Domain Disentangled Generative Adversarial Network for Zero-Shot Sketch-Based 3D Shape Retrieval

Rui Xu, Zongyan Han, Le Hui, Jianjun Qian, Jin Xie*

PCA Lab, Key Lab of Intelligent Perception and Systems for High-Dimensional Information of Ministry of Education
Jiangsu Key Lab of Image and Video Understanding for Social Security
School of Computer Science and Engineering, Nanjing University of Science and Technology, China
{xu_ray, hanzy, le.hui, csjqian, csjxie}@njust.edu.cn

Abstract

Sketch-based 3D shape retrieval is a challenging task due to the large domain discrepancy between sketches and 3D shapes. Since existing methods are trained and evaluated on the same categories, they cannot effectively recognize the categories that have not been used during training. In this paper, we propose a novel domain disentangled generative adversarial network (DD-GAN) for zero-shot sketch-based 3D retrieval, which can retrieve the unseen categories that are not accessed during training. Specifically, we first generate domain-invariant features and domain-specific features by disentangling the learned features of sketches and 3D shapes, where the domain-invariant features are used to align with the corresponding word embeddings. Then, we develop a generative adversarial network that combines the domain-specific features of the seen categories with the aligned domain-invariant features to synthesize samples, where the synthesized samples of the unseen categories are generated by using the corresponding word embeddings. Finally, we use the synthesized samples of the unseen categories combined with the real samples of the seen categories to train the network for retrieval, so that the unseen categories can be recognized. In order to reduce the domain shift between the synthesized domain and the real domain, we adopt the transductive setting to reduce the gap between the distributions of the synthesized unseen categories and real unseen categories. Extensive experiments on the SHREC'13 and SHREC'14 datasets show that our method significantly improves the retrieval performance of the unseen categories.

Introduction

In recent years, with the massive increase in the number of 3D models, 3D shape retrieval has attracted widespread attention. Existing methods mainly contain three categories, including shape-based methods (Iyer et al. 2005; Xie et al. 2015; Zhu et al. 2016; Jiang et al. 2019), text-based methods (Min, Kazhdan, and Funkhouser 2004; Goldfeder and Allen 2008) and sketch-based methods (Eitz et al. 2012; Furuya and Ohbuchi 2013; Li et al. 2014a; Dai, Xie, and Fang 2018). Lately, compared with the shape-based and text-based 3D shape retrieval, sketch-based 3D shape retrieval has achieved more attention from researchers since the sketch is a more intuitive way for humans to interact with data collection. However, due to the abstraction of sketches and the discrepancy between 2D sketches and 3D shapes, sketch-based 3D shape retrieval is still a challenging problem.

Recently, many efforts have been made in sketch-based 3D retrieval. In the early years, many works improve the performance of sketch-based 3D retrieval by learning robust 3D shape features (Wang, Kang, and Li 2015; Xie et al. 2017; Zhu, Xie, and Fang 2016a,b; Xu et al. 2020b). (Wang, Kang, and Li 2015) selected two different views with angles larger than 45 degrees to characterize 3D shapes. And a siamese convolutional neural network was used to extract the features of sketches and projections of 3D shapes. To further improve 3D shape representations, (Xie et al. 2017) proposed to learn the wasserstein barycenters of the multi-views projections. In addition, how to measure the cross-domain similarity between sketches and 3D shapes is also crucial (Dai et al. 2017; He et al. 2018). In (Dai et al. 2017), the discriminative loss is proposed to increase the distinction of different categories in each domain, and the correlation loss is used to minimize the domain discrepancy between sketch and 3D shape. In sketch-based 3D shape retrieval, these methods have achieved remarkable results in categories that are used for both training and evaluation. Nonetheless, due to the imbalance of data and some data without labels in reality, these traditional methods cannot be applied to these scenarios. They cannot effectively retrieve the unseen categories that are not used during training.

In this paper, we propose a novel domain disentangled generative adversarial network (DD-GAN) to effectively retrieve the unseen categories in sketch-based 3D shape retrieval. The key idea of our method is to utilize the generative adversarial network to generate samples of the unseen categories. In order to effectively retrieve the unseen categories, we retrain the network by utilizing the generated samples of the unseen categories and the real samples of the seen categories. Specifically, in our DD-GAN, we first use SketchCNN and ShapeCNN to extract features of sketches and 3D shapes, respectively. We disentangle the features of each domain by using three fully connected layers to obtain domain-invariant features ($i.e.$, pure semantic information) and domain-specific features ($i.e.$, contour style

[*] Corresponding author.

or texture information). Then, we propose the metric module to enhance the discrimination of the domain-invariant features between different categories and align the domain-invariant features with the corresponding word embedding of the category. In the metric module, we adopt the triplet loss to increase the inter-domain distance and reduce the intra-domain distance to enhance the identification of different categories in the semantic space. In addition, we align the domain-invariant features with the corresponding word embedding by computing the cosine similarity between them. After that, the domain-specific features are used as the condition combined with the domain-invariant features by the generative adversarial network to generate samples of the corresponding categories. By utilizing the discriminator to distinguish the generated samples from the real samples, it is desired that the generative adversarial network can generate high-quality samples of different categories, thereby improving the discrimination of domain-invariant features of different categories.

In order to alleviate the domain shift problem of the unseen categories, we use the word embedding of the unseen categories combined with domain-specific features of the seen categories to generate the samples of the unseen categories. By using the discriminator to distinguish the generated samples and the real samples, we can reduce the domain shift of the unseen categories. Finally, we uses the generated samples of the unseen categories combined with the real samples of the seen categories to train our network. We use the obtained domain-invariant features for retrieval. Experimental results on the SHREC'13 and SHREC'14 datasets demonstrate the effectiveness of the proposed method for sketch-based 3D shape retrieval. Especially, our method can effectively retrieve the unseen categories.

In summary, our main contributions are as follows: (1) To the best of our knowledge, we are the first to consider zero-shot sketch-based 3D shape retrieval. (2) We propose a novel domain disentangled generative adversarial network (DD-GAN) that can learn the discriminative features of different domains by decomposing and combining domain-invariant features and domain-specific features to generate samples of different categories. (3) Based on DD-GAN, we employ a transductive setting that utilizes the word embedding of the unseen categories to generate high-quality samples of the unseen categories to alleviate the domain shift problem. (4) Experimental results on the SHREC'13 and SHREC'14 datasets show that our method can effectively retrieve the unseen categories.

Related Work

Sketch-Based 3D Shape Retrieval

In the past, researches employed various hand-crafted features to describe sketches and 3D shapes (Saavedra et al. 2012; Li and Johan 2013; Li et al. 2017a; Yoon and Yoon 2017). (Yoon and Yoon 2017) proposed a sparse coding based methods to match the HOG-SIFT features of sketches and 3D objects. (Saavedra et al. 2012) used histogram of keyshape orientations (HKO) as global descriptors to determine the appropriate viewpoint of 3D shapes, then employed keyshape angular spatial distribution (KAS-D) as local descriptors to match the sketches and 3D shapes. (Li et al. 2017a) proposed a viewpoint entropy distribution to describe 3D shapes, and obtained a set of representative sample views of 3D shapes by adaptive view clustering for 2D-3D comparison.

Recently, with the rapid development of deep learning, deep features extracted by neural networks gradually replaces traditional hand-crafted features (Tasse and Dodgson 2016; Kuwabara, Ohbuchi, and Furuya 2019; Chen et al. 2019; Dai and Liang 2020; Liu and Zhao 2021). Many works focus on learning effective deep features and design methods to measure cross-domain similarity of sketch and 3D shape. (Xu et al. 2020b) selected the best perspective projections of the 3D shapes according to the perspective of the training sketches, and used MVCNN (Su et al. 2015) to extract the features of these projections to obtain the robust 3D shape features. Based on the multi-view pairwise relationship(MVPR) learning, (Li et al. 2017b) proposed a probabilistic framework to infer the pairwise relationship between sketches and projections of 3D shapes to tackle the retrieval problem. (Chen and Fang 2018) designed a transformation network to transform sketch features into 3D shape feature space. At the same time, they used the cross-modality mean discrepancy minimization to enhance the correlations of transformed sketch features and 3D shape features. (He et al. 2018) proposed a novel triplet-center loss, which can directly minimizing the intra-class distance while maximizing the inter-class distance for the two different domains. For the first time, (Qi, Song, and Xiang 2018) aligned the sketch domain and 3D shape domain in an easier common semantic space rather than the usual joint feature space. Different from these methods, we first decompose the features of sketch domains and 3D shape domain into domain-specific features and domain-invariant features to reduce the domain gap, then measure the similarity of domain-invariant features in the common feature space.

Zero-Shot Learning

Nowadays, zero-shot learning has attracted widespread attention and has been applied to various visual tasks, such as image classification (Xian et al. 2016; Mensink, Gavves, and Snoek 2014; Bucher, Herbin, and Jurie 2017; Han, Fu, and Yang 2020), semantic segmentation (Bucher et al. 2019; Kato, Yamasaki, and Aizawa 2019) and sketch-based image retrieval (SBIR) (Dey et al. 2019; Dutta and Biswas 2019; Pandey et al. 2020) etc. Many methods map visual features to high-dimensional semantic space, and utilize attributes as a “bridge” for knowledge transfer from seen categories to unseen categories (Akata et al. 2015; Kodirov, Xiang, and Gong 2017; Han, Fu, and Yang 2020; Han et al. 2021). Recently, few methods use generative model to generate unseen categories, and then train their model from conventional supervision way (Xian et al. 2019; Huang et al. 2019; Gao et al. 2020). For more information about the zero-shot topic, please refer to the comprehensive survey(Xian et al. 2018).

Actually, our task is related to zero-shot SBIR, that both involves two different domains. (Yelamarthi et al. 2018) used conditional variational autoencoder (CVAE) and ad-

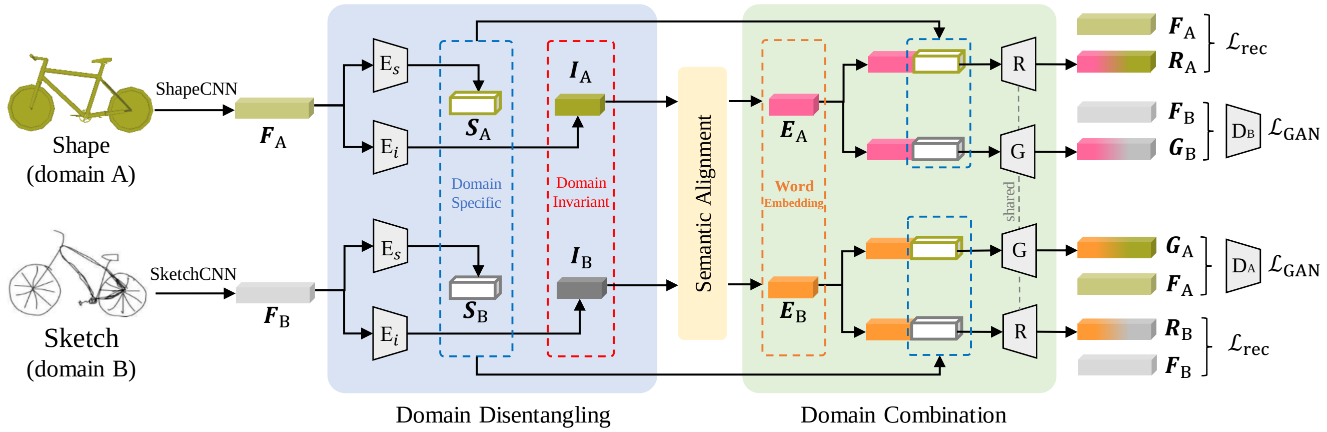


Figure 1: Illustration of our domain disentangled generative adversarial network (DD-GAN). Specifically, we use feature decomposition to obtain domain-specific features and domain-invariant features from sketch features and 3D shape features. The semantic alignment module is used to enhance the discrimination of domain-invariant features, and align semantic features mapped from visual features with corresponding word embeddings. The domain combination module completes reconstruction and cross-domain generation by combining semantic features with different domain variable features.

versarial autoencoder (CAAE) to associate the characters of sketch with that of the image. (Dutta and Akata 2019) proposed a semantically aligned paired cycle-consistent generative (SEM-PCYC) to map the visual information of sketch and image to a common semantic space. And using the generated embeddings of unseen class through the learned mapping for retrieval. (Deng et al. 2020) proposed a progressive cross-domain semantic network first generates semantic features, and then further generates the final retrieval features. A cross-reconstruction loss is used to guarantees reduce the domain gap between sketch and image, and the progressive generation of retrieval features is conducive to transforming knowledge from seen class to unseen class. (Liu et al. 2019) and (Xu et al. 2020a) both employed feature decomposition to reduce the inter-domain difference of sketch and image. (Liu et al. 2019) generated fake images through a style-guide image generator for the final retrieval. (Xu et al. 2020a) mapped the semantic features of two domains into a common feature space as retrieval features.

It is worth mentioning that although our proposed method also uses feature decomposition, our method was completely different from theirs. First of all, the way we decompose was different. They used the negative classification loss to get the domain features/style features, and we obtained domain-specific features through adversarial generation. Second, the role of the domain-specific features were different. (Xu et al. 2020a) just separated domain-specific features to reduce the domain gap between sketch and image. (Liu et al. 2019) further used the domain-specific features of image and the sketch content features to generate fake images, and employed triplet loss to constrain the generated images. However we needed to generate samples of sketch domain and 3D shape domain, the domain-specific features were the condition for us to generate samples of the corresponding domain. In addition, our adversarial generation can better learn the distribution of real data. Finally, we used unseen word embeddings to generate unseen samples, and the unlabeled data was used to improve the quality of our generated unseen samples.

Methodology

we first give the preliminary definition of the zero-shot sketch-based 3D shape retrieval. Given the dataset $\mathcal{D} = \{(\boldsymbol{x}_i, \boldsymbol{y}_i, c_i) \mid c_i \in \mathcal{C}\}$, where $\boldsymbol{x}_i$, $\boldsymbol{y}_i$, and c_i are the sketch, 3D shape, and the label of the i-th sample, respectively. Here, $\mathcal{C}$ denotes the set of different categories. In the zero-shot setting, we split the whole categories into two sets $\mathcal{C}_{seen}$ and $\mathcal{C}_{unseen}$, where $\mathcal{C}_{seen} \cup \mathcal{C}_{unseen} = \mathcal{C}$ and $\mathcal{C}_{seen} \cap \mathcal{C}_{unseen} = \varnothing$. According to the sets $\mathcal{C}_{seen}$ and $\mathcal{C}_{unseen}$, we can obtain the training set $\mathcal{D}_{train} = \{(\boldsymbol{x}_i, \boldsymbol{y}_i, c_i) \mid c_i \in \mathcal{C}_{seen}\}$ and test set $\mathcal{D}_{test} = \{(\boldsymbol{x}_i, \boldsymbol{y}_i, c_i) \mid c_i \in \mathcal{C}_{unseen}\}$. In addition, zero-shot learning (ZSL) can be divided into the inductive setting and the transductive setting according to the availability of unlabeled data. In the inductive setting, we use the set $\mathcal{D}_{train}$ to train the network and use the set $\mathcal{D}_{test}$ for evaluation. However, in the transductive setting, in addition to using the set $\mathcal{D}_{train}$ to train the network, we additionally use the set $\mathcal{D}_{test}$, but do not use the ground truth. Note that in this paper, our method is under the transductive setting.

Domain Disentangling

Feature Disentanglement. In order to reduce the domain discrepancy between 2D sketches and 3D shapes, we develop the domain disentangling module to extract the domain-invariant features and domain-specific features. As shown in Figure 1, we depict the details of domain disentangling. Given the samples of the 3D shape (dubbed domain A) and sketch (dubbed domain B), we first use ShapeCNN and SketchCNN to extract the initial features $\boldsymbol{F}_A \in \mathbb{R}^D$ and $\boldsymbol{F}_B \in \mathbb{R}^D$, where D is the dimension of the feature vector.

Based on multi-view convolution neural network (MVCNN) (Su et al. 2015), ShapeCNN uses the pre-trained ResNet-50 (He et al. 2016) on the multiple 2D images to extract initial feature $\boldsymbol{F}_A$. SketchCNN also uses the pre-trained ResNet-50 on the sketch to extract initial features $\boldsymbol{F}_B$.

Once we obtain the features $\boldsymbol{F}_A$ and $\boldsymbol{F}_B$ of two domains, we employ the domain-invariant encoder E_i and domain-specific encoder E_s to obtain domain-invariant and domain-specific features, respectively. The domain-invariant features and domain-specific features of the 3D shape and sketch are formulated as:

$$\begin{aligned} \boldsymbol{I}_A = E_i(\boldsymbol{F}_A), \boldsymbol{I}_B = E_i(\boldsymbol{F}_B) \\ \boldsymbol{S}_A = E_s(\boldsymbol{F}_A), \boldsymbol{S}_B = E_s(\boldsymbol{F}_B) \end{aligned} \tag{1}$$

where $\boldsymbol{I}_A \in \mathbb{R}^{D'}$, $\boldsymbol{I}_B \in \mathbb{R}^{D'}$, $\boldsymbol{S}_A \in \mathbb{R}^{D'}$, and $\boldsymbol{S}_B \in \mathbb{R}^{D'}$ are the domain-invariant features and domain-specific features of the sketch and 3D shape, respectively. It is desired that the domain-invariant features can characterize the semantic information, while the domain-specific features can preserve the unique characteristic (such as texture information) of the domain itself. In the experiment, we use three fully connected layers to implement the encoders E_i and E_s. Note that the encoders are not shared in different domains.

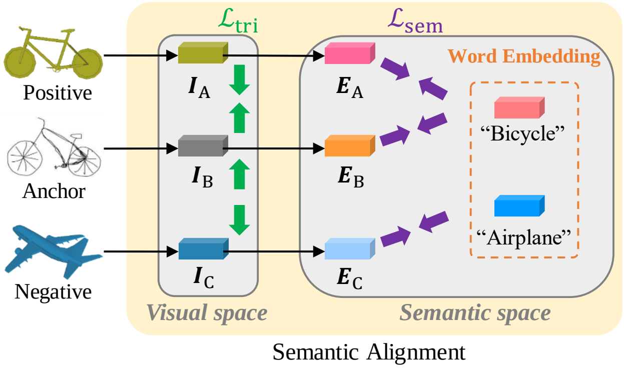

$\mathcal{L}_{\text{tri}}$
$\mathcal{L}_{\text{sem}}$
Word Embedding
Positive
Anchor
Negative
$\boldsymbol{I}_{\text{A}}$
$\boldsymbol{I}_{\text{B}}$
$\boldsymbol{I}_{\text{C}}$
$\boldsymbol{E}_{\text{A}}$
$\boldsymbol{E}_{\text{B}}$
$\boldsymbol{E}_{\text{C}}$
"Bicycle"
"Airplane"
Visual space
Semantic space
Semantic Alignment

Figure 2: Semantic alignment module.

Semantic alignment. In order to identify the unseen categories, we develop the semantic alignment module by mapping the domain-invariant features to the word embedding space to transfer the knowledge from the seen categories to the unseen categories. Here, we also construct the triplet of anchor, positive, and negative for conduct triplet loss, thereby enhancing the discrimination of the domain-invariant features of different categories. As shown in Figure 2, given the triplet of anchor, positive, and negative, we first obtain the domain-invariant features $\boldsymbol{I}_A$, $\boldsymbol{I}_B$, and $\boldsymbol{I}_C$, respectively. Then, we perform the triplet loss to minimize the distance from the anchor sample ($\boldsymbol{I}_B$) to the positive sample ($\boldsymbol{I}_A$) and maximize the distance from the anchor sample ($\boldsymbol{I}_B$) to the negative sample ($\boldsymbol{I}_C$). After that, we use the fully connected layers to align the dimension of the domain-invariant features to the dimension of the wording embedding, which is formulated as:

$$\boldsymbol{E}_A = \phi(\boldsymbol{I}_A), \boldsymbol{E}_B = \phi(\boldsymbol{I}_B), \boldsymbol{E}_C = \phi(\boldsymbol{I}_C) \tag{2}$$

where the $\boldsymbol{E}_A \in \mathbb{R}^d$, $\boldsymbol{E}_B \in \mathbb{R}^d$ and $\boldsymbol{E}_C \in \mathbb{R}^d$ are the semantic features of $\boldsymbol{A}$, $\boldsymbol{B}$, and $\boldsymbol{C}$, respectively. $\phi(\cdot)$ is implemented by the three-layer fully connected layers. Finally, given the word embedding $W = \{\boldsymbol{w}_i \in \mathbb{R}^d \mid i = 1, \ldots, \|\mathcal{C}\|\}$, where $\mathcal{C}$ indicates the set of categories and d is the dimension of the word embedding. By minimizing the cosine distance from the domain-invariant features to the word embedding of the corresponding category, we can align the semantic space to the word embedding space. Therefore, we can recognize the unseen category by using the corresponding word embedding to characterize the semantic information of the unseen category.

Domain Combination

After semantic alignment, we develop the domain combination module by combining the aligned domain-invariant features and domain-specific features to reconstruct the domain itself and generate samples of different domains. As shown in Figure 3, we illustrate the detailed strcture of the domain combination module. Specifically, given the aligned domain-invariant features $\boldsymbol{E}_A$ and $\boldsymbol{E}_B$ of the 3D shape and sketch. To reconstruct the domain, we concatenate the aligned domain-invariant features with the corresponding domain-specific features, which is written as:

$$\boldsymbol{R}_A = \mathcal{R}([\boldsymbol{E}_A; \boldsymbol{S}_A]), \boldsymbol{R}_B = \mathcal{R}([\boldsymbol{E}_B; \boldsymbol{S}_B]) \tag{3}$$

where $\boldsymbol{R}_A \in \mathbb{R}^D$ and $\boldsymbol{R}_B \in \mathbb{R}^D$ are the reconstructed features of the 3D shape and sketch, respectively. $[\cdot;\cdot]$ is the concatenation operation. We perform the L_1 loss to minimize the distance from the reconstructed features ($\boldsymbol{R}_A$, $\boldsymbol{R}_B$) to the corresponding features ($\boldsymbol{F}_A$, $\boldsymbol{F}_B$). In addition, we combine the domain-invariant feature with another domain-specific feature to generate the sample of another domain, which is formulated as:

$$\boldsymbol{G}_A = \mathcal{G}([\boldsymbol{E}_B; \boldsymbol{S}_A]), \boldsymbol{G}_B = \mathcal{G}([\boldsymbol{E}_A; \boldsymbol{S}_B]) \tag{4}$$

where $\boldsymbol{G}_A \in \mathbb{R}^D$ and $\boldsymbol{G}_B \in \mathbb{R}^D$ are the generated features of 3D shape and sketch, respectively. After that, we formulate the adversarial loss by inputting the original feature ($\boldsymbol{F}_A$, $\boldsymbol{F}_B$) and the generated features ($\boldsymbol{G}_A$, $\boldsymbol{G}_B$) to the discriminators (D_A, D_B).

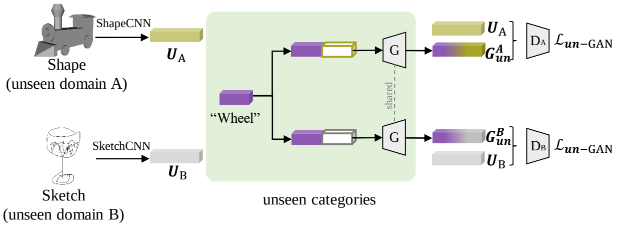

ShapeCNN
$\boldsymbol{U}_{\text{A}}$
Shape
(unseen domain A)
"Wheel"
shared
G
G
$\boldsymbol{U}_{\text{A}}$
$\boldsymbol{G}^{A}_{un}$
D_{A}
$\mathcal{L}_{un\text{-GAN}}$
SketchCNN
$\boldsymbol{U}_{\text{B}}$
Sketch
(unseen domain B)
$\boldsymbol{G}^{B}_{un}$
$\boldsymbol{U}_{\text{B}}$
D_{B}
$\mathcal{L}_{un\text{-GAN}}$
unseen categories

Figure 3: The generation of unseen samples.

Transductive setting. In order to reduce the domain shift between the generated unseen categories and the real unseen categroeis, we extend our DD-GAN to a transductive setting. As shown in Figure 3, we illustrate the details of the transductive setting. Given the 3D shape and sketch of the unseen categories, we first use the ShapeCNN and SketchCNN to extract the initial features $\boldsymbol{U}_A \in \mathbb{R}^D$ and $\boldsymbol{U}_B \in \mathbb{R}^D$.

Then, by combining the word embedding $\boldsymbol{w}_{un}$ of the unseen categories with the domain-specific features $\boldsymbol{S}_A$ and $\boldsymbol{S}_B$, we can generate samples of the unseen categories, which is formulated as:

$$\boldsymbol{G}_A^{un} = \mathcal{G}([\boldsymbol{w}_{un}; \boldsymbol{S}_A]), \boldsymbol{G}_B^{un} = \mathcal{G}([\boldsymbol{w}_{un}; \boldsymbol{S}_B]) \qquad (5)$$

where $\boldsymbol{G}_A^{un} \in \mathbb{R}^D$ and $\boldsymbol{G}_B^{un} \in \mathbb{R}^D$ are the generated samples of the unseen categories. Finally, we formulate the adversarial loss by minimizing the distance from the generated samples ($\boldsymbol{G}_A^{un}$, $\boldsymbol{G}_B^{un}$) to the initial features ($\boldsymbol{U}_A$, $\boldsymbol{U}_B$).

Network Training Strategy

Loss functions. In the semantic alignment module, the triple loss $\mathcal{L}_{tri}$ is used to enhance the discrimination of domain-invariant features, which is formulated as:

$$\mathcal{L}_{tri} = max\{\|\boldsymbol{I}_B - \boldsymbol{I}_A\|_2 - \|\boldsymbol{I}_B - \boldsymbol{I}_A\|_2 + \eta, 0\} \qquad (6)$$

where the $\eta > 0$ is the margin parameter. In addition, we use the semantic loss $\mathcal{L}_{sem}$ to align the domain-invariant features with corresponding word embedding. Cosine similarity is used to metric the similarity between the domain-invariant features and word embeddings, $\mathcal{L}_{sem}$ can be formulate as:

$$\mathcal{L}_{sem} = sim(E_A, \boldsymbol{w}_A) + sim(E_B, \boldsymbol{w}_A) + sim(E_C, \boldsymbol{w}_C) \qquad (7)$$

where the $sim(\cdot, \cdot)$ = $1 - cos(\cdot, \cdot)$, $\boldsymbol{w}_A$ is the word embeddings of anchor sketch $\boldsymbol{B}$ and positive 3D shapes $\boldsymbol{A}$, and $\boldsymbol{w}_C$ is the word embedding of negative 3D shape $\boldsymbol{C}$.

In the domain combination module, the reconstruction loss $\mathcal{L}_{rec}$ is formulated as:

$$\mathcal{L}_{rec} = \|\boldsymbol{R}_B - \boldsymbol{F}_B\|_1 + \|\boldsymbol{R}_A - \boldsymbol{F}_A\|_1 + \|\boldsymbol{R}_C - \boldsymbol{F}_C\|_1 \qquad (8)$$

Note that we also consider the negative samples in the reconstruction loss. The adversarial loss $\mathcal{L}_{GAN}$ is formulated as:

$$\begin{aligned} \mathcal{L}_{GAN} = & \mathbb{E}(\log(D_A(\boldsymbol{F}_A)) + \log(1 - D_A(\boldsymbol{G}_A)) \\ & + \mathbb{E}(\log(D_B(\boldsymbol{F}_B)) + \log(1 - D_B(\boldsymbol{G}_B)) \end{aligned} \qquad (9)$$

The features of unlabeled sketch $\boldsymbol{U}_B$ and 3D shape $\boldsymbol{U}_A$ can be extracted by ShapeCNN and SketchCNN, our DD-GAN can be trained with the loss function $\mathcal{L}_{un-GAN}$, which is computed as:

$$\begin{aligned} \mathcal{L}_{un-GAN} = & \mathbb{E}(\log(D_A(\boldsymbol{U}_B)) + \log(1 - D_A(\boldsymbol{G}_{un}^A)) \\ & + \mathbb{E}(\log(D_B(\boldsymbol{U}_A)) + \log(1 - D_B(\boldsymbol{G}_{un}^B)) \end{aligned} \qquad (10)$$

We finally train our model with the following loss:

$$\begin{aligned} & \mathcal{L}_{total}(E_A, E_s^A, E_B, E_s^B, G, D_A, D_B) = \\ & \mathcal{L}_{tri} + \mathcal{L}_{sem} + \lambda_{rec}\mathcal{L}_{rec} + \mathcal{L}_{GAN} + \mathcal{L}_{un-GAN} \end{aligned} \qquad (11)$$

where the λ_{rec} is the weight to control the importance of cycle-consistent.

Retraining and inference phase. After DD-GAN generates unseen samples in the two domains of sketch and 3D shape, we use the real seen data and the generated unseen data to form a new training set to retrain our model. The Unseen branch is removed and does not participate in retraining. At this time, the feature disentanglement in our model is mainly used to minimize the domain discrepancy between the two domains. In the inference phase, we use the domain-invariant features of query sketches to search the domain-invariant features of 3D shapes.

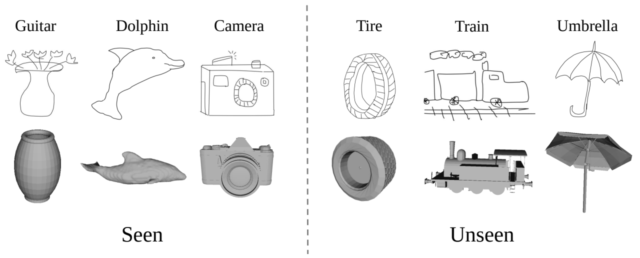


Figure 4: Examples of seen and unseen categories in the SHREC'14 dataset.

Experiment

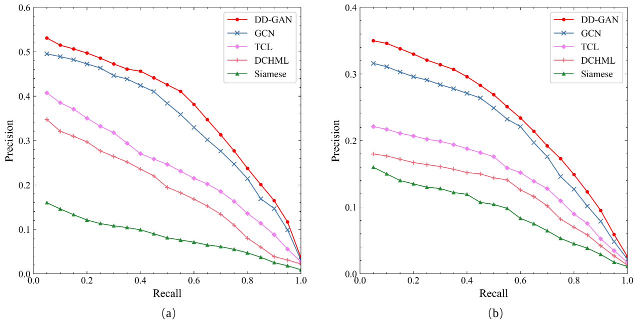
Experimental Settings

Benchmark datasets and evaluation. We evaluated our model on two widely used benchmarks, SHREC'13 (Li et al. 2013) and SHREC'14 (Li et al. 2014b), and compare with the state-of-the-art methods under the fair settings.

SHREC'13 is a benchmark for evaluate sketch-based 3D shape retrieval algorithms, which is created based on Princeton shape benchmark (Shilane et al. 2004) and human sketch dataset. There are a total of 90 categories in the dataset, including 7,200 hand-drawn sketches and 1,258 3D shapes. Each class contains 80 sketches, but the number of 3D shapes is not equal. The conventional sketch-based 3D retrieval methods divide each class of sketch into 50 for training and 30 for testing. In this paper, the training set and test set are divided according to the category, 79 classes for training (seen) and 11 classes for testing (unseen).

SHREC'14 is larger than SHREC'13, which contains 13,680 hand-drawn sketches and 8,987 3D shapes. Similar to SHREC'13, this dataset also contains 80 sketches for each category, 50 for training and 30 for testing. We also redivide the training set and the test set, 151 classes for training (seen) and 20 classes for testing (unseen). As shown in Figure 4, we visualize some examples of seen categories and unseen categories in the SHREC'14 dataset.

Implementation details. We implemented our DD-GAN using PyTorch. ResNet-50 (He et al. 2016) pre-trained on ImageNet is used as backbone of SketchNet and ShapeNet. E_i and E_s have the same architecture with three fully-connected layers followed by two LeakyReLU, which output 300-D domain-invariant features and 300-D domain-specific features respectively. We use the word text-based embedding model (Pennington, Socher, and Manning 2014) to extract 300-D word embeddings. The generator G is a multi-layer perceptron containing three fully-connected layers generate the feature vectors of 2048-D. The sketch domain and 3D shape domain discriminators D_A and D_B also share the same architecture with three fully-connected. The margin η in triplet loss is 20 and the weight λ_{rec} of $\mathcal{L}_{rec}$ is 10 in this paper. We adopt Adam to optimize our model with the learning rate of $1e^{-5}$.



(a)

(b)

Figure 5: Performance comparison of precision-recall curve on the SHREC’13 dataset (a) and the SHREC’14 dataset (b).

Table 1: Zero-shot retrieval results on the SHREC’13 dataset

Method	NN	FT	ST	E	DCG	mAP
Siamese	0.137	0.114	0.203	0.162	0.404	0.171
DCHML	0.318	0.304	0.421	0.288	0.581	0.361
TCL	0.337	0.357	0.537	0.278	0.589	0.426
CGN	0.512	0.458	0.647	0.347	0.673	0.515
Baseline	0.201	0.195	0.324	0.194	0.516	0.231
DD-GAN	**0.544**	**0.484**	**0.661**	**0.364**	**0.696**	**0.551**

Zero-Shot Sketch-Based 3D Shape Retrieval Results

Retrieval performance on SHREC’13. We compared our method with Siamese (Wang, Kang, and Li 2015), DCHML (Dai, Xie, and Fang 2018), TCL (He et al. 2018) and CGN(Dai and Liang 2020) on the SHREC’13 dataset. And we utilize the following widely-adopt metrics to evaluate our proposed method: nearest neighbor (NN), first tier (FT), second tier (ST), E-measure (E), discounted cumulated gain (DCG) and mean average precision (mAP). For a fair comparison, we use the same backbone (ResNet-50) to extract the features of sketches and 3D shapes. In addition, we train all the above methods under our zero-shot data division to be the same as our DD-GAN. As we can see in Figure 5 (a), we apply precision-recall curve to compare our method with others, and our proposed method outperforms these methods. The comparison results are also shown in Table 1. We use the model as our baseline, which has an encoder with the same architecture as E_i and trained with the triple loss $\mathcal{L}_{tri}$. It can be seen that the performance of the proposed DD-GAN method is significantly better than these methods. Comparing with other methods, our model learns the knowledge from seen to unseen. In addition, the generated high-quality unseen samples also let the model learn the distribution of real unseen samples.

Retrieval performance on SHREC’14. We also evaluated our method on the SHREC’14 dataset and compared with Siamese (Wang, Kang, and Li 2015), DCHML (Dai, Xie, and Fang 2018), TCL (He et al. 2018) and CGN(Dai and Liang 2020). The comparison results of NN, FT, ST, E, DCG and mAP are shown in Table 2. And the precision-recall curve on SHREC’14 is shown in Figure 5 (b). Since the SHREC’14 dataset is larger than SHREC’13 and there are more categories, the retrieval task is also more difficult. The retrieval performance of all methods is greatly reduced compared to the performance on SHREC’13 dataset, but our method is still the best.

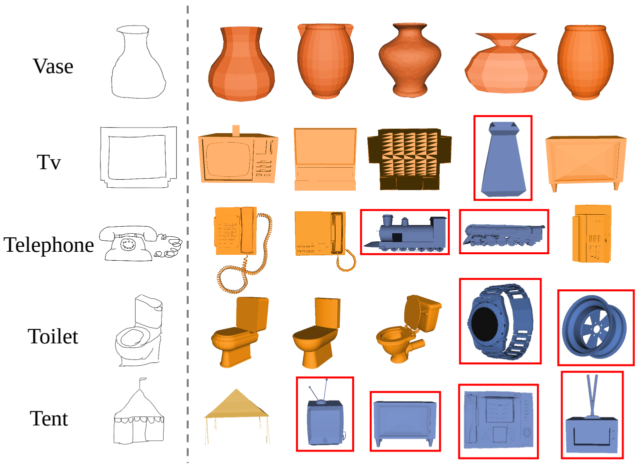


Figure 6: Top-5 zero-shot sketch-based 3D shape retrieval results obtained by our DD-GAN on the SHREC’13 dataset (top two rows) and the SHREC’14 dataset (next three rwos). The failure cases are marked by red rectangles.

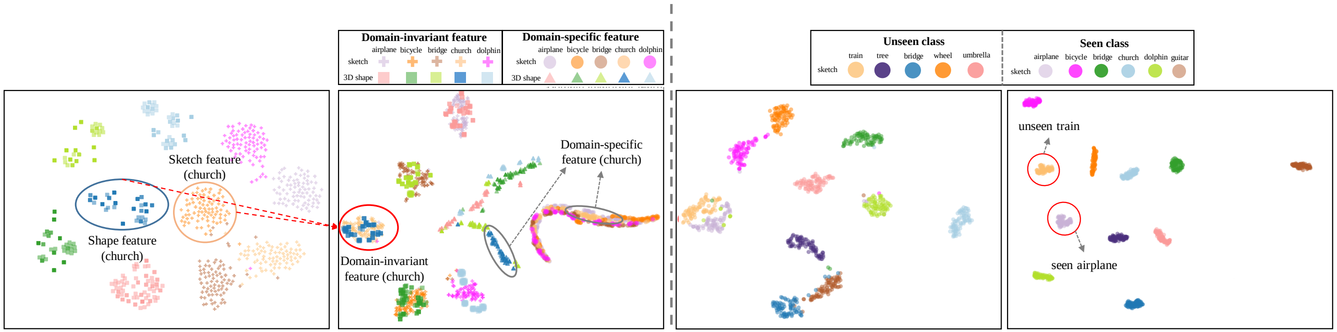
Table 2: Zero-shot retrieval results on the SHREC’14 dataset

Method	NN	FT	ST	E	DCG	mAP
Siamese	0.097	0.102	0.113	0.052	0.314	0.108
DCHML	0.157	0.134	0.145	0.084	0.379	0.187
TCL	0.279	0.257	0.153	0.125	0.459	0.237
CGN	0.401	0.324	0.429	0.178	0.571	0.332
Baseline	0.128	0.109	0.127	0.063	0.335	0.124
DD-GAN	**0.425**	**0.354**	**0.462**	**0.196**	**0.592**	**0.371**

Qualitative results. As shown in Figure 6, we also visualize some successful and failed retrieval results of our DD-GAN on SHREC13 and SHREC’14. For example, sketch query of “Telephone” retrieval some 3D shape of “Train” probably because the telephone receiver and train are both rectangular. For “Toilet” sketch, our model retrieves the “Wristwatch” and “Wheel” of 3D shape. Maybe the model captures that they all have circles. The “Tent” sketch and the “TV” 3D shape are also similar due to the rectangular structure. These wrongly retrieved 3D shapes usually have similar visual or semantics to the query sketches.

Ablation Study

Semantic alignment. We conduct the experiments on the SHREC’13 dataset to verify the effectiveness of our proposed model. It is worth noting that for the convenience of experiments, we use initial features extracted by SketchC-



(a) Triplet loss (b) Triplet loss and feature disentanglement (c) Without transductive setting (d) With transductive setting

Figure 7: t-SNE Visualization of different features. Comparing (a) and (b), after performing feature disentanglement, the distribution of the 3D shape and sketch becomes consistent. Comparing (c) and (d), after performing the transductive setting, the generated unseen samples can be better distinguished from the real visible samples.

Table 3: The performance on SHREC'13 dataset

Method	NN	FT	ST	E	DCG	mAP
Baseline	0.201	0.195	0.324	0.194	0.516	0.231
Baseline+SA	0.275	0.224	0.362	0.241	0.542	0.254
DD-GAN (O/T)	0.522	0.464	0.649	0.351	0.682	0.523
DD-GAN (W/T)	**0.544**	**0.484**	**0.661**	**0.364**	**0.696**	**0.551**

NN and ShapeCNN as input instead of sketches and 3D shapes in subsequent experiments. Here we add semantic loss $\mathcal{L}_{sem}$ to the baseline to demonstrate the effectiveness of the semantic alignment module. As shown in Table 3, semantic alignment module (Baseline + SA) improves the retrieval performance of the model on the unseen category. This can prove that our semantic alignment module can transfer knowledge from seen classes to unseen classes.

Domain disentangling. As shown in Figure 7, we use t-SNE (Van der Maaten and Hinton 2008) to visualize the features of sketches and 3D shapes from different five categories. In Figure 7 (a), we use our baseline with semantic loss $\mathcal{L}_{sem}$ to extract features of sketches and 3D shapes from the initial features (Baseline + SA). We can see that although the features of sketches and 3D shapes are discriminative, the same categories of sketch features and 3D shape features still have the domain gap. To prove the effectiveness of domain disentangling, we added domain disentangling and domain combination to the model (DD-GAN(O/T)) in Figure 7 (a). The reason why we add domain combination here is that it ensures that domain-specific features and domain-invariant features can be decoupled. As shown in Figure 7 (b), after disentangling domain-specific features, the domain-invariant features of the same class in the two domains can be effectively narrowed in the common feature space. For example, in Figure 7 (a), although the 3D shape features (blue circle) and sketch features (orange circle) of "Church" belong to the same category, they are still away from each other. After feature disentanglement, the domain-invariant features of "Church" in the two domains are close together (red circle), as shown in Figure 7 (b). In Table 3, the quantitative evaluation of the DD-GAN (O/T) is also much better than (Baseline + SA).

Transductive setting. We compare our full model DD-GAN (W/T) to its variants without transductive setting (DD-GAN (O/T)). We first generate unseen samples of sketch domain through the two models, which combine the unseen word embedding with domain-specific feature of seen class sketch. Then we apply t-SNE to visualize the generated samples together with real samples of seen classes in Figure 7. As shown in Figure 7 (c), the DD-GAN (O/T) has not seen the real unseen samples, that make the generated unseen samples are similar to the real seen categories. For example, the generated unseen samples of "Train" category is close to the real seen class "Airplane". This will reduce the quality of the generated unseen samples, resulting in little improvement in the model retraining effect. In Figure 7 (d), the unseen samples generated by DD-GAN (W/T) are obviously different from the real seen categories. In addition, as shown in Table 3, the retrieval performance of DD-GAN (W/T) is better than DD-GAN (O/T). This proves that we introduced unlabeled data in an unsupervised way, which effectively avoided the domain shift problem and improved the quality of the generated samples.

Conclusion

In this paper, we are the first to explore zero-shot sketch-based 3D shape retrieval. In order to make the model effectively retrieve unseen categories, we propose a domain disentangled generative adversarial network (DD-GAN). Our model can not only reduce the inter-domain difference between sketch and 3D shape, but also minimize the domain discrepancy between seen categories and unseen categories. Extensive experiments on the SHREC'13 dataset and SHREC'14 dataset can demonstrate the effectiveness of the proposed method for zero-shot sketch-based 3D retrieval. In the future work, we will consider exploring a more effective way to generate high-quality unseen samples in the zero-shot sketch based 3D shape retrieval task.

Acknowledgments

This work was supported by the National Science Foundation of China (Grant Nos. 61876084, 61876083, 62176124), and the Postgraduate Research & Practice Innovation Program of Jiangsu Province (No. KYCX21_0302).

References

Akata, Z.; Perronnin, F.; Harchaoui, Z.; and Schmid, C. 2015. Label-embedding for image classification. *IEEE transactions on pattern analysis and machine intelligence*, 38(7): 1425–1438.

Bucher, M.; Herbin, S.; and Jurie, F. 2017. Generating visual representations for zero-shot classification. In *ICCV Workshops*, 2666–2673.

Bucher, M.; Vu, T.-H.; Cord, M.; and Pérez, P. 2019. Zero-shot semantic segmentation. *Advances in Neural Information Processing Systems*, 32: 468–479.

Chen, J.; and Fang, Y. 2018. Deep cross-modality adaptation via semantics preserving adversarial learning for sketch-based 3D shape retrieval. In *ECCV*, 605–620.

Chen, J.; Qin, J.; Liu, L.; Zhu, F.; Shen, F.; Xie, J.; and Shao, L. 2019. Deep sketch-shape hashing with segmented 3D stochastic viewing. In *CVPR*, 791–800.

Dai, G.; Xie, J.; and Fang, Y. 2018. Deep correlated holistic metric learning for sketch-based 3D shape retrieval. *IEEE Transactions on Image Processing*, 27(7): 3374–3386.

Dai, G.; Xie, J.; Zhu, F.; and Fang, Y. 2017. Deep correlated metric learning for sketch-based 3D shape retrieval. In *AAAI*.

Dai, W.; and Liang, S. 2020. Cross-modal guidance network for sketch-based 3D shape retrieval. In *ICME*, 1–6. IEEE.

Deng, C.; Xu, X.; Wang, H.; Yang, M.; and Tao, D. 2020. Progressive cross-modal semantic network for zero-shot sketch-based image retrieval. *IEEE Transactions on Image Processing*, 29: 8892–8902.

Dey, S.; Riba, P.; Dutta, A.; Llados, J.; and Song, Y.-Z. 2019. Doodle to search: Practical zero-shot sketch-based image retrieval. In *CVPR*, 2179–2188.

Dutta, A.; and Akata, Z. 2019. Semantically tied paired cycle consistency for zero-shot sketch-based image retrieval. In *CVPR*, 5089–5098.

Dutta, T.; and Biswas, S. 2019. Style-Guided Zero-Shot Sketch-based Image Retrieval. In *BMVC*, volume 2, 9.

Eitz, M.; Richter, R.; Boubekeur, T.; Hildebrand, K.; and Alexa, M. 2012. Sketch-based shape retrieval. *ACM Transactions on graphics (TOG)*, 31(4): 1–10.

Furuya, T.; and Ohbuchi, R. 2013. Ranking on cross-domain manifold for sketch-based 3D model retrieval. In *Cyberworlds*, 274–281. IEEE.

Gao, R.; Hou, X.; Qin, J.; Chen, J.; Liu, L.; Zhu, F.; Zhang, Z.; and Shao, L. 2020. Zero-vae-gan: Generating unseen features for generalized and transductive zero-shot learning. *IEEE Transactions on Image Processing*, 29: 3665–3680.

Goldfeder, C.; and Allen, P. 2008. Autotagging to improve text search for 3D models. In *JCDL*, 355–358.

Han, Z.; Fu, Z.; Chen, S.; and Yang, J. 2021. Contrastive Embedding for Generalized Zero-Shot Learning. In *CVPR*, 2371–2381.

Han, Z.; Fu, Z.; and Yang, J. 2020. Learning the redundancy-free features for generalized zero-shot object recognition. In *CVPR*, 12865–12874.

He, K.; Zhang, X.; Ren, S.; and Sun, J. 2016. Deep residual learning for image recognition. In *CVPR*, 770–778.

He, X.; Zhou, Y.; Zhou, Z.; Bai, S.; and Bai, X. 2018. Triplet-center loss for multi-view 3D object retrieval. In *CVPR*, 1945–1954.

Huang, H.; Wang, C.; Yu, P. S.; and Wang, C.-D. 2019. Generative dual adversarial network for generalized zero-shot learning. In *CVPR*, 801–810.

Iyer, N.; Jayanti, S.; Lou, K.; Kalyanaraman, Y.; and Ramani, K. 2005. Three-dimensional shape searching: state-of-the-art review and future trends. *Computer-Aided Design*, 37(5): 509–530.

Jiang, J.; Bao, D.; Chen, Z.; Zhao, X.; and Gao, Y. 2019. M-LVCNN: Multi-loop-view convolutional neural network for 3D shape retrieval. In *AAAI*, volume 33, 8513–8520.

Kato, N.; Yamasaki, T.; and Aizawa, K. 2019. Zero-shot semantic segmentation via variational mapping. In *ICCV Workshops*, 0–0.

Kodirov, E.; Xiang, T.; and Gong, S. 2017. Semantic autoencoder for zero-shot learning. In *CVPR*, 3174–3183.

Kuwabara, S.; Ohbuchi, R.; and Furuya, T. 2019. Query by Partially-Drawn Sketches for 3D Shape Retrieval. In *Cyberworlds*, 69–76. IEEE.

Li, B.; and Johan, H. 2013. Sketch-based 3D model retrieval by incorporating 2D-3D alignment. *Multimedia Tools and Applications*, 65(3): 363–385.

Li, B.; Lu, Y.; Godil, A.; Schreck, T.; Aono, M.; Johan, H.; Saavedra, J. M.; and Tashiro, S. 2013. *SHREC13 track: large scale sketch-based 3D shape retrieval*.

Li, B.; Lu, Y.; Godil, A.; Schreck, T.; Bustos, B.; Ferreira, A.; Furuya, T.; Fonseca, M. J.; Johan, H.; Matsuda, T.; et al. 2014a. A comparison of methods for sketch-based 3D shape retrieval. *Computer Vision and Image Understanding*, 119: 57–80.

Li, B.; Lu, Y.; Johan, H.; and Fares, R. 2017a. Sketch-based 3D model retrieval utilizing adaptive view clustering and semantic information. *Multimedia Tools and Applications*, 76(24): 26603–26631.

Li, B.; Lu, Y.; Li, C.; Godil, A.; Schreck, T.; Aono, M.; Burtscher, M.; Fu, H.; Furuya, T.; Johan, H.; et al. 2014b. SHREC14 track: Extended large scale sketch-based 3D shape retrieval. In *3DOR*, volume 2014, 121–130.

Li, H.; Wu, H.; He, X.; Lin, S.; Wang, R.; and Luo, X. 2017b. Multi-view pairwise relationship learning for sketch based 3D shape retrieval. In *ICME*, 1434–1439. IEEE.

Liu, Q.; Xie, L.; Wang, H.; and Yuille, A. L. 2019. Semantic-aware knowledge preservation for zero-shot sketch-based image retrieval. In *ICCV*, 3662–3671.

Liu, Q.; and Zhao, S. 2021. Guidance Cleaning Network for Sketch-Based 3D Shape Retrieval. In *Journal of Physics: Conference Series*, volume 1961, 012072. IOP Publishing.

Mensink, T.; Gavves, E.; and Snoek, C. G. 2014. Costa: Co-occurrence statistics for zero-shot classification. In *CVPR*, 2441–2448.

Min, P.; Kazhdan, M.; and Funkhouser, T. 2004. A comparison of text and shape matching for retrieval of online 3D models. In *TPDL*, 209–220. Springer.

Pandey, A.; Mishra, A.; Verma, V. K.; Mittal, A.; and Murthy, H. 2020. Stacked adversarial network for zero-shot sketch based image retrieval. In *WACV*, 2540–2549.

Pennington, J.; Socher, R.; and Manning, C. D. 2014. Glove: Global vectors for word representation. In *EMNLP*, 1532–1543.

Qi, A.; Song, Y.-Z.; and Xiang, T. 2018. Semantic Embedding for Sketch-Based 3D Shape Retrieval. In *BMVC*, volume 3, 11–12.

Saavedra, J. M.; Bustos, B.; Schreck, T.; Yoon, S. M.; and Scherer, M. 2012. Sketch-based 3D Model Retrieval using Keyshapes for Global and Local Representation. In *3DOR*, 47–50.

Shilane, P.; Min, P.; Kazhdan, M.; and Funkhouser, T. 2004. The princeton shape benchmark. In *PSMA*, 167–178. IEEE.

Su, H.; Maji, S.; Kalogerakis, E.; and Learned-Miller, E. 2015. Multi-view convolutional neural networks for 3D shape recognition. In *ICCV*, 945–953.

Tasse, F. P.; and Dodgson, N. 2016. Shape2vec: semantic-based descriptors for 3D shapes, sketches and images. *ACM Transactions on graphics (TOG)*, 35(6): 1–12.

Van der Maaten, L.; and Hinton, G. 2008. Visualizing data using t-SNE. *Journal of machine learning research*, 9(11).

Wang, F.; Kang, L.; and Li, Y. 2015. Sketch-based 3D shape retrieval using convolutional neural networks. In *CVPR*, 1875–1883.

Xian, Y.; Akata, Z.; Sharma, G.; Nguyen, Q.; Hein, M.; and Schiele, B. 2016. Latent embeddings for zero-shot classification. In *CVPR*, 69–77.

Xian, Y.; Lampert, C. H.; Schiele, B.; and Akata, Z. 2018. Zero-shot learninga comprehensive evaluation of the good, the bad and the ugly. *IEEE transactions on pattern analysis and machine intelligence*, 41(9): 2251–2265.

Xian, Y.; Sharma, S.; Schiele, B.; and Akata, Z. 2019. f-vaegan-d2: A feature generating framework for any-shot learning. In *CVPR*, 10275–10284.

Xie, J.; Dai, G.; Zhu, F.; and Fang, Y. 2017. Learning barycentric representations of 3D shapes for sketch-based 3D shape retrieval. In *CVPR*, 5068–5076.

Xie, J.; Fang, Y.; Zhu, F.; and Wong, E. 2015. Deepshape: Deep learned shape descriptor for 3D shape matching and retrieval. In *CVPR*, 1275–1283.

Xu, X.; Deng, C.; Yang, M.; and Wang, H. 2020a. Progressive domain-independent feature decomposition network for zero-shot sketch-based image retrieval. *arXiv preprint arXiv:2003.09869*.

Xu, Y.; Hu, J.; Wattanachote, K.; Zeng, K.; and Gong, Y. 2020b. Sketch-based shape retrieval via best view selection and a cross-domain similarity measure. *IEEE Transactions on Multimedia*, 22(11): 2950–2962.

Yelamarthi, S. K.; Reddy, S. K.; Mishra, A.; and Mittal, A. 2018. A zero-shot framework for sketch based image retrieval. In *ECCV*, 300–317.

Yoon, G.-J.; and Yoon, S. M. 2017. Sketch-based 3D object recognition from locally optimized sparse features. *Neurocomputing*, 267: 556–563.

Zhu, F.; Xie, J.; and Fang, Y. 2016a. Heat diffusion long-short term memory learning for 3D shape analysis. In *ECCV*, 305–321. Springer.

Zhu, F.; Xie, J.; and Fang, Y. 2016b. Learning cross-domain neural networks for sketch-based 3D shape retrieval. In *AAAI*, volume 30.

Zhu, Z.; Wang, X.; Bai, S.; Yao, C.; and Bai, X. 2016. Deep learning representation using autoencoder for 3D shape retrieval. *Neurocomputing*, 204: 41–50.

Thank you for reading these instructions carefully. We look forward to receiving your electronic files!