Journal of Computers Vol. 34 No. 6, December 2023, pp. 1-14
doi: 10.53106/199115992023123406001

# Disentangling Representation of Variational Autoencoders Based on Cloud Models

Jin Dai[1], Zhifang Zheng[2*]

[1] Chongqing Key Laboratory of Computational Intelligence, Chongqing University of Posts and Telecommunications, Chongqing 400065, China

daijin@cqupt.edu.cn

[2] School of Computer Science and Technology, Chongqing University of Posts and Telecommunications, Chongqing 400065, China

s200231020@stu.cqupt.edu.cn

*Received 17 December 2022; Revised 13 March 2023; Accepted 1 May 2023*

**Abstract.** Variational autoencoder (VAE) has the problem of uninterpretable data generation process, because the features contained in the VAE latent space are coupled with each other and no mapping from the latent space to the semantic space is established. However, most existing algorithms cannot understand the data distribution features in the latent space semantically. In this paper, we propose a cloud model-based method for disentangling semantic features in VAE latent space by adding support vector machines (SVM) to feature transformations of latent variables, and we propose to use the cloud model to measure the degree of disentangling of semantic features in the latent space. The experimental results on the CelebA dataset show that the method obtains a good disentangling effect of semantic features in the latent space, which proves the effectiveness of the method from both qualitative and quantitative aspects.

**Keywords:** variational autoencoder, disentangling representation, cloud model, transformation of features

## 1 Introduction

In recent years, deep generative models have received a lot of attention due to their large number of applications in deep learning, and variational autoencoders [1], as an unsupervised deep generative model, have shown great application in data generation. Through the continuous improvement of VAE structure by many scholars, VAE has developed various variants of VAE and the quality of generated images has been greatly improved. VAE is able to learn smooth potential representations of input data and control the distribution of the hidden variable $z$. Therefore, it has been a great success in computer vision fields such as still image generation [2], text generation [3-5], target detection [6-7], and semantic image drawing [8-9].

However, the learning process of VAE is like a "black box", in which the deep learning process mostly relies on a lot of engineering experience and skills, and it is difficult to thoroughly understand how the latent variables affect the results. The lack of interpretability severely limits its application in real-world tasks. Therefore, in order to improve the interpretability and transparency of VAE and to establish a trust relationship between the user and the model, disentangled representations of the latent variables are needed to read the internal mechanisms.

In response to the unknown internal mechanism of VAE, many scholars have also studied the interpretability of the generative model and proposed a large number of explanatory methods to help users understand the internal working mechanism of the model. however, the research on interpretability is still in its early stage and there are still a large number of scientific problems that have yet to be solved. Currently, there is no fully unified understanding of the definition, evaluation and measurement of interpretability, and different scholars view and solve the problem from different perspectives, giving different meanings to interpretability, so the proposed interpretation methods also have their own focus [10-11].

In this paper, we believe that in order to make the deep generative model reach interpretability, it is necessary to enable human to understand the interior of the model and decouple the features in the latent space from the human cognitive perspective. Humans cognize and think through natural language, and concepts are the basic units of natural language [12], and it is necessary to give semantic and conceptual meaning to the latent variable

* Corresponding Author

features in order to make the latent space of VAE transparent. And in the field of conceptual uncertainty, cloud model [13], as a two-way cognitive model, can realize the mutual transformation between concepts and data through cloud transformation, so this paper proposes to realize the feature disentangling metric with the help of cloud model.

In this paper, the Gaussian distribution of VAE is replaced with a cloud model, and the sampling space is expanded while adding constraints on reconstruction loss. A feature transformation is performed for the hidden space after sampling from the VAE so as to explain the generative logic in it. In this paper, we verify the interpretable results on the VAE hidden space through theoretical analysis and experimental results, so that the semantic variables of deconvolution can control the facial attributes, and the cloud model is used for quantitative assessment of the degree of deconvolution between different features.

The main contributions of this paper are as follows:

(1) Using the cloud model as the prior distribution of the variational autoencoder, so that more features in the latent space are sampled to increase the representation of the reconstructed data, and the generation quality of reconstructed images is improved.

(2) Construct mappings from the hidden space to the image space and the semantic space, and perform feature transformations on the variables in the hidden space for the purpose of feature decoupling. And semantic editing is performed on the pre-trained model, and the decoupling process of the hidden space is inferred backwards by the generated results.

(3) Qualitative to quantitative conversion of separated concepts with the help of cloud models, and quantitative measurement of the degree of decoupling using similarity measures of cloud models.

## 2 Related Work

### 2.1 Explainable Studies of VAE

To solve the uninterpretable problem of VAE, researchers have proposed a variety of VAE variants according to different task requirements. In terms of disentangled representations of VAE, there are three main types as follows.

(1) Based on disentangled representations with prior constraints, beta variational autoencoder ($\beta$-VAE) proposed by Higgins et al. [14] in 2017 is an unsupervised visual disentangled representation learning model, which adds an additional hyperparameter $\beta$ to the VAE objective to strengthen the independence constraint on the approximate posterior distribution, although it suffers from poor reconstruction fidelity. So Kim et al. [15] and Chen et al. [16] improved the $\beta$-VAE and successively proposed penalty terms that can directly encourage the posterior cumulative distribution $q(z)$ to obey the factorial factorial distribution in 2018. The factor variational autoencoder (Factor-VAE) proposed by Kim et al. directly adds penalty terms to the original VAE optimization function. Beta total correlation variational autoencoder ($\beta$-TCVAE) proposed by Chen et al [16] further decomposes the second term of the objective function from a theoretical derivation perspective and is used to enhance the decoupling performance of the model. In addition to the above methods of altering the Kullback–Leibler (KL) canonical term by considering it as a whole, there are also methods to constrain the KL term with a more refined derivation [17-18] to improve the disentangling ability.

(2) Disentangled representations based on structured models, i.e., the network or structure to improve. Eslami et al. [19] proposed the VAE based structured image model attend-infer-repeat (AIR) in 2016, which motivates the network to iteratively learn disentangled representations of scene objects by constructing the coded inference network as a form of recurrent neural network. Li et al. [20] proposed the latent tree variational autoencoder (LTVAE) in 2019, whose representation structure is composed of multiple hyper potential variables, which can autonomously select a subset of potential features for each hyper potential variable and learn the dependency structure among different hyper potential variables. Yang [21] et al. proposed causal variational autoencoder (Causal-VAE) in 2021, considering the relationship between changing factors in data from the perspective of causality. This method supports the generation of images with causality semantics and the creation of counterfactual results. Xu [22] et al. proposed counterfactual fairness variational autoencoder (CF-VAE) to obtain a structured representation of knowledge about the domain, enabling predictive models to achieve counterfactual fairness.

(3) Based on disentangled representations of knowledge induction preferences, Bouchacourt et al. in 2018 [23] proposed the multi-level variational autoencoder (ML-VAE), a multilevel variational autoencoder, which shares relevant factor potential representations in intra-group data and can visualize disentangled representations by swapping potential representations to generate new type images. Vowels et al. in 2020 [24] proposed the gated variational autoencoder (Gated-VAE) which enables the incorporation of prior knowledge from any available domain during the network training process, making the model more widely applicable. Xu [25] et al proposed Multi variational autoencoder (multi-VAE) to separate features from continuous views by controlling mutual information capacity, so that common outlier information is effectively excavated. Zhu [26] et al. proposed weakly supervised variational autoencoder (SW-VAE), which is a weakly supervised training method. This method uses the input observations as supervised signals, which allows the model to improve significantly on the untangling task.

However, although these algorithms can achieve effective disentangled representations of data to some extent, the learning process still lacks a clear physical semantic orientation, so this paper proposes the concept of embedding semantics in the latent space of VAE for feature disentangling.

### 2.2 Relevant Theories

This section introduces the algorithms and models involved in this paper, including the variational autoencoder and cloud model algorithms, as well as the cloud similarity metric algorithm.

**VAE.** VAE is an important class of generative models, which consists of two processes: encoding and decoding. As shown in Fig. 1, $x$ is the image data that can be observed, $z$ is the latent variable that contains the key features of $x$, The process from $z$ to $x$ is represented as the generative model $P_\theta(x \mid z)$ ; the process from $x$ to $z$ is represented as the recognition model $q_\phi(z \mid x)$ .

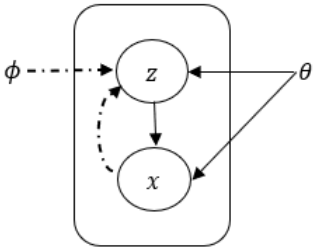



**Fig. 1.** VAE diagram model

The optimization objective function of the VAE model has two terms, the first term is the KL scatter between the variational posterior distribution $q_\phi(z \mid x)$ and the true posterior distribution $p(z \mid x)$ , and the second term is the variational lower bound of the true data $x$. After a series of variational extrapolations, the final objective of VAE is:

$$\mathcal{L}_{\text{VAEs}}(\theta,\phi,x,x') = \underset{\theta,\phi}{\operatorname{argmin}}[\frac{1}{2}\mu_\theta(z) - x^2 + \frac{1}{2}\sum_{j=1}^{dim(Z)}\sigma_\phi^2(x)_j + \mu_\phi^2(x)_j - log\sigma_\phi^2(x)_j] \ . \tag{1}$$

**Cloud Model.** The cloud model uses expectation *Ex*, entropy *En*, and hyperentropy *He* as numerical features to represent qualitative concepts: expectation *Ex* reflects the information center value of the corresponding qualitative knowledge and is the determining feature of the concept; entropy *En* is used to measure the ambiguity of the qualitative concept; hyperentropy *He* is the entropy of the entropy, reflecting the random degree of numerical affiliation to the qualitative concept [27], as shown in Fig. 2.

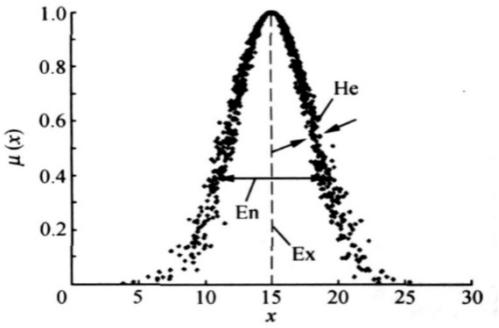



**Fig. 2.** Digital characteristics of the cloud

The cloud generation algorithm, called forward cloud generator, enables the mapping of qualitative concepts to their quantitative counterparts. It generates cloud drops based on the numerical characteristics of clouds, and each cloud drop is a concrete realization of that concept.

Algorithm 1: One-dimensional forward cloud generator

Input: Numerical characteristics *Ex*, *En*, *He* of the normal cloud, the number of cloud drops to be generated *n*.

Output: Quantitative values of cloud drops, and the degree of certainty of each cloud drop representing the concept $\mu(x_i)$ ($i = 1, 2, \ldots, n$).

Step 1: Generate a normal random number $y_i = R_N(En, He)$ with the expected value *En* and standard deviation *He*.

Step 2: Generate a normal random number $x_i = R_N(Ex, |y_i|)$ with the expected value *Ex* and variance $y_i$.

Step 3: Calculate the degree of certainty $\mu(x_i) = \exp\left(\frac{-(x-Ex)^2}{2y_i^2}\right)$.

Step 4: Determine the degree of $\mu(x_i)$ of $x_i$ as a cloud droplet in this number field, which is a concrete realization of the linguistic value represented by this cloud in quantity, and $\mu(x_i)$ as a measure of the degree of belonging to this linguistic value of $x_i$.

Step 5: Repeat steps 1 to 4 until *n* cloud drops are generated.

**Cloud Similarity Metric.** Cloud models are often used to understand the correlation between different clouds by similarity measures, and the similar cloud model (SCM) [28] is a distance-based cloud similarity measure. The cloud droplet distance method SCM is to generate a series of large number of cloud droplets by a forward cloud generator and calculate the distance value between the droplets to represent the similarity between two cloud models.

Algorithm 2: SCM metric algorithm

Input: the first cloud model ($Ex_1$, $En_1$, $He_1$), the second cloud model ($Ex_2$, $En_2$, $He_2$), and the number of cloud drops *n*.

Output: Similarity of the two clouds.

Step 1:. $Drop_1(i) = Cloud(Ex_1, En_1, He_1, n)$.

Step 2:. $Drop_2(i) = Cloud(Ex_2, En_2, He_2, n)$.

Step 3:. Sort ($Drop_1$)

Step 4:. Sort ($Drop_2$)

Step 5: Filter the cloud drops that fall within the 3*En* rule, the number of cloud drops1 is $n_1$ and the number of cloud drops2 is $n_2$ after filtering.

Step 6: Calculate the Distance(*j*) obtained by squaring the corresponding difference between cloud drops 1 and cloud drops 2.

Step 7: Find the average Similar = $\sum (Distance(j))/C_{n_2}^{n_1}$ .

# 3 Disentangling Characterization of Variational Autoencoders Based on Feature Transformation

Since the generation process of the variational autoencoder is like a "black box", this paper explains the latent space part of VAE in order to make its internal working mechanism easier to understand. Most of the current work on feature disentangling focuses on adding penalty terms to the loss function to impose constraints and restrictions, and the disentangling work still remains at a relatively superficial level. To explore the uninterpretable problem in depth, this paper considers that the main difficulty lies in the lack of semantic information in the internal latent space and the unknown mapping relationship between the latent space and the image space, so the interpretable goal of this paper is to explore the mapping relationship by giving semantic information to the internal latent space and separating the features through feature transformation.

The design idea of this paper is to replace the prior distribution of VAE with a Gaussian cloud model. This model design not only improves the quality of reconstructed images, but also the cloud model has excellent ability to represent conceptual uncertainty, so it is conducive to the quantitative measure of disentangling representation after the transformation of latent variable features, and can perform conceptual representation of the separation of semantic features. The main ways to disentangle the feature transformation for VAE that change the prior distribution include internal space modeling, dimensionality reduction transformation of the latent space, explaining the semantic separation and disentangling of the internal latent space by generating changes in specific semantic features in the image, and measuring the degree of disentangling by cloud similarity.

This section presents the overall framework for the interpretation of VAE. The main process is to first change the prior distribution of VAE to optimize the training of the model and improve the image generation quality. Then the hidden space is modeled and the semantic attributes are classified using SVM, and the facial attribute editing is designed by semantic classification boundary, and the generated image after editing is decoupled and quantified using cloud model similarity.

## 3.1 Overall Architecture

The variational autoencoder interpretable method based on feature transformation proposed in this paper is a series of feature transformations for the latent variables sampled from the cloud distribution, so as to solve the variables inside the latent space, and its work mainly includes the following parts: (1) optimization of the variational autoencoder based on the cloud model. (2) construction of semantic subspaces and classification of semantic variables, corresponding to the separation module in the graph. (3) editing of the semantic space, corresponding to the editing module in the graph. (4) disentangling quantification based on the cloud model, which corresponds to the evaluation module in the Fig.. The overall framework diagram is shown in Fig. 3.

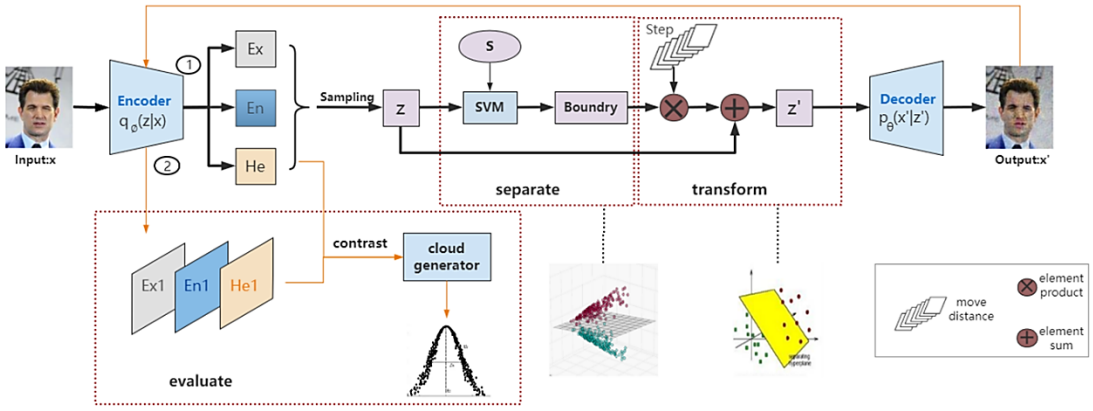



**Fig. 3.** Overall structure diagram

### 3.2 Optimization of Variational Autoencoder

In this paper, the cloud model is used as the prior distribution of VAE, which can make the sampling space larger, equivalent to a process of sampling Gaussian distribution several times, which can make the latent variables of certain detailed features can be sampled; at the same time, the objective function is optimized in order to reduce the error, and the representation learning is carried out for reconstructed samples, and the loss function is shown in equation (2), which adds the reconstruction to the original VAE objective function loss as the penalty term. After the improvement, the reconstructed image quality is improved.

$$\mathcal{L}_{\mathrm{CMVAE}}(\theta,\phi,\alpha,x,x') = \underset{\theta,\phi}{\operatorname{argmin}}[-E_{Z\sim q_\phi(z|x)} logp_\theta(x'|z) + \alpha D_{KL}(q_\phi(z\mid x)\,\|\,p_\theta(z)) + (1-\alpha) D_{KL}(q_\phi(z'\mid x')\,\|\,q_\phi(z\mid x))]. \tag{2}$$

The improved VAE model architecture consists of four main components, an encoder $E_\theta$ composed of a convolutional network, which is able to encode the input image data as Gaussian cloud features. A distribution of the cloud model is generated using the cloud features, which is sampled as a priori distribution. A decoder $D_\phi$ composed of multilayer transposed convolutional layers is decoded to reconstruct the input data. The reconstructed samples are encoded again and the representation is obtained as a constraint in the loss function, thus training the model. The architecture of the model is illustrated in Fig. 4.

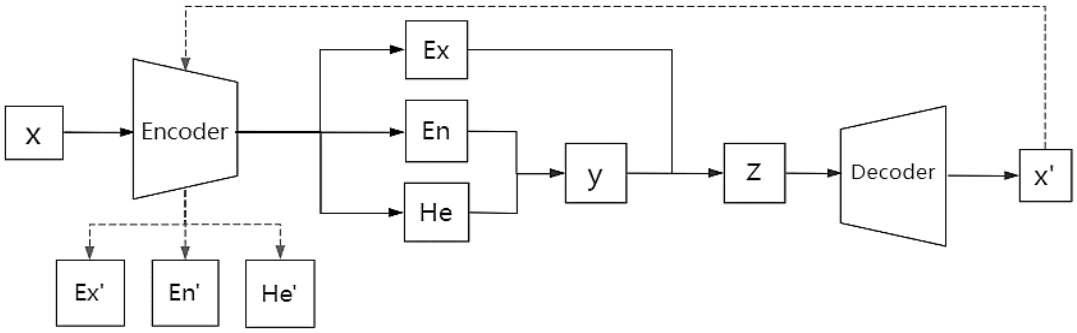



**Fig. 4.** Map of the variational self-encoder model based on the cloud model

### 3.3 Construction of Semantic Subspaces and Classification of Semantic Variables

Since the lack of semantic concepts in the latent space of the variational autoencoder is what makes the internal working mechanism difficult to understand and manipulate, this paper first constructs a semantic subspace to achieve a mapping from the latent space to the semantic space.

In the variational autoencoder, the decoder input latent variable $z(z_1, z_2, \ldots, z_m)$ to the output image $x(x_1, x_2, \ldots, x_m)$ can be regarded as a mapping in the $m$-dimensional latent space $Z \subseteq R^m$ to the high-dimensional image space $X \subseteq R^{H*W*C}$ and constructing its mapping relation as in equation (3). The image space contains numerous semantic information, and the mapping from the image space $X$ to the semantic space $S \subseteq R^n$ is constructed, and its relation is shown in equation (4). By bridging the image space, the semantic labels are embedded in the latent space to obtain the semantic subspace, which the mapping from the latent space $Z$ to the semantic space $S$. The semantic function is shown in equation (5).

$$x = d(z). \tag{3}$$

$$s = f(x). \tag{4}$$

$$s = f(d(z)). \tag{5}$$

In the above equation, $x$ denotes the output image in the image space, $z$ denotes the sampled latent variable, and $s$ denotes the semantic label added to the latent space containing semantic information. In this semantic subspace, a direct functional relationship is constructed between the latent variables and the semantics, which will lay the foundation for the feature transformation in this space.

The prior distribution in this model is a Gaussian cloud distribution, $z$ s generated by two consecutive Gaussian samples. Since the sampling space distribution itself is a continuous distribution, the sampled eigenvalues are also consistent with the continuity of the distribution, and according to the semantic function in the semantic subspace, the semantic corresponding to the continuous $z$ is also continuous, so this paper makes the assumption that for any binary semantics in the semantic subspace, there exists a hyperplane to divide the semantics so that the two sides of the plane represent two different semantics [29].

The core idea of SVM is to find an optimal classification hyperplane that satisfies the classification requirements and maximizes the classification interval while ensuring the classification accuracy.

The latent variable $z$ and his corresponding label $y$ are used as the sample set $(z_i, y_i)$, $i = 1, 2, \ldots, l$, $z \in R^m$, $y \in \{\pm 1\}$ for training the SVM, the hyperplane is $w \cdot z + b = 0$. In order for the classification to correctly classify all samples and have a classification interval, it is required to satisfy:

$$y_i\left[\left(w.z_i\right)+b\right] \geq 1, i=1,2,\ldots,l. \tag{6}$$

This leads to a classification interval of $2/\|w\|$ , after optimization of the LaGrange function, the optimal classification function is:

$$g\left(z\right)=sgn\left\{\left(\sum_{j=1}^{l} a_j^* y_j\left(z_j \cdot z_i\right)\right)+b^*\right\},$$
$$a_j^* > 0, j=1,2,\ldots,l \ . \tag{7}$$

In the semantic subspace, the hyperplane obtained after training by SVM optimization, which has a normal vector $n$, defines the length of the latent variable $z$ to the hyperplane as Eq. (8), and thus the set with length 0 is the hyperplane: $\{z \in R^m : p = 0\}$.

$$p = n^T z \ . \tag{8}$$

When the length changes linearly, its semantics also changes linearly, so construct a linear relationship between the semantic function and the length:

$$s = f\left(d\left(z\right)\right) = \lambda p = \lambda\left(n^T z\right), \lambda > 0 \ . \tag{9}$$

### 3.4 Editing of the Semantic Space

In the semantic subspace, a decision boundary of binary semantic features can be found by the operation in part 3.3. This boundary can divide the latent space into two groups. Taking the semantic feature of gender as an example, the latent variables representing male features and female features are distributed on both sides of the boundary in the classified semantic subspace. In this paper, we believe that if the target concept can be operated from the latent space, that is, if the semantic concept of the generated image can be changed by simply changing the latent variables and thus, the variables in the latent space have been disentangled to some extent, and the process of interpreting the inside of the latent space can be proved in reverse by generating the results.

The operation in the latent space is shown in Fig. 5, and the specific process is as follows: for some semantic boundary, the latent variable $z$ is moved along its normal direction, and the moving process is a linear change, as in equation (10), and the semantic property is changed when the moving distance is large enough to cross the boundary.

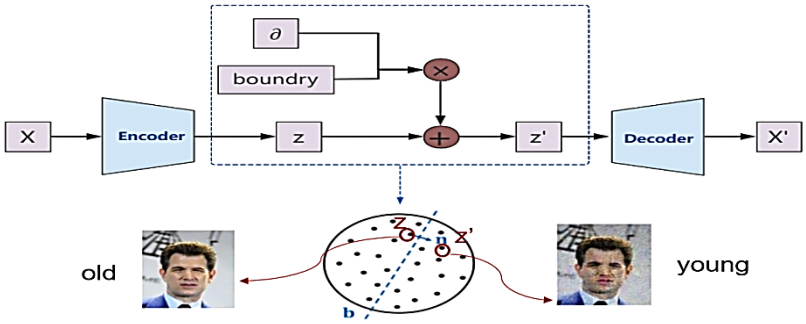



**Fig. 5.** Edit schematic

$$z' = z + \partial n \ . \tag{10}$$

When $\partial > 0$, $z$ is shifted in the positive direction on the same side as the normal vector direction; when $\partial < 0$, $z$ is shifted in the opposite direction on the opposite side of the normal vector direction.

After inputting the changed $z'$ into the decoder of VAE, the resulting $X'$ is the generated graph after the change of the corresponding specific feature. the semantics in $X'$ changes with the latent variable as follows:

$$s' = f\left(d\left(z'\right)\right) = f\left(d\left(z\right)\right) + \lambda\partial \ . \tag{11}$$

### 3.5 Disentangling Evaluation Based on Cloud Model

The analysis of untangling is based on encoding the input image and generating an image while traversing each dimensional value of the latent variable, when changing the dimension of a latent variable, if the generated image changes only one factor of the image, this means that the latent variable is well disentangled [30]. Since there are no standard evaluation metrics for the decomposition model, most interpretable studies stay on the qualitative analysis of experimental results to understand the internal decomposition process intuitively.

In this paper, after separating and editing the improved VAE internal latent variables for new picture generation, the cloud model is used to quantify the degree of explain ability and the semantic conceptual variables isolated in the latent space, and the quantification process is shown in Fig. 6 below.

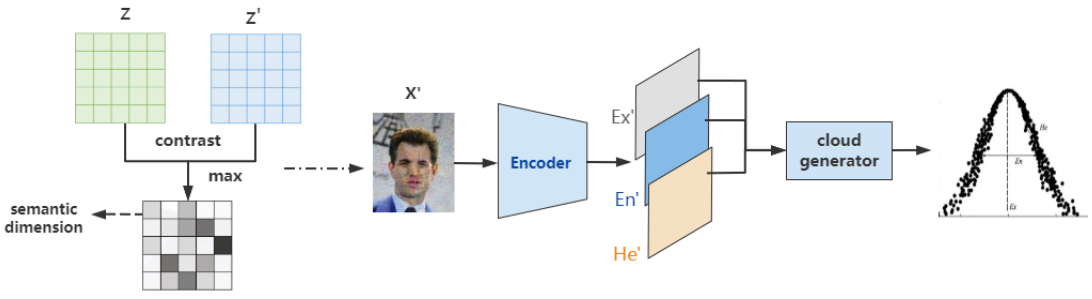



**Fig. 6.** Disentangling evaluation flow chart

The original latent variable $z$, which is sampled by the encoder, is compared with the semantically edited latent variable $z'$, and the dimension with the maximum difference after the comparison is used as the attribute dimension of this semantics. If the difference between before and after is the largest, it means that these dimensions of the latent variable are perpendicular to the hyperplane, and then the degree of semantic change is greater when moving the latent variable along the normal vector direction, which means that this dimension has the strongest

relevance to the semantics, so the dimension of the latent variable that is orthogonal to the hyperplane is the disentangled semantic concept dimension.

The edited image is then input into the encoder, and the digital features (*Ex*, *En*, *He*) of the cloud model can be obtained through the improved architecture of the VAE encoder. The dimensions of the digital features ($Ex_i$, $En_i$, $He_i$) representing specific semantic features are input into the forward cloud generator, and based on the one-dimensional forward cloud generator algorithm, the conversion between qualitative concept $c_i$ and quantitative values can be realized, and the disentangled semantic concept can be quantified through the cloud map. The three digital features in the cloud map illustrate the attribute value, fuzziness and affiliation of the concept, respectively.

In order to quantify the degree of disentangling of the method in this paper, a cloud similarity measure based on cloud droplet distance SCM is used by abstracting the cloud graphs corresponding to the semantic concepts, and this method is used to measure the degree of similarity between different clouds. When the similarity of the cloud graphs before and after editing is lower, it means that the cloud graphs represent different semantic concepts, i.e., the degree of disentangling of this semantic feature is higher.

The process of quantitative representation based on the cloud model is as follows:

$$\begin{aligned} &i\ dimension \leftarrow \max\left|z - z'\right| , \\ &\{Ex, En, He\} \leftarrow Encoder(X) , \\ &c_i \leftarrow \{Ex_i, En_i, He_i\} , \\ &cloud \leftarrow c_i , \\ &cloud\,\text{similarity} \leftarrow \text{cloud}_i . \end{aligned} \tag{12}$$

## 4 Experiments

In this section, in order to verify the effectiveness of the method in this paper, the following experiments are conducted on VAE: (1) the quality optimization of VAE generation based on Gaussian cloud model (2) the classification of hidden space features based on support vector machine. (3) the generation effect of hidden space feature editing. (4) Evaluation experiments of quantitative disentangling.

### 4.1 VAE Optimization Based on Gaussian Cloud Model

To demonstrate the effectiveness of the optimization method proposed in this paper, experiments were conducted on four datasets on VAE, VQ-VAE [31], AVAE [32] and the CMVAE model proposed in this thesis. The four datasets are CelebA [33] (128 × 128), The Car Connection Picture [34] (128 × 128) , NICO [35] (128 × 128)and CIFAR -10 [36] (64 × 64), and they contain four different objects and contexts for human faces, cars, images of various vehicles in different contexts and animals, etc. For each dataset this experiment selects 6K images after alignment and cropping, where the ratio of training set to test set is 10:1. For full generation of the images, the context in which all objects are located is not cropped and is mostly preserved in this paper.

All models use the same network architecture and training data to allow for fair comparisons. The experiments are performed for 1K iterations, and to avoid experimental randomness, the results are averaged over five experiments. The sample reconstruction errors between the model proposed in this thesis and other models are shown in Table 1 below. In the table, it can be seen that the VAE model generates poor image quality on the four data, while the reconstruction error of this paper's method is lower than the other models on all four data sets, proving the improvement of the generation quality.

**Table 1.** Losses of images generated by different models

| | CelebA | The Car Connection Picture | NICO | CIFAR-10 |
|---|---|---|---|---|
| VAE | 277.41 | 503.45 | 516.83 | 1.34 |
| VQ-VAE | 83.78 | 157.48 | 394.76 | 1.23 |
| AVAE | 188.39 | 305.89 | 287.46 | 1.19 |
| CMVAE | 27.2 | 55.12 | 84.3 | 1.15 |

### 4.2 SVM Parameter Settings and Classification Results

According to section 3.3 of this paper, support vector machines are used to classify the latent variables in the latent space to find semantic boundaries. In this paper, experiments are conducted on the CalebA dataset, and different sizes of 8K, 10K, 16K and 20K datasets are selected for training. Each size dataset is uniformly aligned and cropped, and then input to the encoder of the trained optimized VAE for coding and sampling to obtain the latent variables, which are used as training data. And the dataset comes with attribute tags, and five tags (gender, smile, age, eyebrows, and glasses) from the forty attribute tags are selected as tags to train the support vector machine with five different attributes, and the kernel function of the support vector machine uses a linear kernel function.

The ratio of the training set to the test set is 7:3, and the number of positive and negative samples for each feature is 1:1. The classification accuracy of the five attribute SVMs is averaged as the final accuracy through the training of different size of face datasets, and the results are obtained as shown in Table 2.

**Table 2.** Accuracy of SVM classification for different size datasets

| Nums | Auc |
|---|---|
| 8K | 0.832 |
| 10K | 0.856 |
| 16K | 0.831 |
| 20K | 0.843 |

It can be concluded that the highest classification accuracy of the SVM is obtained on a face dataset of size 10K, and thus the next experiments use a data size of 10K, and the high classification accuracy also shows that there is a hyperplane to divide the binary attributes in the latent space.

### 4.3 Latent Space Feature Editing Results

This part of the experiment shows the results after editing and manipulating the features in the latent space, aiming to demonstrate the separability and manipulability of the semantic variables in the latent space. Based on the semantic boundary obtained after SVM classification, the latent variables are moved linearly along the vertical direction of the boundary in the semantic subspace, and the farthest boundaries of the positive and negative directions of z-movement are set to 3 and -3, respectively, in the experiments, and are moved 3 steps in the positive and negative directions, respectively. Five semantic features are edited, namely, gender, smile, age, eyebrows, and glasses, and the generated graphs after editing the latent variables are shown in Fig. 7.

In Fig. 7, the image near the middle part of each row is closest to the original synthesis of the optimized VAE, and the three samples from the middle to the left are the output results of each step after moving the hidden variable z three steps along the negative direction, and similarly the three samples on the right are the results of moving three steps along the positive direction. From the Fig., it can be seen that moving the hidden variable according to a specific semantic boundary can change the semantic features of the image, and the degree of feature change is positively correlated with the distance moved. For example, for the gender semantics in the first line, when $z$ is moved in the negative direction, the facial features show more feminine elements, such as eye shadow and lipstick, while when $z$ is moved in the positive direction, the male-specific beard elements appear. It can be seen that the learning process of VAE can learn these semantically related features, and he is not a single feature part, but a change of numerous related features that can express the semantics of gender, and the same for the editing of several other semantic features.

Through the generation results after editing, it can be concluded that variable dimensions with specific semantic concepts can be found in the latent space. Changing these semantic dimensions can change the corresponding features of the generated images, proving that the method in this paper can decouple the variables representing semantics from the black box general hidden space.

From the generated results, we can see that the quality of the images near the middle is better, and the blurring appears in the samples with more obvious feature changes, which is influenced by the quality of the reconstructed images by the model itself on the one hand. Because the hidden variable $z'$ after editing is not constrained by the loss function constructed by the trained model on the other hand, so the phenomenon that the edge samples are presented.

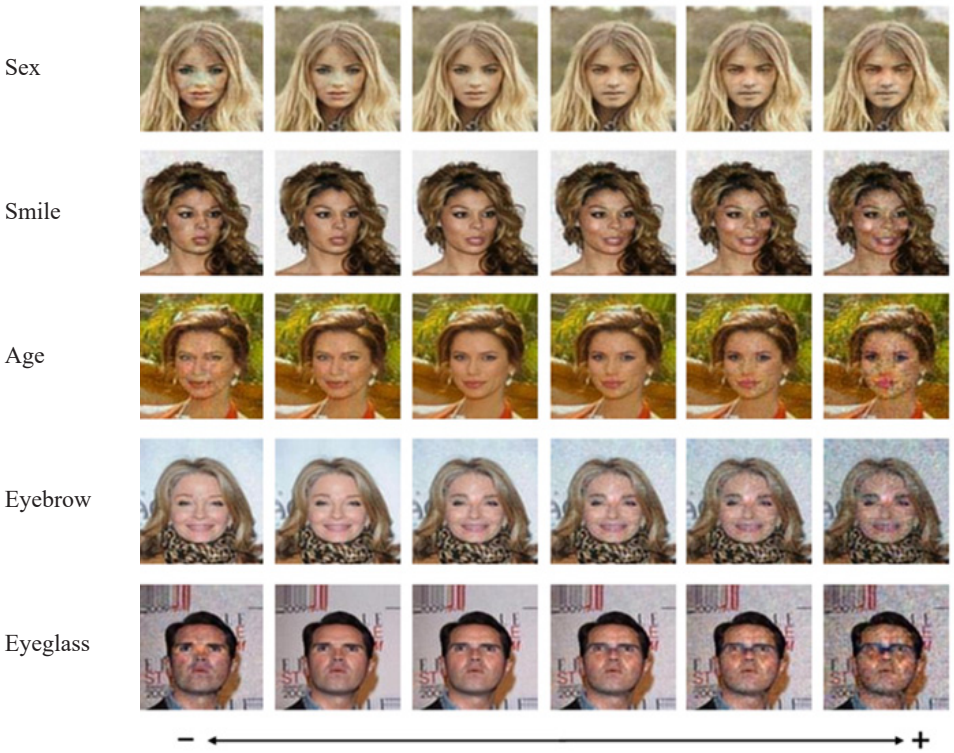



**Fig. 7.** Latent variable editing generation diagram

### 4.4 Feature Disentangling Evaluation

In the previous section, the interpretable process of VAE is qualitatively illustrated by the results of the generated images, and the effect of the separated decoupled variables on the picture features is intuitively felt. However, images cannot accurately illustrate the degree of interpretation, so this paper quantitatively characterizes the decoupled abstract semantic concepts with the help of the cloud model used a priori after optimizing the VAE, and quantitatively measures the degree of decoupling by the difference in the change of the conceptual cloud features before and after editing and by using the cloud similarity.

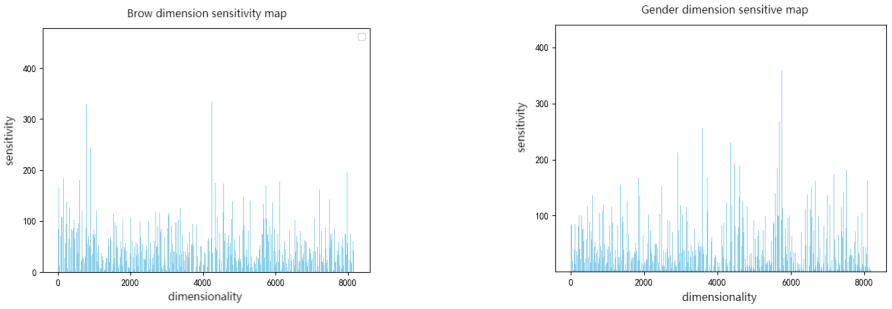


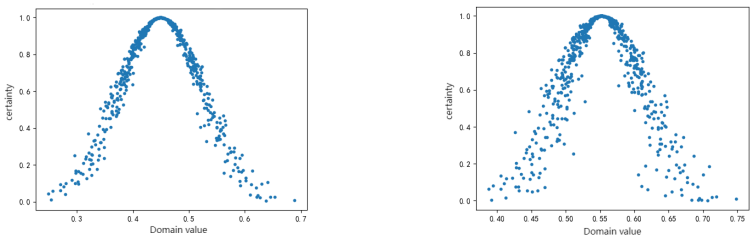

**Fig. 8.** Semantic dimensional sensitivity map

By making the difference between the dimensions of the hidden variables before and after editing, the sensitivity of each dimension to different semantics can be reflected. The dimension with the highest sensitivity is

used as the semantic feature dimension. Fig. 8 shows the sensitivity of each dimension of the hidden variables to the semantics of eyebrows and gender, and their semantic feature dimensions are 4217 and 5630, respectively, and the same for other semantic dimensions.

The semantic feature dimensions after encoding the pictures are transformed and these semantic features separated from the hidden space are described quantitatively. In this paper, a cloud generation algorithm is used to implement the mapping of concepts, and 600 cloud drops are generated for each concept in the experiments, each cloud drop being a concrete realization of that concept at a time.

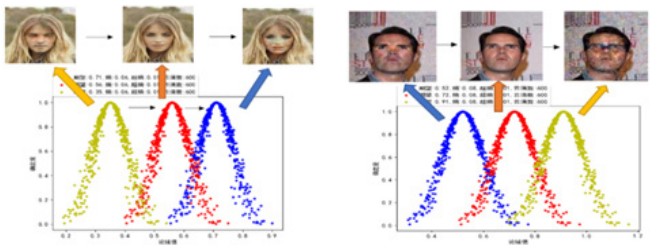
Two randomly selected concept cloud maps representing smile and age are shown in Fig. 9. The Fig. shows that the expectation of two different concepts of smile and age are 0.45 and 0.57, representing two different concept values. The expectation changes as the characteristics of the editing hidden variable change. The entropy values of smile and age are 0.08 and 0.06, reflecting the randomness and vagueness of this dimension representing this concept, which shows that the characteristic dimensional representation of smile is more than that of the concept dimension of age vague. The super entropy values of the two Figs are 0.01 and 0.02, respectively, which can also be seen from the images that the distribution of the cloud representing age is more discrete, indicating that the randomness of the affiliation to age is greater.



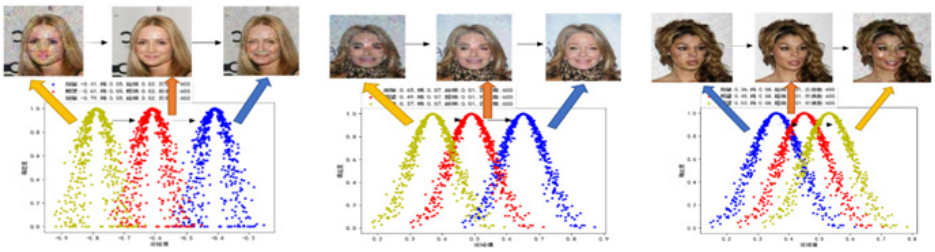

(a) Smile semantics (b) Age semantics

**Fig. 9.** Cloud model diagram of semantic concepts

The cloud image before and after feature editing is compared with the original cloud image of the reconstructed image, as shown in Fig. 10. It can be seen that the entropy and super entropy of similar semantics did not change when the image features were changed, and only the expected value representing the attribute value changed, indicating that the fuzziness and affiliation of the semantic concepts were not changed during the feature editing process. According to the SCM cloud similarity measure, the similarity degree of three cloud pictures of the same semantic meaning is used as a measure of the decoupling degree of this semantic meaning, and the results of the measure are shown in Table 3, from which it can be obtained that the decoupling degree of glasses and gender is larger.



(a) Sex (b) Eyeglasses



(c) Age (d) Eyebrow (e) Smile

**Fig. 10.** Feature editing cloud model diagram

**Table 3.** Decoupling degree table

| | Gender | Smile | Age | Eyebrows | Glasses |
|---|---|---|---|---|---|
| Decoupling degree | 0.147 | 0.069 | 0.155 | 0.114 | 0.159 |

## 5 Conclusion

In this paper, a variational autoencoder disentanglement method based on cloud model is introduced, which can isolate the features representing specific semantics from VAE hidden Spaces. In this paper, the decoupled characterization is carried out in VAE when the cloud model is a priori distribution, and the characteristic variables of hidden space are classified and edited through the feature transformation after the modeling of hidden space. The internal hidden space is interpreted by generating directional changes of image features. In this paper, the similarity measurement between cloud models is used to quantitatively evaluate the degree of disentanglement in hidden space, and a series of experiments prove the effectiveness of the proposed method. This paper provides a new idea for the disentanglement representation of hidden space, which can be applied to the directional editing of image features. In the future, the hidden space can be further analyzed in detail, so as to solve the disentanglement more thoroughly.

## 6 Acknowledgement

Grant: National Natural Science Foundation of China (61936001, 61772096), Natural Science Foundation of Chongqing (cstc2021jcyj-msxmX0849).

## References

[1] D.P. Kingma, M. Welling, Auto-encoding variational bayes. <https://arxiv.org/abs/1312.6114>, 2013 (accessed 20.12.2013).
[2] J. Walker, C. Doersch, A. Gupta, M. Hebert, An uncertain future: forecasting from static images using variational autoencoders, in: Proc. Computer Vision–ECCV 2016: 14th European Conference, 2016.
[3] V. Balasubramanian, I. Kobyzev, H. Bahuleyan, I. Shapiro, O. Vechtomova, Polarized-vae: proximity based disentangled representation learning for text generation. <https://arxiv.org/abs/2004.10809>, 2020 (accessed 22.04.2020).
[4] W.X. Shi, H. Zhou, N. Mian, L. Li, Dispersed exponential family mixture vaes for interpretable text generation, in: Proc. International Conference on Machine Learning, 2020.
[5] S.Y. Dai, Z. Gan, Y. Cheng, C. Tao, L. Carin, J. Liu, APo-VAE: text generation in hyperbolic space, <https://arxiv.org/abs/2005.00054>, 2020 (accessed 30.04.2020).
[6] Y.Z. Zhu, M.R. Min, A. Kadav, H.P. Graf, S3VAE: Self-supervised sequential vae for representation disentanglement and data generation, in: Proc. of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020.
[7] C.C. Lin, Y. Hung, R. Feris, L. He, Video instance segmentation tracking with a modified vae architecture, in: Proc. of

the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020.
[8] R.A. Yeh, C. Chen, T.Y. Lim, A.G. Schwing, M. Hasegawa-Johnson, M.N. Do, Semantic image inpainting with deep generative models, in: Proc. of the IEEE conference on computer vision and pattern recognition, 2017.
[9] J. Xu, Y. W. Teh, Controllable semantic image inpainting, <https://arxiv.org/abs/1806.05953>, 2018 (accessed 15.06.2018).
[10] Q.S. Zhang, S.C. Zhu, Visual interpretability for deep learning: a survey, Frontiers of Information Technology & Electronic Engineering 19(1)(2018) 27-39.
[11] S.L. Ji, J.F. Li, T.Y. Du, B. Li, Survey on Techniques, Applications and Security of Machine Learning Interpretability, Journal of Computer Research and Development 56(10)(2019) 2071-2096.
[12] G.Y. Wang, C.L. Xu, Q.H. Zhang, X.R. Wang, p-order Normal Cloud Model Recursive Definition and Analysis of Bidirectional Cognitive Computing, Chinese Journal of Computers 36(11)(2013) 2316-2329.
[13] D.Y. Li, H.J. Meng, X.M. Shi, Membership Clouds and Membership Cloud Generators, Journal of Computer Research and Development 32(6)(1995) 15-20.
[14] I. Higgins, L. Matthey, A. Pal, C. Burgess, X. Glorot, M. Botvinick, S. Mohamed, A. Lerchner, Beta-vae: learning basic visual concepts with a constrained variational framework, in: Proc. of the 5th International Conference on Learning Representations, 2017.
[15] H. Kim, A. Mnih, Disentangling by factorizing, in: Proc. of the 35th International Conference on Machine Learning. Stockholm, 2018.
[16] R.T.Q. Chen, X.C. Li, R. Grosse, D. Duvenaud, Isolating sources of disentanglement in VAEs, in: Proc. of the 32nd International Conference on Neural Information Processing Systems, 2018.
[17] G. Mita, M. Filippone, P. Michiardi, An identifiable double vae for disentangled representations, in: Proc. of the International Conference on Machine Learning, 2021.
[18] A. Komanduri, Y. Wu, W. Huang, F. Chen, X. Wu, SCM-VAE: learning identifiable causal representations via structural knowledge, in: Proc. of the IEEE International Conference on Big Data, 2022.
[19] S. Eslami, N. Heess, T. Weber, Y. Tassa, D. Szepesvari, K. Kavukcuoglu, G.E. Hinton, Attend, infer, repeat: fast scene understanding with generative models, in: Proc. of the 30th International Conference on Neural Information Processing Systems, 2016.
[20] X.P. Li, Z.R. Chen, L.K.M. Poon, N.L. Zhang, Learning latent superstructures in variational autoencoders for deep multidimensional clustering, in: Proc. of the 7th International Conference on Learning Representations, 2019.
[21] M. Yang, F. Liu, Z Chen, X. Shen, J. Hao, J. Wang, Causalvae: disentangled representation learning via neural structural causal models, in: Proc. of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2021.
[22] Z. Xu, J. Liu, D. Cheng, J. Li, L. Liu, K. Wang, Disentangled representation with causal constraints for counterfactual fairness. <https://arxiv.org/abs/2208.09147>, 2022 (accessed 27.05.2022).
[23] D. Bouchacourt, R. Tomioka, S. Nowozin, Multi-level variational autoencoder: learning disentangled representations from grouped observations, in: Proc. of the 32nd AAAI Conference on Artificial Intelligence, 2018.
[24] M. Vowels, N. Camgoz, R. Bowden, Gated variational autoencoders: incorporating weak supervision to encourage disentanglement, in: Proc. of the 15th IEEE International Conference on Automatic Face and Gesture Recognition, 2020.
[25] J. Xu, Y. Ren, H. Tang, X. Pu, X. Zhu, M. Zeng, L. He, Multi-VAE: learning disentangled view-common and view-peculiar visual representations for multi-view clustering, in: Proc. of the IEEE/CVF International Conference on Computer Vision, 2021.
[26] J. Zhu, H. Xie, W. Abd-Almageed, SW-VAE: weakly supervised learn disentangled representation via latent factor swapping, in: Proc. of the European Conference on Computer Vision, 2022.
[27] J. Yang, G.Y. Wang, Q. Liu, Y.K. Guo, Y. Liu, W.Y. Gan, Y.C. Liu, Retrospect and prospect of research of normal cloud model, Chinese Journal of Computers 41(3)(2018) 724-744.
[28] Y. Zhang, D.N. Zhao, D.Y. Li, The Similar Cloud and the Measurement Method, Information and Control 33(2)(2004) 129-132.
[29] Y. Shen, J. Gu, X. Tang, B. Zhou, Interpreting the latent space of gans for semantic face editing, in: Proc. of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020.
[30] Y. Bengio, A. Courville, P. Vincent, Representation learning: a review and new perspectives, IEEE Transactions on Pattern Analysis and Machine Intelligence 35(8)(2013) 1798-1828.
[31] A.V.D. Oord, O. Vinyals, K. Kavukcuoglu, Neural discrete representation learning, in: Proc. of Advances in Neural Information Processing Systems, 2017.
[32] A. Plumerault, H.L. Borgne, C. Hudelot, AVAE: Adversarial variational auto encoder. <https://arxiv.org/abs/2012.11551>, 2020 (accessed 10.01.2021).
[33] Z.W. Liu, P. Luo, X. Wang, X. Tang, Deep learning face attributes in the wild, in: Proc. of the IEEE International Conference on Computer Vision, 2015.
[34] N. Gervais, Data decorators, <https://www.graviti.cn/open-datasets/ CarConn -ectionPicture>, 2021 (accessed 23.05.2021).
[35] Y. He, Z.Y. Shen, P. Cui, Towards non-i.i.d. image classification: a dataset and baselines, Pattern Recognition 110(2021) 107383.
[36] A. Krizhevsky, Learning multiple layers of features from tiny images, Technical Report, University of Toronto, 2009.