decoder modules of autoencoder (AE) and VAE network architectures have found extensive application in various neural network frameworks. Additionally, the investigation of the variational lower bound serves as a typical implementation of optimal transport theory, and advancements in this aspect hold the potential to propel the development of optimal transport theory. Among these models, research on VAE models is regarded as more foundational and significant than others [6, 7, 15, 32].

From a modeling perspective, the VAE follows an autoencoder-like architecture consisting of an encoder and a decoder [13]. The objective is to not only achieve effec-

---

Jiashuang Zhou and Yongqi Liu have contributed equally to this work.

✉ Xiaoqin Du
  xiaoqindu@wtu.edu.cn

  Jiashuang Zhou
  2115363027@mail.wtu.edu.cn

  Yongqi Liu
  2115363016@mail.wtu.edu.cn

[1] Department of Computer Science and Artificial Intelligence, Wuhan Textile University, Wuhan 430200, Hubei Province, China

tive reconstruction of the input but also generate a latent representation that is meaningful and informative [14]. In the vanilla VAE, the approximate posterior is restricted to be a multivariate Gaussian with a diagonal covariance structure. However, this modeling approach suffers from non-identifiable. To address this issue, we sample from a more flexible distribution, specifically a mixture of Gaussians, which allows for a richer latent space representation. By incorporating a Gaussian mixture model (GMM) [15], comprising multiple Gaussian distributions, the model's marginal distribution over observed variables better captures the data characteristics. The effectiveness of Gaussian mixture VAE (GMVAE) has been demonstrated in various existing works. DLGMM [16] employs a mixture of Gaussian distribution as the approximate posterior for VAE, while VaDE [17] replaces the single Gaussian prior of VAE with a mixture of Gaussians, making it suitable for clustering tasks. Similarly, GMVAE [18] assumes a multimodal prior distribution to model complex data. Lee et al. [19] apply variational inference and a mixture of Gaussian prior optimized using the expectation–maximization (EM) algorithm for meta-learning. Additionally, Bai et al. [20] adopt a Gaussian mixture VAE and incorporate a contrastive loss to capture latent correlations for classification. Other works, such as Figueroa et al. [21] for semi-supervised learning, Collier et al. [22] for unsupervised clustering with continuous relaxation of discrete variables, Yang et al. [23]for handling complex spread in deep latent space using graph embedding, and Abdulaziz et al. [24] employing GMVAE with auxiliary loss functions, have also utilized GMVAE for various applications. In this paper, we focus on the generative manner, aiming to improve image synthesis performance with GMVAE. To the best of our knowledge, while some methods combine these approaches, there are distinctions in our specific modeling approach, resulting in superior results compared to existing methods.

In the quest to discover encoding functions that disentangle [25] high-level concepts from each other, the consciousness prior is regarded as one among several tools to guide the learner toward better high-level representations [26]. In the context of VAE-based models, the objective is to capture factors in the latent space through independent variables in the representation, which can be valuable for various downstream tasks. A notable attempt in this direction is the β-VAE [27], which introduced a regularizer hyperparameter, β, limit the capacity of the latent channel and exert implicit pressure for independence in the learned posterior. Theoretical analysis of β-VAE based on the information bottleneck principle [28] was provided by Burgess et al. [29]. Hu et al. explored that constraining mean variable alone can achieve better disentanglement and reconstruction performance and introduced mean constraint VAE [30]. Other models, such as FactorVAE [31], β-TCVAE
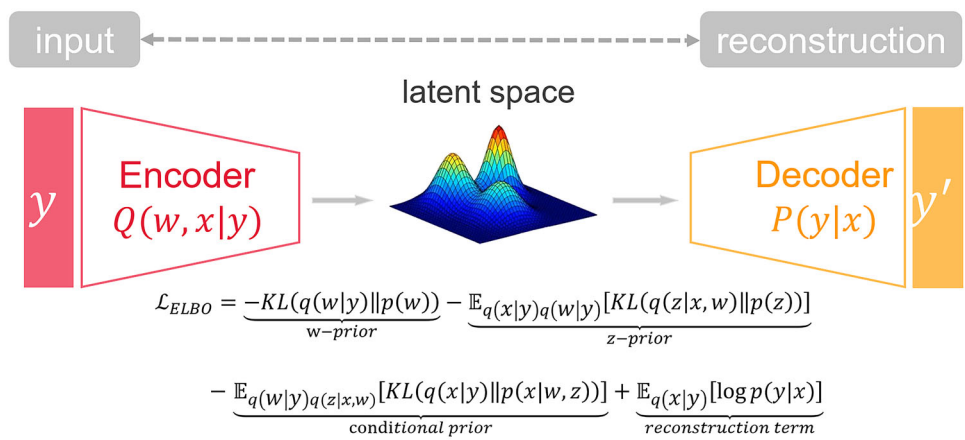
[32], and InfoVAE [33], adopted different regularization approaches, including mutual information reweighting and Hilbert–Schmidt independence criterion (HSIC) [34], to encourage disentanglement and independence between latent variables. Drawing inspiration from the work of Esmaeili et al. [35], who employed a factorized decomposition to encourage independence between groups of latent variables, we apply a similar approach to our loss function. While a deep hierarchy of latent stochastic variables can lead to a more expressive model, no direct connection has been established between disentangling sub-Gaussian distributions within a GMM and introducing the total correlation (TC) term. In our approach, by incorporating the TC term, we establish interdependencies among the sub-Gaussian distributions after the hierarchical decomposition. This enables the decoupling of several components within the sub-Gaussian distributions, and we add an extra regularization term to prevent posterior collapse.

Another challenging issue in the latent space of Gaussian mixture models is the overlapping and hard-to-classify nature of different sub-distributions. Existing loss functions are insufficient to address this problem effectively. Geometrically, minimizing the variances of sub-Gaussian distributions and maximizing the distances between different sub-Gaussian distributions can effectively tackle this issue, aligning with the principles of Fisher discriminant analysis. The geometric interpretation and optimization framework of Fisher distance have been extensively studied by experts in the field [36]. Building upon this, we establish a theoretical relationship between Fisher distance and Gaussian mixture models. By introducing the Fisher term, our aim is to constrain the distances between samples, thereby maximizing within-class differences and minimizing between-class distances. Through comprehensive experiments and an ablation study, we demonstrate the effectiveness of incorporating the Fisher term.

In summary, our contributions are as follows:

- Our first contribution is to utilize a more powerful representation model, the Gaussian mixture model (GMM), for fitting the ground truth distribution, and derives ELBO from the Bayesian equation. We enhance the expressiveness of the latent space by constructing a one-sample-one-GM approach, in contrast to the one-sample-one-standard Gaussian distribution in the vanilla VAE. However, validated by experiments, our model is more effective. Furthermore, we also found that the distribution of the coefficient vectors depends on the dimensionality of the latent variables, which results in different distributions of coefficient vectors needing to be chosen for different datasets as well as tasks.
- Our second contribution is to introduce the decoupling of the total correlation (TC) term into the Gaussian mix-

Fig. 1 Variational autoencoder network architecture with Gaussian mixture prior

$$\mathcal{L}_{ELBO} = \underbrace{-KL(q(w|y)\|p(w))}_{w-prior} - \underbrace{\mathbb{E}_{q(x|y)q(w|y)}[KL(q(z|x,w)\|p(z))]}_{z-prior}$$

$$- \underbrace{\mathbb{E}_{q(w|y)q(z|x,w)}[KL(q(x|y)\|p(x|w,z))]}_{conditional\ prior} + \underbrace{\mathbb{E}_{q(x|y)}[\log p(y|x)]}_{reconstruction\ term}$$

ture model, which results in the decoupling of individual Gaussian components. We apply the total correlation term to Gaussian mixture distributions, enabling the decoupling of individual sub-Gaussian distributions. In the case of complex latent variables, such as 2 or higher for the dimension of the latent variables, this technique can also be used for hierarchical disentanglement to achieve improved fidelity and diversity.

- Our third contribution is to address the challenge of hard-to-classify samples. We use the Fisher discriminant as a regularization term. This method helps to minimize within-class distance and maximize between-class distance, which improves clustering quality.

Our experiments and ablation study with various datasets demonstrate the model's improved performance.

Fig. 2 Probabilistic graphic model for the Gaussian mixture variational autoencoder (GMVAE) showing the generative model (left) and the variational family (right)

2 Theory and methods

2.1 Gaussian mixture prior

A Gaussian mixture model (GMM) can be seen as a combination of T individual Gaussian models, providing enhanced expressive capabilities by leveraging various probability distributions. Let $z = (z_1, z_2, \ldots, z_T)$ denote the set of sub-Gaussian distributions, where $z_i \sim \mathcal{N}(\mu_i, \sigma_i)$. The weighting factor for each sub-Gaussian distribution is $w = (w_1, w_2, \ldots, w_T)$, where $w_i \in \mathbb{R}$. The calculation method for the latent variable x is as follows:

$$x = \sum_{i=1}^{T} w_i z_i = w^T z \tag{1}$$

At this moment, the latent variable follows a Gaussian mixture distribution.

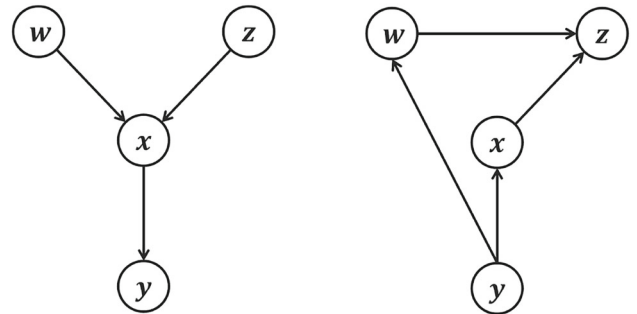In standard VAE, the posterior distribution is combined with a parameter-free isotropic Gaussian prior. The training process involves optimizing two losses simultaneously: the KL divergence and the reconstruction loss. However, calculating the KL divergence between two Gaussian mixture models poses a significant challenge.

Our modeling approach differs from that of Nat et al. [18]. In their experiments, the global data sample is modeled as a Gaussian mixture model, with individual samples belonging to one of the sub-Gaussian distribution spaces. However, their modeling approach is inaccurate, as individual samples still follow a certain Gaussian distribution. In contrast, our model calculates the corresponding K sub-Gaussian distributions and their coefficients from a single sample, yielding a weighted Gaussian mixture distribution. Theoretically, employing more complex modeling techniques leads to improved representation, and the generated data align more closely with real data. Our experiments demonstrate that the images generated by our proposed method are clearer and more distinguishable than those generated by other models.

The generation and inference processes of the GMVAE generative model, as depicted in Fig. 1, are trained using the variational inference objective, specifically the evidence

lower bound (ELBO), expressed as follows:

$$\mathcal{L}_{\text{ELBO}} = \mathbb{E}_q \left[\log \frac{p(\boldsymbol{y}, \boldsymbol{x}, \boldsymbol{w}, \boldsymbol{z})}{q(\boldsymbol{x}, \boldsymbol{w}, \boldsymbol{z}|\boldsymbol{y})} \right]. \tag{2}$$

where generative model $p(\boldsymbol{y}, \boldsymbol{x}, \boldsymbol{w}, \boldsymbol{z}) = p(\boldsymbol{w}) p(\boldsymbol{z}) p(\boldsymbol{x}|\boldsymbol{w}, \boldsymbol{z})$ $p(\boldsymbol{y}|\boldsymbol{x})$, cognition model $q(\boldsymbol{x}, \boldsymbol{w}, \boldsymbol{z}|\boldsymbol{y}) = q(\boldsymbol{x}|\boldsymbol{y}) q(\boldsymbol{w}|\boldsymbol{y})$ $q(\boldsymbol{z}|\boldsymbol{x}, \boldsymbol{w})$.

Considering the factorization of the probabilistic graphic model and the nature of logarithmic computation, the ELBO of the GMVAE-generated model can be decomposed as:

$$\begin{aligned}
\mathcal{L}_{\text{ELBO}} &= \int q(\boldsymbol{x}, \boldsymbol{w}, \boldsymbol{z}|\boldsymbol{y}) \log \frac{p(\boldsymbol{w})}{q(\boldsymbol{w}|\boldsymbol{y})} \cdot \frac{p(\boldsymbol{z})}{q(\boldsymbol{z}|\boldsymbol{x}, \boldsymbol{w})} \\
&\quad \cdot \frac{p(\boldsymbol{x}|\boldsymbol{w}, \boldsymbol{z})}{q(\boldsymbol{x}|\boldsymbol{y})} \cdot p(\boldsymbol{y}|\boldsymbol{x}) \mathrm{d}q(\boldsymbol{x}, \boldsymbol{w}, \boldsymbol{z}|\boldsymbol{y}) \\
&= - \underbrace{KL(q(\boldsymbol{w}|\boldsymbol{y})||p(\boldsymbol{w}))}_{w-\text{prior}} \\
&\quad - \underbrace{\mathbb{E}_{q(\boldsymbol{x}|\boldsymbol{y}) q(\boldsymbol{w}|\boldsymbol{y})}[KL(q(\boldsymbol{z}|\boldsymbol{x}, \boldsymbol{w})||p(\boldsymbol{z}))]}_{z-\text{prior}} \\
&\quad - \underbrace{\mathbb{E}_{q(\boldsymbol{w}|\boldsymbol{y}) q(\boldsymbol{z}|\boldsymbol{x}, \boldsymbol{w})}[KL(q(\boldsymbol{x}|\boldsymbol{y})||p(\boldsymbol{x}|\boldsymbol{w}, \boldsymbol{z}))]}_{\text{conditional prior}} \\
&\quad + \underbrace{\mathbb{E}_{q(\boldsymbol{x}|\boldsymbol{y})}[\log p(\boldsymbol{y}|\boldsymbol{x})]}_{\text{reconstruction term}} \tag{3}
\end{aligned}$$

Subsequently, we can identify four sub-terms within the ELBO: w-prior, z-prior, conditional prior, and reconstruction term. The w-prior and z-prior terms impose constraints on the sub-Gaussian distributions and their corresponding coefficients, respectively. These terms aim to align the sub-Gaussian distributions as closely as possible with the standard Gaussian distribution, thereby bringing the Gaussian mixture model closer to the true underlying distribution in the latent space. The conditional prior term ensures that the distribution obtained by sampling from the ground truth aligns as closely as possible with the distribution obtained by sampling from the latent space. Lastly, the reconstruction term evaluates the faithfulness of the model by measuring the proximity of the generated data to the ground truth data. The objective is to generate data that closely resembles the real data, thus enhancing the fidelity of the generative process.

1. **W-prior :** The weight coefficients of the sub-Gaussian obey different distributions, and the w-prior is calculated differently.

   - Assuming that w follows a Gaussian distribution, the w-prior term is expressed as KL divergence, and the model is denoted as **HGMVAE-G**.
   - Assuming that w is uniformly distributed, the degeneracy of the w-prior term is the information entropy, and the model is denoted as **HGMVAE-U**.

2. **Z-prior :** Unlike GMVAE [18], our approach decomposes the z-prior by introducing a total correlation term. It makes each sub-Gaussian distribution be independent from others, decouple from the latent space, and has stronger controllable generative ability. See the next section for decomposition in detail.

3. **Conditional prior :** Conditional prior restricts that the latent variables computed from the samples are similar to those obtained by sampling from the mixed Gaussian distribution. In this paper, the KL divergence of the mixed Gaussian model can be expressed as a weighted sum of the KL divergence of the sub-Gaussian distribution, expressed by the formula:

$$\begin{aligned}
&\mathbb{E}_{q(\boldsymbol{w}|\boldsymbol{y}) q(\boldsymbol{z}|\boldsymbol{x}, \boldsymbol{w})}[KL(q(\boldsymbol{x}|\boldsymbol{y})||p(\boldsymbol{x}|\boldsymbol{w}, \boldsymbol{z}))] \\
&= \sum_i \sum_j w_i \hat{w}_j KL(\mathcal{N}(\boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i), \mathcal{N}(\hat{\boldsymbol{\mu}}_j, \hat{\boldsymbol{\Sigma}}_j)) \tag{4}
\end{aligned}$$

$\boldsymbol{\mu}_i$, $\boldsymbol{\Sigma}_i$ denotes the mean and variance calculated from the sample, and w_i denotes the mixing coefficient of the sub-Gaussian distribution in the mixed Gaussian model.

4. **Reconstruction term :** The computation of the reconstruction term differs depending on the appliance domain. If the downstream task is to generate the data of 0-1 black and white image, the reconstruction term can use the binary cross-entropy loss function. If the generated image is a grayscale or color image, the reconstruction term can use the mean square error (MSE) loss function.

2.2 Methods of disentanglement

In order to make the distributions in GMVAE and their variables disentangleable, we introduce the total correlation (TC) term, which is inspired by hierarchically factorized VAE [35]. For z-prior term,

$$\begin{aligned}
&- KL(q(\boldsymbol{z}|\boldsymbol{x}, \boldsymbol{w})||p(\boldsymbol{z})) \\
&= -\mathbb{E}_{q(\boldsymbol{z}|\boldsymbol{x}, \boldsymbol{w})} \frac{q(\boldsymbol{z}|\boldsymbol{x}, \boldsymbol{w})}{p(\boldsymbol{z})} \\
&= -\mathbb{E}_{q(\boldsymbol{z}|\boldsymbol{x}, \boldsymbol{w})} \left[\log \frac{q(\boldsymbol{z}|\boldsymbol{x}, \boldsymbol{w})}{\prod_k q(z_k|\boldsymbol{x}, \boldsymbol{w})} + \log \frac{\prod_k q(x_k|y)}{\prod_k p(z_k)} \right. \\
&\quad \left. + \log \frac{\prod_k p(z_k)}{p(\boldsymbol{z})} \right] \\
&= \mathbb{E}_{q(\boldsymbol{z}|\boldsymbol{x}, \boldsymbol{w})} \left[\underbrace{\log \frac{p(\boldsymbol{z})}{\prod_k p(z_k)} - \log \frac{q(\boldsymbol{z}|\boldsymbol{x}, \boldsymbol{w})}{\prod_k q(z_k|\boldsymbol{x}, \boldsymbol{w})}}_{A} \right] \\
&\quad - \underbrace{\sum_k KL(q(z_k|\boldsymbol{x}, \boldsymbol{w})|p(z_k))}_{B} \tag{5}
\end{aligned}$$

In the above equation, $z = (z_1, \ldots, z_k)$, where z_i denotes the sub-latent variables sampled from the sub-Gaussian distribution, and z denotes the matrix consisting of the sub-latent variables.

we can decompose it into two sub-components A and B. Term A matches the total correlation between variables in the inference model relative to the total correlation in the generative model. The total correlation can be calculated by the following equation:

$$TC(z) = \mathbb{E}_{q_\phi(z)} \left[\log \frac{q_\phi(z)}{\prod_k q_\phi(z_k)} \right]$$
$$= \text{KL} \left(q_\phi(z) \| \prod_k q_\phi(z_k) \right) \tag{6}$$

which introduces disentanglement mechanism naturally. Term B minimizes the KL divergence between the inference marginal and prior marginal for each distribution of GMM z_k, which is formally identical to Eq. 5.

In cases with complex latent variables, such as when the dimension of the latent variables is 2, the variable of distribution z_k contains sub-variables $z_{k,i}$, which means $z_k = (z_{k,1}, \ldots, z_{k,d})$, and we can recursively decompose the KL on the marginals z_k.

$$- KL \left(q(z_k|\boldsymbol{x}, \boldsymbol{w}) | p(z_k) \right)$$
$$= \mathbb{E}_{q(z|\boldsymbol{x},\boldsymbol{w})} \left[\underbrace{\log \frac{p(z_k)}{\prod_k p(z_{k,d})} - \log \frac{q(z_k|\boldsymbol{x}, \boldsymbol{w})}{\prod_d q(z_{k,d}|\boldsymbol{x}, \boldsymbol{w})}}_{C} \right]$$
$$\underbrace{- \sum_d KL \left(q(z_{k,d}|\boldsymbol{x}, \boldsymbol{w}) | p(z_{k,d}) \right)}_{D} \tag{7}$$

Although Hierarchical KL decomposition has already appeared in hierarchically factorized VAE [35], our use case is not quite the same. Equation 5 makes the individual sub-Gaussian distributions of the mixed Gaussian model statistically independent of each other by introducing a TC term, and Eq. 7 makes the individual components of the sub-Gaussian distributions independent from each other by introducing a total correlation term. If $z_{k,d}$ is sufficiently complex, which means $z_{k,d} = (z_{k,d,1} \ldots z_{k,d,e})$, we can still continue the hierarchical decomposition similar to hierarchically factorized VAE [35]. But this operation imposes a greater computational cost.

2.3 Fisher term for regularization

In Nat's experiment [18], each value of $\boldsymbol{w}$ corresponds to a specific style of the digit, indicating that different sub-Gaussian distributions control different styles. To ensure that each feature is as independent as possible during sampling, it is desirable for samples of the same style to be close to each other and samples of different styles to be far away from each other. This implies that the within-class distance variance should be minimized for samples within the same sub-Gaussian distribution, while the between-class distance should be maximized.

Consequently, the objective becomes one of minimizing the between-class distance and maximizing the within-class distance, aligning with the principles of Fisher discriminant analysis. Building upon this idea, we adopt a latent space consisting of K classes, corresponding to K sub-Gaussian distributions in this paper. Each sub-Gaussian distribution follows $\mathcal{N}(w_i \boldsymbol{\mu}_i, w_i^2 \boldsymbol{\Sigma}_i)$, where w_i represents the mixture weight of each sub-Gaussian distribution, and $\boldsymbol{\mu}_i$ and $\boldsymbol{\Sigma}_i$ denote the mean and variance of the sub-Gaussian distribution, respectively.

Let n_i be the number of samples sampled from each sub-Gaussian distribution, denoted as $z_{i,j}$ for the i-th class and the j-th sample. By constructing the samples set $D = \{z_{i,j}\}$, where the total number of samples is $N = \sum_k n_i$, we can proceed to define the between-class covariance matrix $\boldsymbol{S}_B$ and the within-class covariance matrix $\boldsymbol{S}_W$.

First, within-class distance $\boldsymbol{S}_k \in \mathbb{R}$ is defined as:

$$\boldsymbol{S}_k = \sum_{n_i} (z_i - w_i \boldsymbol{\mu}_i)(z_i - w_i \boldsymbol{\mu}_i)^T$$
$$= n_i w_i^2 \boldsymbol{\Sigma}_i \tag{8}$$

Within-class covariance matrix $\boldsymbol{S}_W$ is defined as the sum of the covariance matrices of each class:

$$\boldsymbol{S}_W = \sum_k \boldsymbol{S}_k \tag{9}$$

Thus, the definition of the between-class covariance matrix $\boldsymbol{S}_B$ is obtained as:

$$\boldsymbol{S}_B = \sum_{k=1}^{K} n_k (w_k \boldsymbol{\mu}_k - \boldsymbol{m})(w_k \boldsymbol{\mu}_k - \boldsymbol{m})^T \tag{10}$$

In the training process of this paper, the global mean of the data after processing is 0 using normalization. In the implementation, a weak assumption is introduced: the global mean vector $\boldsymbol{m} = \boldsymbol{0}$. Then, the between-class covariance matrix $\boldsymbol{S}_B$ can be written as:

$$\boldsymbol{S}_B = \sum_{k=1}^{K} n_k w_k^2 \boldsymbol{\mu}_k \boldsymbol{\mu}_k^T \tag{11}$$

We want to maximize the between-class variance and minimize the within-class variance, so we can define the Fisher

regularization term F_{reg} as:

$$tr(S_W^{-1} S_B) = tr\left(\left(\sum_{k=1}^{K} n_k w_k^2 \Sigma_i\right)^{-1}\left(\sum_{k=1}^{K} n_k w_k^2 \mu_k \mu_k^T\right)\right) \quad (12)$$

Assuming that the number of samples sampled from each sub-Gaussian distribution is the same, i.e., $n_1 = n_2 = \cdots = n_k$. The calculation of the Fisher regularization term could be simplified as follows:

$$F_{\text{reg}} = tr\left(\left(\sum_{k=1}^{K} w_k^2 \Sigma_i\right)^{-1}\left(\sum_{k=1}^{K} w_k^2 \mu_k \mu_k^T\right)\right) \quad (13)$$

So, the total loss can be written as:

$$\mathcal{L} = \underbrace{-KL(q(\boldsymbol{w} \mid y) \| p(\boldsymbol{w}))}_{\boldsymbol{w} - \text{prior}}$$
$$- \underbrace{\mathbb{E}_{q(x|y)q(\boldsymbol{w}|y)}[KL(q(z \mid x, \boldsymbol{w}) \| p(z))]}_{z - \text{prior}}$$
$$- \mathbb{E}_{q(x|y)} \underbrace{\left[\log \frac{p(x \mid \boldsymbol{w}, z)}{\prod_k p(x_k \mid \boldsymbol{w}, z)} - \log \frac{q(x \mid y)}{\prod_k q(x_k \mid y)}\right]}_{A}$$
$$- \sum_k \underbrace{KL(q(x_k \mid y) \| p(x_k \mid \boldsymbol{w}, z))}_{B}$$
$$+ \underbrace{\mathbb{E}_{q(x|y)}[\log p(y \mid x)]}_{\text{reconstruction term}} + F_{reg} \quad (14)$$

3 Experiments and evaluations

In this section, we validate the effectiveness of our HGMVAE model on several downstream clustering (3.1), classification (3.2) and generation (3.3) tasks. The conventional VAE typically employs fully connected neural networks to compute the latent variables, which can result in over-fitting and a larger number of data parameters. To address this, we utilize convolutional neural networks (CNNs) in our network architecture. Our CNN model consists of five convolutional layers with a kernel size of 3×3, followed by two fully connected layers. Notably, we exclude fully connected neural networks and pooling layers in order to retain the essential information of the data. The network is trained using stochastic gradient descent (SGD) optimization, minimizing the KL divergence cost, and initialized with the network parameters from the VAE. Despite the simplicity of our model, it demonstrates excellent performance on the datasets used in this paper. To ensure reliable results, all experiments were conducted 10 times with the same network structure, and the quantitative experimental results were obtained by averaging the outcomes.

3.1 Clustering results

3.1.1 Setup

For our clustering experiments, we primarily utilize the **MNIST** [37] dataset. We evaluate the performance using three metrics: Silhouette Coefficient (**SC**) [38], Calinski Harabasz Index (**CH**) [39], and Davies Bouldin Index (**DB**) [40]. The SC measures the similarity between samples within the same category and the dissimilarity between samples of different categories. A value closer to 1 indicates high similarity within categories and significant dissimilarity between categories. The CH Index assesses the clustering quality based on the within-class covariance (within-cluster variance) and between-class covariance (between-cluster variance). A higher value signifies smaller within-class covariance, larger between-class covariance, and better clustering performance. The DB Index evaluates the clustering by considering both the within-class distance (within-cluster distance) and between-class distance (between-cluster distance). A smaller value indicates smaller within-class distances and larger between-class distances, reflecting improved clustering results.

3.1.2 Visualization of learned embeddings

We compare our model of different z-prior with GMVAE [18]. The results are presented in Table 1, where the best-performing model is indicated in bold. Across different latent space dimensions, the proposed model in this paper outperforms the GMVAE model.

In analyzing unsupervised clustering, the behavior of different models across varying latent dimensions can be seen in Fig. 3. Results indicate that clustering with weight coefficients following a Gaussian distribution outperforms when latent variable dimensions are less than 16, while clustering with weight coefficients following the uniform distribution is better for dimensions greater than or equal to 16. This can be explained by the fact that in lower dimensionalities, encoding processes lose more information for different latent variables, creating different importance levels. Conversely, in higher dimensionalities, the information contained in latent variables is relatively consistent, resulting in similar importance levels and learned weight distributions that obey the uniform distribution.

The clustering performance of models with various dimensions on the MNIST dataset is displayed in Fig. 3. The figure reveals that dimension 8 yields the most favorable clustering outcomes. In terms of information encoding, a latent variable with too short dimension results in a loss of information and worsens clustering performance. On the contrary, a latent variable with too long dimension introduces excessive noise and also deteriorates clustering performance.

Table 1 The clustering results on the MNIST dataset

Metrics	Model	L-dim.						
		2	4	8	16	32	64	128
SC↑	GMVAE	0.143	0.382	0.404	0.413	0.290	0.213	0.172
	HGMVAE-G	**0.259**	**0.467**	**0.442**	0.415	0.289	0.178	0.190
	HGMVAE-U	0.257	0.408	0.427	**0.433**	**0.363**	**0.292**	**0.271**
CH↑	GMVAE	1154.896	2958.787	3428.849	3386.021	1893.010	1175.965	1132.841
	HGMVAE-G	1967.923	**3782.171**	**3930.585**	3173.546	2039.884	1224.031	1361.923
	HGMVAE-U	**1987.279**	3290.893	3775.829	**3736.877**	**2693.011**	**2227.500**	**1931.504**
DB↓	GMVAE	6.556	1.026	0.956	0.868	1.646	3.735	4.068
	HGMVAE-G	**3.768**	**0.767**	**0.853**	0.996	1.823	3.452	2.749
	HGMVAE-U	5.021	0.939	0.893	**0.868**	**1.333**	**1.810**	**1.493**

L-dim. denotes the dimension of latent space

Bold indicates best, bolditalic indicates second best, italic indicates third best
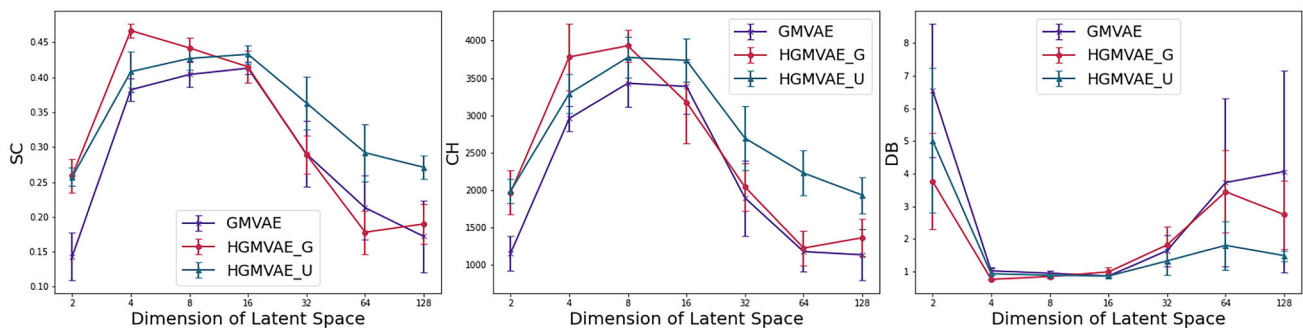

Fig. 3 Experimental results of GMVAE, HGMVAE-G, and HGMVAE-U for clustering in latent dimensions of 2, 4, 8, 16, 32, 64, and 128, respectively. From left to right are Silhouette Coefficient (SC), Calinski Harabasz Index (CH), and Davies Bouldin Index (DB), respectively

Conducting comparative experiments on the dataset can help identify the optimal latent variable dimensions. To facilitate a clearer observation of the clustering effect in latent space, we present a visualization of the clusters on MNIST in Fig. 4.

3.2 Classification results

Based on our clustering task, we find that clustering is most effective when using 8 dimensions for the latent variables. In our experimentation with the CIFAR10 and MINIST datasets, we train for 10 epochs using 8 dimensions for the latent variables, a learning rate of 0.001, and the AdamW optimizer. We use SVM as our classifier and then compare the impact of the latent variables obtained from different VAE variants in the classification task.

The evaluation metric employed in this paper is classification accuracy. On the CIFAR10 test dataset, HGMVAE_G exhibited an improvement of 1.8% and 3.6% over VAE and GMVAE, respectively. On the MNIST test dataset, both HGMVAE_G and HGMVAE_U outperformed other models in accuracy.

3.3 Generation results

3.3.1 Setup

The most important metric for evaluating the generated task is to calculate the similarity between the generated image and the original image. In this paper, we use four different evaluation metrics. we include four metrics: Fréchet Inception Distance (**FID**), Structural Similarity (**SSIM**), Multi-Scale Structural Similarity (**MS-SSIM**), and Learned Perceptual Image Patch Similarity (**LPIPS**). FID is primarily used to measure the difference between generated images and real images. SSIM and MS-SSIM are used to measure the structural similarity between two images. SSIM is a single-scale metric, while MS-SSIM considers multiple scales. LPIPS is a deep learning-based metric for assessing the perceptual difference between images. It is used to evaluate the perceptual quality of images. We validate on four datasets: 3D Chair [42], CelebA [43], MNIST [37], and Fashion MNIST [44].

Table 3 shows the generation performance obtained by these baselines; in most cases, our model is the best.
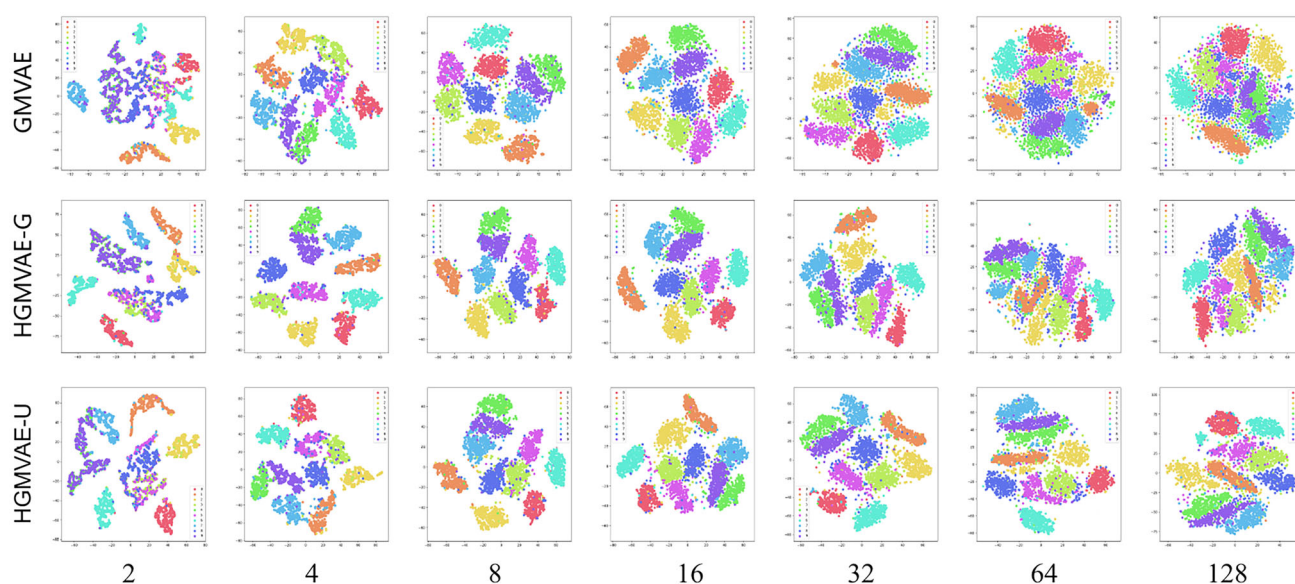
Fig. 4 Latent space of 10-class dataset with full labels projected by t-SNE [41]

Table 2 The results of classification task on MNIST and CIFAR10 datasets

	VAE	Beta-VAE	GMVAE	SWAE	WAE_MMD	WAE_RBF	HGMVAE_G	HGMVAE_U
CIFAR10	0.957	0.959	0.961	0.883	0.896	0.873	**0.975**	0.965
MINIST	0.978	0.964	0.985	0.983	0.98	0.979	**0.996**	0.981

Bold indicates best, bolditalic indicates second best, italic indicates third best

3.3.2 Visualization results

The results are obtained by training 10 epochs with the dimensions of the latent variables chosen as 128, the learning rate chosen as 0.001, and the optimizer chosen as AdamW. Figure 5a shows the fidelity of every two rows. Figure 5b shows gradual change in 2-D latent space, including changes of gender, hair color, hair length, background color, smile angle, and face orientation.

In order to validate whether a generative model learns disentangled representations, we test its ability to recognize independent components underlying the data. In digit dataset (Fig. 6a), it represents as keeping content unchanged and varying angle, handwritten stroke, width, and thickness of digits. In CelebA (Fig. 6b), it characterized by transformations of size, style of legs or back, material, azimuth, etc.

3.4 Ablation study

We conducted ablation experiments on the clustering and generation tasks using the proposed model in this paper. Our experiments compared different w-prior terms and Fisher regularization terms to determine their impact on performance.

Based on the results presented in Tables 1, 2, and 3, we find that using a more robust Gaussian mixture model for modeling the hidden space can lead to superior performance. However, different distributions of the prior terms also have different effects on performance. Specifically, our results in Tables 4 and 5 show that the one-sample-one-GMM modeling approach outperforms the approach using the Gaussian distribution.

Furthermore, we investigate the impact of the Fisher regularization term on our experiments. We find that incorporating this term makes sub-Gaussian distributions more independent, improving clustering performance. On the generation task, our model without the regularization term has a similar performance as GMVAE, but incorporating the regularization term results in substantial improvements in metrics. For example, on the 3D Chair dataset, HGMVAE-G w/o F_{reg} shows a 59.84% improvement in FID, HGMVAE-U w/o F_{reg} shows a 65.76% improvement in FID compared to HGMVAE-G, and HGMVAE-U w/o F_{reg} shows an increase in FID.

Table 3 Reconstruction performance comparison

		VAE	beta-VAE	GMVAE	SWAE	WAE_IMQ	WAE_RBF	HGMVAE-G	HGMVAE-U
3D Chair	FID↓	154.098	144.263	**103.343**	106.848	196.952	106.727	*92.201*	*88.605*
	SSIM↑	0.903	0.926	*0.968*	0.963	0.872	**0.966**	0.717	*0.975*
	MS-SSIM↑	0.933	0.957	*0.986*	0.982	0.887	**0.986**	0.932	*0.991*
	LPIPS↓	0.125	0.099	**0.042**	0.046	0.166	*0.040*	0.122	*0.030*
CelebA	FID↓	115.438	117.074	121.693	98.670	134.997	**92.816**	**92.201**	*86.353*
	SSIM↑	0.546	0.530	0.568	*0.758*	0.458	**0.766**	0.717	**0.757**
	MS-SSIM↑	0.834	0.821	0.849	**0.947**	0.764	**0.951**	0.932	*0.949*
	LPIPS↓	0.194	0.202	0.185	**0.099**	0.243	*0.101*	0.122	**0.102**
MNIST	**FID↓**	50.465	61.032	43.089	*32.374*	**36.731**	39.790	37.973	*30.958*
	SSIM↑	0.834	0.774	0.941	**0.965**	*0.966*	0.962	0.930	*0.966*
	MS-SSIM↑	0.959	0.930	0.983	*0.991*	*0.992*	0.990	0.986	*0.991*
	LPIPS↓	0.119	0.142	0.054	*0.021*	**0.031**	0.037	0.084	**0.029**
Fashion MNIST	FID↓	104.231	111.043	101.992	95.150	96.954	**93.800**	*86.890*	*79.415*
	SSIM↑	0.706	0.687	0.752	*0.813*	**0.795**	0.790	0.766	*0.816*
	MS-SSIM↑	0.847	0.827	0.884	*0.921*	0.917	**0.918**	0.912	*0.934*
	LPIPS↓	0.113	0.122	0.104	*0.071*	**0.087**	0.091	0.091	*0.069*

Bold indicates best, bolditalic indicates second best, italic indicates third best

Fig. 5 **a** Image reconstructions on MNIST. Every two lines represents a reconstruction, the original image is above, while the generated image from ours is below. **b** Latent manifold on CelebA. Give four images at corners to generate a transformation process between them

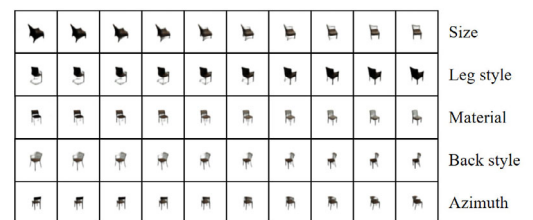(a) Image reconstructions on MNIST
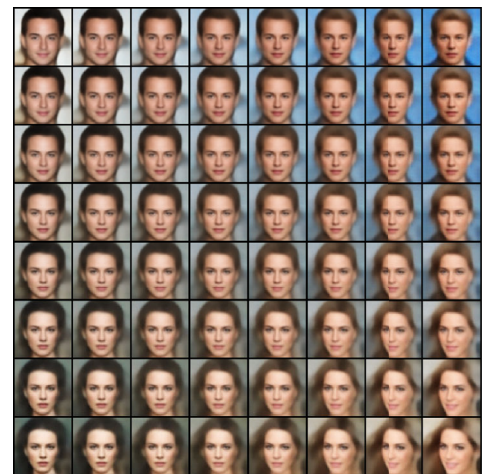
(b) Learned manifold on CelebA

Fig. 6 Latent traversals on MNIST and 3D Chair

(a) Latent traversals on MNIST

(b) Latent traversals on 3D Chair

Table 4 Ablation experiment on clustering

Metric	Model	L-dim.						
		2	4	8	16	32	64	128
SC↑	GMVAE	0.143	0.382	0.404	0.413	0.29	0.213	0.172
	HGMVAE-G	**0.259**	**0.467**	**0.442**	0.415	0.289	0.178	0.19
	w/o F_{reg}	−0.051	0.289	0.361	0.38	0.289	0.281	0.266
	HGMVAE-U	0.257	0.408	0.427	**0.433**	**0.363**	**0.292**	**0.271**
	w/o F_{reg}	−0.003	0.282	0.36	0.409	0.312	0.187	0.192
CH↑	GMVAE	1154.896	2958.787	3428.849	3386.021	1893.01	1175.965	1132.841
	HGMVAE-G	**1967.923**	**3782.171**	**3930.585**	3173.546	2039.884	1224.031	1361.923
	w/o F_{reg}	325.915	1782.553	3167.959	2748.083	2308.934	2065.936	1906.594
	HGMVAE-U	1987.279	3290.893	3775.829	**3736.877**	**2693.011**	**2227.5**	**1931.504**
	w/o F_{reg}	582.317	2134.179	2792.347	3241.303	1961.919	1461.447	1321.775
DB↓	GMVAE	6.556	1.026	0.956	0.868	1.646	3.735	4.068
	HGMVAE-G	**3.768**	**0.767**	**0.853**	0.996	1.823	3.452	2.749
	w/o F_{reg}	7.558	3.913	1.139	1.629	2.108	3.132	3.639
	HGMVAE-U	5.021	0.939	0.893	**0.868**	**1.333**	**1.81**	**1.493**
	w/o F_{reg}	6.835	2.573	1.33	1.09	3.864	4.49	4.255

A comparison of HGMVAE of two kinds of w-prior (Gaussian and uniform distribution) with and without (w/o) Fisher regularization in clustering on four datasets. The bold indicated the best results
Bold indicates best, bolditalic indicates second best, italic indicates third best

Table 5 Ablation experiment on generation

Dataset	Metric	GMVAE	HGMVAE-G	HGMVAE-U	G w/oF_{reg}	U w/o F_{reg}
3D Chair	FID↓	103.343	40.986	**37.778**	103.076	110.321
	SSI↑	0.968	0.966	**0.975**	0.966	0.956
	MS-SSIM↑	0.986	0.985	**0.991**	0.985	0.98
	LPIPS↓	0.042	0.037	**0.03**	0.04	0.052
CelebA	FID↓	121.693	52.663	**44.545**	99.977	91.225
	SSIM↑	0.568	0.717	**0.757**	0.709	0.739
	MS-SSIM↑	0.849	0.932	**0.949**	0.929	0.943
	LPIPS↓	0.185	0.122	**0.102**	0.127	0.107
MNIST	FID↓	43.089	37.973	**30.958**	39.36	41.697
	SSIM↑	0.941	0.93	**0.966**	0.91	0.9
	MS-SSIM↑	0.983	0.986	**0.991**	0.987	0.986
	LPIPS↓	0.054	0.084	**0.029**	0.087	0.087
Fashion MNIST	FID↓	101.992	86.89	**79.415**	95.707	94.828
	SSIM↑	0.752	0.766	**0.816**	0.718	0.763
	MS-SSIM↑	0.884	0.912	**0.934**	0.893	0.911
	LPIPS↓	0.104	0.091	**0.069**	0.107	0.1

A comparison of HGMVAE of two kinds of w-prior (Gaussian and uniform distribution) with and without (w/o) Fisher regularization in generation on four datasets. The bold indicated the best results
Bold indicates best, bolditalic indicates second best, italic indicates third best

4 Conclusion

In this paper, we introduce the hierarchical disentanglement in Gaussian mixture variational autoencoder (HGMVAE) as a novel approach for disentangled representation learning tasks. HGMVAE combines the learning of Gaussian mixture latent spaces and the hierarchical disentanglement of feature and label embeddings. Not only does HGMVAE achieve better performance, but it also provides insights into unsupervised clustering and model interpretability. However, it is important to acknowledge that the modeling of the latent space as a Gaussian mixture model and the hierarchical disentanglement of the variational lower bound lead to increased computational costs compared to standard VAEs. Despite this limitation, the benefits and advancements brought about by HGMVAE outweigh these challenges.

Data availability The authors declare that there are no conflicts of interest; we do not have any possible conflicts of interest. The authors confirm that the data supporting the findings of this study are available within the article.

Declarations

Conflict of interest The authors declare that there are no conflicts of interest, we do not have any possible conflicts of interest. The authors confirm that the data supporting the findings of this study are available within the article.

References

1. Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., Bengio, Y.: Generative adversarial nets. Neural Inf. Process. Syst. **35**, 53–65 (2014)
2. Radford, A., Metz, L., Chintala, S.: Unsupervised representation learning with deep convolutional generative adversarial networks. arXiv:1511.06434 (2015)
3. Wang, X., Gupta, A.: Generative image modeling using style and structure adversarial networks. In: Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part IV 14, pp. 318–335 (2016). Springer
4. Karras, T., Laine, S., Aila, T.: A style-based generator architecture for generative adversarial networks. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 4401–4410 (2019)
5. Kingma, D.P., Welling, M.: Auto-encoding variational bayes. arXiv:1312.6114 (2013)
6. Sohn, K., Lee, H., Yan, X.: Learning structured output representation using deep conditional generative models. Adv. Neural Inf. Process. Syst. **28** (2015)
7. Van Den Oord, A., Vinyals, O., et al.: Neural discrete representation learning. Adv. Neural Inf. Process. Syst. **30** (2017)
8. Van Den Oord, A., Kalchbrenner, N., Kavukcuoglu, K.: Pixel recurrent neural networks. In: International Conference on Machine Learning, pp. 1747–1756. PMLR (2016)
9. Salimans, T., Karpathy, A., Chen, X., Kingma, D.P.: Pixelcnn++: improving the pixelcnn with discretized logistic mixture likelihood and other modifications. arXiv:1701.05517 (2017)
10. Ho, J., Jain, A., Abbeel, P.: Denoising diffusion probabilistic models. Adv. Neural Inf. Process. Syst. **33**, 6840–6851 (2020)
11. Rombach, R., Blattmann, A., Lorenz, D., Esser, P., Ommer, B.: High-resolution image synthesis with latent diffusion models. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 10684–10695 (2022)
12. Dhariwal, P., Nichol, A.: Diffusion models beat gans on image synthesis. Adv. Neural Inf. Process. Syst. **34**, 8780–8794 (2021)
13. Bank, D., Koenigstein, N., Giryes, R.: Autoencoders. arXiv:2003.05991 (2020)
14. Michelucci, U.: An introduction to autoencoders. arXiv:2201.03898 (2022)
15. Reynolds, D.A., et al.: Gaussian mixture models. Encycl. Biom. **741**(659–663) (2009)
16. Nalisnick, E., Hertel, L., Smyth, P.: Approximate inference for deep latent gaussian mixtures. In: NIPS Workshop on Bayesian Deep Learning, vol. 2, p. 131 (2016)
17. Jiang, Z., Zheng, Y., Tan, H., Tang, B., Zhou, H.: Variational deep embedding: an unsupervised and generative approach to clustering. arXiv:1611.05148 (2016)
18. Dilokthanakul, N., Mediano, P.A., Garnelo, M., Lee, M.C., Salimbeni, H., Arulkumaran, K., Shanahan, M.: Deep unsupervised clustering with gaussian mixture variational autoencoders. arXiv:1611.02648 (2016)
19. Lee, D.B., Min, D., Lee, S., Hwang, S.J.: Meta-gmvae: Mixture of Gaussian vae for unsupervised meta-learning. In: International Conference on Learning Representations (2021)
20. Bai, J., Kong, S., Gomes, C.P.: Gaussian mixture variational autoencoder with contrastive learning for multi-label classification. In: International Conference on Machine Learning, pp. 1383–1398. PMLR (2022)
21. Figueroa, J.A.: Semi-supervised learning using deep generative models and auxiliary tasks. In: NeurIPS Workshop on Bayesian Deep Learning (2019)
22. Collier, M., Urdiales, H.: Scalable deep unsupervised clustering with concrete gmvaes. arXiv:1909.08994 (2019)
23. Yang, L., Cheung, N.-M., Li, J., Fang, J.: Deep clustering by gaussian mixture variational autoencoders with graph embedding. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 6440–6449 (2019)
24. Abdulaziz, A., Zhou, J., Di Fulvio, A., Altmann, Y., McLaughlin, S.: Semi-supervised gaussian mixture variational autoencoder for pulse shape discrimination. In: ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 3538–3542. IEEE (2022)
25. Bengio, Y., Courville, A., Vincent, P.: Representation learning: a review and new perspectives. IEEE Trans. Pattern Anal. Mach. Intell. **35**(8), 1798–1828 (2013)
26. Bengio, Y.: The consciousness prior. arXiv:1709.08568 (2017)
27. Higgins, I., Matthey, L., Pal, A., Burgess, C., Glorot, X., Botvinick, M., Mohamed, S., Lerchner, A.: beta-vae: learning basic visual concepts with a constrained variational framework. In: International Conference on Learning Representations (2017)
28. Tishby, N., Pereira, F.C., Bialek, W.: The information bottleneck method. arXiv preprint physics/0004057 (2000)
29. Burgess, C.P., Higgins, I., Pal, A., Matthey, L., Watters, N., Desjardins, G., Lerchner, A.: Understanding disentangling in $beta$-vae. arXiv:1804.03599 (2018)
30. Hu, M.-f, Liu, Z.-y, Liu, J.-w: mcvae: disentangling by mean constraint. Vis. Comput. **40**, 1229–1243 (2023)
31. Kim, H., Mnih, A.: Disentangling by factorising. In: International Conference on Machine Learning, pp. 2649–2658. PMLR (2018)
32. Chen, R.T., Li, X., Grosse, R.B., Duvenaud, D.K.: Isolating sources of disentanglement in variational autoencoders. Adv. Neural Inf. Process. Syst. **31** (2018)

33. Zhao, S., Song, J., Ermon, S.: Infovae: Information maximizing variational autoencoders. arXiv:1706.02262 (2017)
34. Gretton, A., Bousquet, O., Smola, A., Schölkopf, B.: Measuring statistical dependence with Hilbert–Schmidt norms. In: Algorithmic Learning Theory: 16th International Conference, ALT 2005, Singapore, October 8–11, 2005. Proceedings 16, pp. 63–77. Springer (2005)
35. Esmaeili, B., Wu, H., Jain, S., Bozkurt, A., Siddharth, N., Paige, B., Brooks, D.H., Dy, J., Meent, J.-W.: Structured disentangled representations. In: The 22nd International Conference on Artificial Intelligence and Statistics, pp. 2525–2534. PMLR (2019)
36. Vahdat, A., Kautz, J.: Nvae: a deep hierarchical variational autoencoder. Adv. Neural Inf. Process. Syst. **33**, 19667–19679 (2020)
37. LeCun, Y., Bottou, L., Bengio, Y., Haffner, P.: Gradient-based learning applied to document recognition. Proc. IEEE **86**(11), 2278–2324 (1998)
38. Rousseeuw, P.J.: Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. J. Comput. Appl. Math. **20**, 53–65 (1987)
39. Caliński, T., Harabasz, J.: A dendrite method for cluster analysis. Commun. Stat.-Theory Methods **3**(1), 1–27 (1974)
40. Davies, D.L., Bouldin, D.W.: A cluster separation measure. IEEE Trans. Pattern Anal. Mach. Intell. **2**, 224–227 (1979)
41. Maaten, L., Hinton, G.: Visualizing data using t-sne. J. Mach. Learn. Res. **9**(11) (2008)
42. Aubry, M., Maturana, D., Efros, A.A., Russell, B.C., Sivic, J.: Seeing 3d chairs: exemplar part-based 2d-3d alignment using a large dataset of cad models. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 3762–3769 (2014)
43. Liu, Z., Luo, P., Wang, X., Tang, X.: Large-scale celebfaces attributes (celeba) dataset. Retrieved August 15(2018), 11 (2018)
44. Xiao, H., Rasul, K., Vollgraf, R.: Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms. arXiv:1708.07747 (2017)

Yongqi Liu received the B.S. degree in Information and Computer Science from Wuhan Textile University, Wuhan, Hubei, China, in 2021. She is currently working toward the M.S. degree in Software Engineering with the School of Computer Science and Artificial Intelligence, Wuhan Textile University, Wuhan, Hubei, China. Her research interests include generative model, computer vision, and machine learning.

Xiaoqin Du received the M.Sc. and Ph.D. degrees from China University of Geosciences and Huazhong University of Science and Technology (Wuhan, Hubei, China), in 2000 and 2007 respectively. He is currently with Wuhan Textile University (Wuhan, Hubei, China), as a full Associate Professor at School of Computer Science and Artificial Intelligence. From Aug 31 2012 to Mar 1 2013, he held a visiting scholar with the University of Texas at Brownsville, the college of Computer and Information Sciences, Texas, USA. He has authored dozens of publications and two lecture notes on Artificial Intelligence and Machine Learning. His research interests include Machine Learning, Reinforcement Learning, Deep Learning and Artificial Intelligence for computer games. He was the recipient of the Advanced Student Award, China University of Geosciences in 1999. From Jul. 2000 to Sept. 2002, he worked as a software development engineer for GIS and 3D modeling at Wuhan Supresoft Inc. Wuhan, Hubei, China. He also received an award "the third class award of progress of science and technology of Hubei Province", NELL-DVS2000 Video Security System, Wuhan, Hubei, China. From 2003 to 2007, he researched on the algorithm of real-time task schedule in grid environment (as a participant), supported by the National Natural Science Foundation of China.

Jiashuang Zhou received the B.S. degree in Computer Science from Wuhan Textile University, Wuhan, Hubei, China, in 2021. He is currently working toward the M.S. degree in Software Engineering with the School of Computer Science and Artificial Intelligence, Wuhan Textile University, Wuhan, Hubei, China. His research interests include generative model, computer vision, and machine learning.