Text-guided Synthetic Geometric Augmentation for Zero-shot 3D Understanding

Kohei Torimi[1,2], Ryosuke Yamada[2], Daichi Otsuka[2], Kensho Hara[2],
Yuki M. Asano[3], Hirokatsu Kataoka[2,4], Yoshimitsu Aoki[1]

[1]Keio University
[2]National Institute of Advanced Industrial Science and Technology (AIST)
[3]University of Technology Nuremberg
[4]University of Oxford
{kohei.torimi, ryosuke.yamada, ootsuka.da, kensho.hara}@aist.go.jp,
yuki.asano@utn.de, hirokatsu.kataoka@aist.go.jp, aoki@elec.keio.ac.jp

Abstract

*Zero-shot recognition models require extensive training data for generalization. However, in zero-shot 3D classification, collecting 3D data and captions is costly and labor-intensive, posing a significant barrier compared to 2D vision. Recent advances in generative models have achieved unprecedented realism in synthetic data production, and recent research shows the potential for using generated data as training data. Here, naturally raising the question: Can synthetic 3D data generated by generative models be used as expanding limited 3D datasets? In response, we present a synthetic 3D dataset expansion method, **Te**xt-guided **G**eometric **A**ugmentation (**TeGA**). TeGA is tailored for language-image-3D pretraining, which achieves SoTA in zero-shot 3D classification, and uses a generative text-to-3D model to enhance and extend limited 3D datasets. Specifically, we automatically generate text-guided synthetic 3D data and introduce a consistency filtering strategy to discard noisy samples where semantics and geometric shapes do not match with text. In the experiment to double the original dataset size using TeGA, our approach demonstrates improvements over the baselines, achieving zero-shot performance gains of 3.0% on Objaverse-LVIS, 4.6% on ScanObjectNN, and 8.7% on ModelNet40. These results demonstrate that TeGA effectively bridges the 3D data gap, enabling robust zero-shot 3D classification even with limited real training data.*

1. Introduction

Zero-shot recognition models have made remarkable progress leveraging paired image-text data. In particular, CLIP [30] has shown the strong potential such models offer for zero-shot open-vocabulary image classification. Highly accurate zero-shot recognition models are increasingly applied in industrial applications too, such as robotics, manufacturing, and autonomous driving. Yet, this progress is so far mostly limited to the 2D domain. We argue that multi-modal representation learning that leverages not only images and text but also 3D data is critical for zero-shot 3d classification tasks.

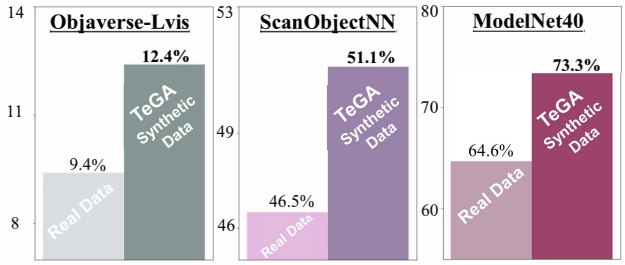

Figure 1. Our proposed TeGA (Text-guided Geometric Augmentation) assigns text guidance and a generative text-to-3D model for high-efficient dataset expansion which dramatically augments limited real data. Although we employ existing methods (*e.g.,* Point-E) and simple tricks within text prompting, the proposal performs enough noteworthy results that 3D dataset with add synthetic data with TeGA outperforms ShapeNet trained model on e.g., Objaverse-LVIS, ModenNet-40 and ScanObjectNN under the setting of zero-shot 3D classification.

One of the reasons for the success of vision-language models such as CLIP lies in the vast scale of training data utilized. These range from 400 million [30] to several billions of image-text pairs collected from the Web [34]. However, collecting 3D data is much more challenging and costly due to the scarcity of high-quality 3D data on the

Web. This is because the primary approaches require capturing 3D data in real-world environments with LiDAR or manually creating CAD models. Consequently, zero-shot recognition in 3D vision has lagged behind progress in 2D vision, and the limited availability of 3D datasets remains a bottleneck issue. To address this challenge, recent research on zero-shot 3D classification mainly focuses on distilling knowledge from a large-scale pre-trained visual language model. However, despite these efforts, the knowledge distillation does not fundamentally fix the issue of the limited availability of 3D data.

To this end, we propose to use recent text-to-3D generative models. These can produce highly realistic 3D synthetic data that are almost indistinguishable from the training data [16, 20, 22, 26, 28]. This remarkable evolution of the text-to-3D model simply begs the question: Can synthetic 3D data generated from text-to-3D models be used as expanding limited 3D training datasets?

This paper thus proposes a dataset expansion method for zero-shot 3D recognition called **Te**xt-guided **G**eometric **A**ugmentation (**TeGA**). The proposed TeGA expands the training language-image-3D dataset for models that perform knowledge distillation from visual language models and enables reducing the costs associated with 3D data collection and annotation. Specifically, we automatically generate synthetic 3D data and rendered images using an off-the-shelf text-to-3D model (*e.g.,* Point-E [26]) and use the latent space of the generative model to generate diverse geometric shapes with the same semantic content based on text prompts. Furthermore, TeGA also introduces a consistency filtering strategy to remove noisy data that does not match the text prompt, *i.e.,* misalignment between language, images, and 3D data.

To verify the effectiveness of TeGA, we train ModelNet40 [10] with ShapeNet and the synthetic dataset generated by TeGA and perform experiments on three representative zero-shot 3D classification datasets, Objaverse-LVIS [7], ScanObjectNN [38] and ModelNet40 [40] As a result, we show that TeGA improves performance in zero-shot 3D classification across three benchmark datasets. Specifically, TeGA achieves 12.4% on Objaverse-LVIS, 51.1% on ModelNet40, 73.3% on ScanObjectNN, and as shown in Figure 1. These results suggest that combining real and synthetic data improves the generalisation of visual models. We also found that consistency filtering plays important roles in learning. Our contributions in this paper are as follows:

- We introduce TeGA, a method for dataset expansion that leverages text-guided generative models to tackle the problem of 3D data scarcity. TeGA enables us to automatically generate synthetic language-image-3D data tailored to specific semantics category of original 3D dataset by introducing text-guided prompt and consistency filtering.
- TeGA enhances zero-shot 3D classification by expanding existing datasets with synthetic 3D data generated from a generative model, addressing the challenges of data scarcity in 3D vision tasks.

2. Related Work

Multi-Modal Representation Learning. In zero-shot 3D classification, recent research has focused on multi-modal learning with limited 3D data by distilling CLIP knowledge [13, 14, 21, 29, 41] because CLIP possesses extensive knowledge in vision-language tasks. For example, MixCon3D [10] improved the performance of zero-shot 3D classification by contrastive learning across the image-text-point cloud using CLIP knowledge. In addition, ULIP-2 [41] is a multimodal pre-training that automatically generates comprehensive language descriptions for 3D shapes without manual annotation. These conventional methods achieve high zero-shot performance on ModelNet40 [40], ScanObjectNN [38], and Objaverse-LVIS [7]. However, these methods basically focus only on improving training methods, and there are limits to how much performance can be improved with this approach alone. We believe it is essential for the model and data sides to evolve together. This paper thus explores the potential of synthetic dataset expansion to achieve scalable geometric learning for zero-shot 3D classification.

Text-to-3D Models. Recent text-to-image models have experienced rapid growth. Inspired by this, text-to-3D models have also become a prominent area of research. Text-to-3D models also face the challenge of limited 3D data, making it difficult to generate open-vocabulary 3D data directly from text. To address this, recent text-to-3D models employ a strategy that leverages text-to-2D models or CLIP's extensive language and image knowledge. They first generate images or image features and then lift these images to 3D data [4, 20, 28, 37, 39]. For example, DreamFusion [28] efficiently generates 3D data using CLIP's knowledge and NeRF's gradient updates. In addition, Zero123-XL [22] trains on the relatively large 3D dataset called Objaverse-XL [8] and achieves highly accurate 3D generation. Recently, it has also been applied to explicit 3D representations such as point clouds and meshes [16, 26]. Using a text-to-image and image-to-point cloud pipeline, Point-E [26] quickly generates point clouds that correspond to text prompts. In this paper, we utilize text-to-3D models for dataset expansion.

Generative Models as Data++. Recent generative models have evolved to generate realistic synthetic data that is visually almost indistinguishable [25, 31, 32]. And then the data generated from generative models can be thought of as data for recognition models with controllability and rich representation as data++ [15, 33]. It reduces the col-

lecting cost with real-world data while enabling the efficient generation of meaningful synthetic datasets. For example, StableRep [36] learns visual representations by incorporating synthetic images generated by Stable Diffusion [31] into multi-positive contrastive learning. StableRep is designed to learn images comprehensively across different views by treating multiple synthetic images of the same input text prompt as positive samples. In addition, SynCLR [35] performs contrastive learning using only synthetic images and captions. Despite training without real images, it performs equally well and better than traditional self-supervised learning such as DINOv2 [27] and OpenCLIP [5] in image classification and semantic segmentation. These studies have shown that it is possible to build high performance recognition models while training only synthetic data and keeping data collection costs low by learning visual representations. Inspired by these studies, we utilize data generated by generative models for contrastive learning.

3. Problem Setting and Preliminary

Inspired by the effectiveness of synthetic data as training data in 2D, we propose leveraging synthetic data as a solution to the scarcity of 3D data by using it to augment datasets. Specifically, we propose the method TeGA, that utilizes generative text-to-3D models to create datasets for language-image-3D pretraining. Therefore, in this section, we introduce the proposed TeGA as a synthetic dataset expansion method, capable of learning rich geometric representations to enhance zero-shot recognition model generalization. Finally, we outline the formulation of a language-image-3D learning method, aiming to learn feature spaces for embedding alignment across three different modalities.

Problem Setting. Zero-shot tasks generally demand large datasets to generalize to unseen classes [30], yet, in zero-shot 3D classification, the cost of collecting and annotating 3D data is a significant bottleneck and the performance of the models remains low [10, 21, 42]. Recently, synthetic dataset expansion has achieved notable performance improvements in several fields [15, 33]. Moreover, recent approaches in zero-shot 3D classification have demonstrated that pretraining on language-image-3D data enables knowledge distillation from models pretrained on other modalities, thereby improving performance under limited data conditions [13, 14, 21, 29, 41]. We consider these findings critical for improving the accuracy of zero-shot 3D classification. Hence, we present a multi-modal synthetic dataset expansion method called TeGA that generates synthetic language-image-3D data from text prompts using text-to-3D models.

To frame the zero-shot 3D classification, we provide an overview of the dataset setup. We begin by training a model $f(\theta)$ to handle multiple types of visual inputs from a source dataset $D_o = \{(x_i^I, x_i^T, x_i^P)\}_{i=1}^{n_o}$, which consists of image (x_i^I), text (x_i^T), and 3D point cloud (x_i^P) modalities. The goal of this task is to classify N semantic classes in a target dataset $D_t = \{(x_j^I, x_j^T, x_j^P)\}_{j=1}^{n_t}$. Here, n_o and n_t indicate the number of samples, emphasizing the model's ability to generalize to previously unknown classes.

The core of dataset expansion lies in extending the source dataset with a synthetic dataset, where $D_s = \{(x_k^I, x_k^T, x_k^P)\}_{k=1}^{n_s}$, to enhance the model $f(\theta)$'s ability to generalize on unseen category shapes. The key to the task is to design D_s to strengthen the adaptability of $f(\theta)$ to new concepts. The construction of this extended dataset, $D_o \cup D_s$, enriches the model's ability to recognize unseen category shapes.

Text-to-3D Models as Dataset Generator. Cognitive science suggests that people use past experience to recognize unfamiliar objects [1, 17]. For example, children use memories of other toys to imagine new ways to play with a new toy. This is a fundamental insight for the use of generative models as training data. The use of synthetic data generated from generative models for zero-shot recognition is very similar to the process by which humans recognize new concepts of objects from previous knowledge. Recent advancements in text-to-3D models enable the generation of realistic 3D synthetic data that is both rich and semantically meaningful [4, 20, 26, 28]. Unlike traditional generative models such as GANs [11] and VAEs [18], text-to-3D models accept direct text prompts, allowing for user-specified, scenario-driven 3D shape generation. Therefore, this paper aims to improve the generalization ability of the model $f(\theta)$ by constructing a synthetic dataset through a text-to-3D model for dataset expansion.

Inspired by the recent success of synthetic data training using generative models based on diffusion models, we adopt a diffusion-based generative model as our text-to-3D model. Generally, text-to-3D models based on diffusion models can be simply represented as follows:

$$P = G_\theta\big(T, \omega\big). \tag{1}$$

where $G_\theta\big(\cdot\big)$ represents the generator that inputs text T and outputs a point cloud T, and ω represents the guidance scale that is used to adjust how strongly the input text is reflected in the point cloud generation process. In this paper, we utilize a generative text-to-3D model $G_\theta(\cdot)$ as generating a point cloud x_i^P from a text x_i^T.

Language-Image-3D Contrastive Learning. The goal of language-image-3D contrastive learning is to align the 3D embeddings with the rich, pre-trained feature spaces of images and text to facilitate the classification of 3D data including unseen classes. This facilitation enables language-image-3D contrastive learning to mitigate the impact of the 3D data scarcity problem. In fact, models leveraging language-image-3D contrastive learning outperform mod-

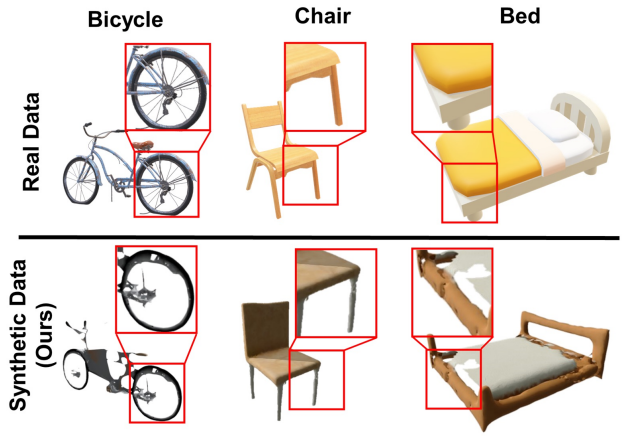

Figure 2. A visualization of synthetic 3D data generated from real 3D data and Point-E. Synthetic 3D data shows that it is more difficult to generate detailed geometrical detail compared to real data.

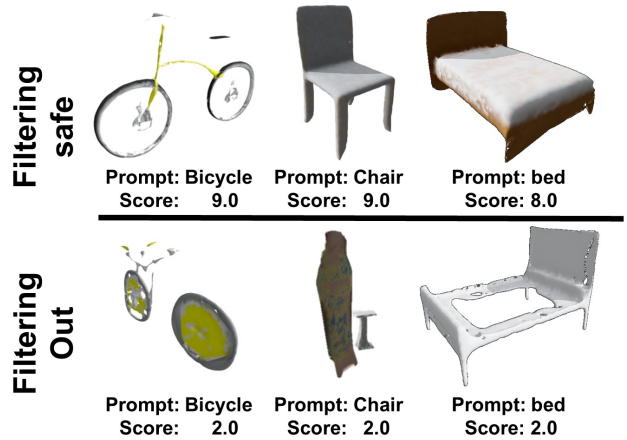

Figure 3. A visualization of consistency filtering. The upper shows samples which passed filter; the lower shows samples which filtered out. Our filtering can detect error cases while generation process.

els trained with only text and 3D data in zero-shot 3D classification. Given the typically limited 3D datasets, a practical approach is to leverage CLIP's knowledge as a shared embedding space. In this setup, we freeze CLIP's image and text encoders and align the 3D point cloud encoder to this shared space using contrastive learning. For a triplet of image, text, and point cloud (x_i^I, x_i^T, x_i^P), the contrastive objective maximizes similarity within the shared embedding space as follows:

$$L_{\text{All}} = -\frac{1}{2N}\sum_{i=1}^{N}\sum_{(A,B)\in\mathcal{S}}\left(\log\frac{\exp(h_i^A\cdot h_i^B/\tau)}{\sum_j \exp(h_i^A\cdot h_j^B/\tau)} + \log\frac{\exp(h_i^B\cdot h_i^A/\tau)}{\sum_j \exp(h_i^B\cdot h_j^A/\tau)}\right) \tag{2}$$

where $\mathcal{S} = \{(I,T),(P,I),(P,T)\}$ represents the pairs across modalities; in other words, (I,T) denotes the image-text pair, (P,I) the point cloud-image pair, and (P,T) the point cloud-text pair. In addition, the normalized features are defined as $h_i^A = g_A(f_A(x_i^A))/\|g_A(f_A(x_i^A))\|$, where f_A and g_A denote the encoder and learnable projection head for each modality A. Similarly, h_j^B is defined for modality B, with $(A,B)\in\mathcal{S}$ representing pairs across image, text, and point cloud modalities. The temperature parameter τ is a learnable parameter. It controls the strength of the penalty for samples with high similarity. Through this contrastive objective, we enable 3D shapes to align within CLIP's embedding space, facilitating a coherent language-image-3D representation.

4. Proposed Method: TeGA

In this section, we introduce TeGA, expanding language-image-3D datasets with high-quality synthetic 3D data generated from text-to-3D models.

Overview of TeGA. To tackle the problem of 3D data scarcity in zero-shot 3D classification, the proposed method automatically generates a synthetic dataset composing language, image, and 3D modalities using a text-to-3D model. In detail, to compose an expansion dataset, the proposed method combines the desired text, the point cloud generated by the text-to-3D model, and the images rendered from the generated point cloud. The proposed method can generate large amounts of data without requiring human data collection or annotation and expand real datasets. However, the generation process may err in some cases due to failure to generate or render, meaning that the alignment between each modality may not be accurate. This error may cause the model to collapse during training. To address this problem, we introduce a consistency filtering strategy of TeGA and the quality of the expanded dataset is improved by filtering low-quality data.

Synthetic Dataset Construction. Our goal is to generate a synthetic dataset $D_t = \{(x_i^I, x0_i^{T\prime}, x_i^{P\prime})\}_{i=1}^{n_o}$ consisting of text x_i^T, images $x_i^{I\prime}$, and point clouds $x_i^{P\prime}$ using the text x_i^T from the real dataset $D_s = \{(x_i^I, x_i^T, x_i^P)\}_{i=1}^{n_s}$ that also comprises text x_i^T, images x_i^I, and point clouds x_i^P. We use Point-E [26] as the text-to-3D model of TeGA because Point-E is suitable for generating large amounts of data due to its ability to rapidly generate point clouds. Also, we utilized only ShapeNet as the real dataset before applying TeGA instead of several datasets because our goal is not to outperform state-of-the-art methods but to serve as a training dataset expansion. ShapeNet is a dataset containing

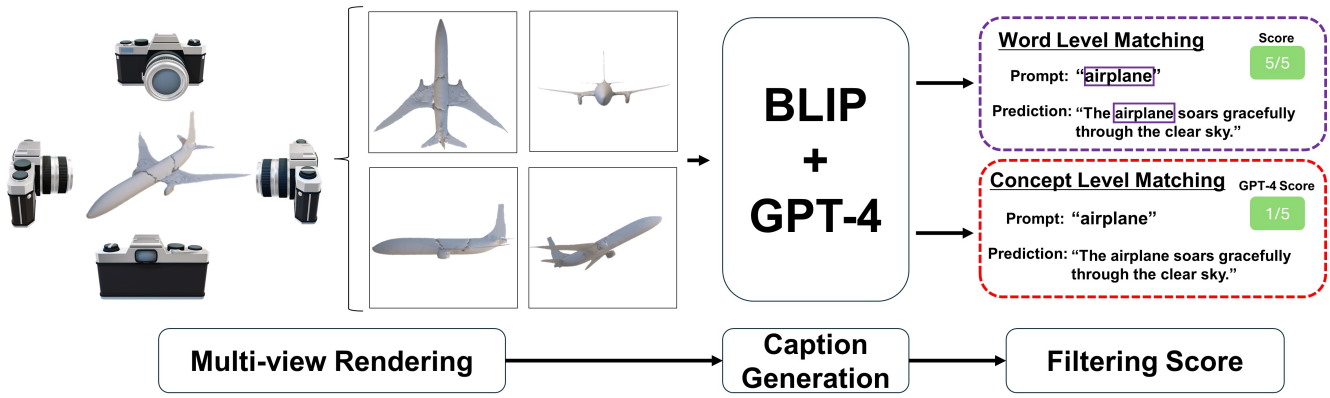

Figure 4. The overview of consistency filtering process. The purpose of this process is to remove misaligned data that may introduce model collapse during training. Specifically, rendered multi-view images are input into BLIP to generate captions. Then, the captions are summarized to one caption by GPT-4. Finally, the quality of the generated data is evaluated by comparing the text used for generation with the generated captions through two matching methods: word-level matching and concept-level matching.

52,470 point cloud samples across 55 classes.

We first generate a point cloud by inputting the category names x_i^T of the real dataset D_s as text prompts into Point-E. The generation process is represented as follows:

$$x_i^{P\prime} = G_\theta\left(x_i^T, \omega\right) \tag{3}$$

We have the point clouds and text, so we render images from the point clouds. However, without mesh information, the rendered images may appear to lack accurate shape details. To address this, we perform meshing using the Ball Pivoting Algorithm [2]. While various meshing techniques exist, Point-E's point clouds are not designed for meshing, and, in our experience, using state-of-the-art meshing methods often leads to the collapse of the rendered 3D data. Therefore, we adopted the older meshing technique, the Ball Pivoting Algorithm. Then, images $x_i^{I\prime}$ are generated by rendering the 3D data from 20 viewpoints. The viewpoints are determined by rotating it clockwise in 18-degree increments, with focal lengths dynamically determined by the Open3D library. The processes of meshing and rendering are represented as follows:

$$x_i^{I\prime} = \mathcal{R}(\mathcal{M}(x_i^{P\prime})) \tag{4}$$

where $\mathcal{M}(\cdot)$ denotes meshing process and $\mathcal{R}(\cdot)$ denotes rendering process. The sets of inputted text x_i^T, rendered images $x_i^{I\prime}$, and generated point clouds $x_i^{P\prime}$ that pass the consistency filtering are adopted as the synthetic dataset D_t.

Figure 2 shows examples of synthetic and real 3D data generated by Point-E. The real 3D data is a sample from Objaverse-LVIS. Figure 2 shows that the synthetic 3D data generated by Point-E outputs a 3D shape that matches the input text. On the other hand, there are cases where even the smallest details of the 3D shape are not generated accurately. For example, the synthetic 3D data generated by Point-E has difficulty in generating even the spokes of a bicycle.

Consistency Filtering. Alignment across modalities is crucial for language-image-3D pretraining. In our generation process, alignment issues may occur at two stages: generating point clouds from text using Point-E and rendering images from point clouds. Failures in point cloud generation often result from Point-E's misinterpretation of the text. Also, rendering failures occur when the point cloud is unsuitable for meshing, such as when parts of the point cloud are incomplete. Training a model with such misaligned data could lead to a breakdown of modality alignment within the model. To address this, we apply consistency filtering to each generated data sample.

If alignment is maintained between the input text x_i^T and the final output image $x_i^{I\prime}$ in the data generation process, it can be inferred that the intermediate output, the point cloud $x_i^{P\prime}$, is also aligned. This ensures that the alignment of the entire generated dataset is verified. Therefore, we evaluate the alignment between the text and images. Based on existing evaluation methods for 3D data [12], our filtering compares the integrated text generated from captions of multi-view images with the input text using GPT-4 and algorithm-based approaches. The process is shown in Figure 4.

For each synthetic data, we first select two images showing the front and back of the 3D data among the generated images during the dataset creation process. Next, we input the two images into BLIP [19] to generate each caption. To simplify, the generated captions for all viewpoints are combined into one unified text y_i using GPT-4. The combined caption is then compared with the text used for generation and evaluates whether the synthetic data is aligned or not. The evaluation is conducted using two metrics: a text-based consistency metric and a semantic alignment metric. For the text-based consistency metric, we then compute a

Table 1. Accuracy with and without filtering.

Filtering	O-LVIS Top1	S-Object Top1	M40 Top1
w / o filtering	11.1	45.4	**73.4**
w / filtering	**11.5**	**48.7**	71.3

Table 2. Accuracy when varying Guidance scale.

Guidance scale	O-LVIS Top1	S-Object Top1	M40 Top1
0.3	11.4	44.8	70.5
3.0	**11.5**	46.6	**71.1**
30	10.3	**48.3**	69.7

match score between this shared caption and the original text query. Specifically, if the generated caption y_i contains the text x_i^T, a score of 5 is assigned; if it does not, a score of 1 is assigned.

$$\mathrm{s}_{\text{text}}(y_i, x_i^T) = \begin{cases} 5, & \text{if } x_i^T \subseteq y_i, \\ 1, & \text{otherwise.} \end{cases} \tag{5}$$

For the semantic alignment metric, inspired by the alignment evaluation in T3Bench [12], we use GPT-4 to evaluate the semantic alignment of the generated caption relative to the original prompt. Specifically, the common captions and text prompts generated from the two images are used to generate a five-point semantic-based similarity score using GPT-4.

$$\mathrm{s}_{\text{sem}}(y_i, x_i^T) = \mathrm{GPT4}(y_i, x_i^T) \in \{1, 2, 3, 4, 5\}. \tag{6}$$

For more information about the GPT-4 prompts, see Appendix A.2.

The two calculated scores $\mathrm{s}_{\text{text}}, \mathrm{s}_{\text{sem}}$ are summed to obtain the final score s.

$$\mathrm{s}(y_i, x_i^T) = \mathrm{s}_{\text{text}}(y_i, x_i^T) + \mathrm{s}_{\text{sem}}(y_i, x_i^T) \tag{7}$$

If this score $\mathrm{s}(y_i, x_i^T)$ exceeds the threshold δ, the data is considered aligned and included in the dataset; otherwise, it is discarded. We experimentally validated this threshold δ during data generation with ShapeNet and finally set it to 3.5 for use in the dataset. Filtering with this threshold in ShapeNet resulted in approximately half of the data being filtered out. Figure 3 shows samples of our consistency filtering. The filtered data has well-formed meshes, making it visually easy to identify the objects. In contrast, the data that failed the filtering process often has collapsed meshes or represents unrecognizable objects.

5. Experiments

In this section, we evaluate the effectiveness of the proposed TeGA in zero-shot 3D classification. Section 5.1 first introduces our experimental setup, followed by Section 5.2, which presents the results of the exploratory experiments. Additionally, Section 5.3 discusses the main experimental results in comparison with our baseline and previous zero-shot 3D classification models. Finally, Section 5.4 analyzes the key components of TeGA.

5.1. Experimental Setup

Training Dataset. Recent research has achieved high performance in zero-shot 3D classification using a 3D dataset that integrates four sources: ShapeNet [3], 3D-FUTURE [9], ABO [6], and Objaverse [7]. However, our main goal is not to outperform state-of-the-art methods, but to test whether synthetic 3D data generated from text-to-3D models can serve as a training dataset expansion. For this reason, we use ShapeNet as a baseline in this paper.

Zero-Shot 3D Classification. Zero-shot 3D classification is the task of classifying objects of unseen categories that are not included in the training data. Following the experimental setup of MixCon3D, we use ModelNet40 [40], ScanObjectNN [38], and Objaverse-LVIS [7] as evaluation 3D datasets. ModelNet40 is a CAD dataset containing 12,311 samples of 40 categories, including chairs and tables. ScanObjectNN is captured from real-world environments using RGBD sensors and contains 2,902 samples of 15 categories. Objaverse-LVIS is a relatively large dataset with a collection of 1,156 categories, comprising a total of 46,206 samples.

Implementation Details. We used eight Nvidia H100 GPUs with a batch size of 1,024. Other settings followed the default configurations of MixCon3D. We also employed PointBERT as the point cloud encoder in MixCon3D. We trained the model for 200 epochs with the AdamW optimizer [23], a warmup epoch of 10, and a cosine learning rate decay schedule [24]. The base learning rate was set to 1e-3, based on the linear learning rate scaling rule: $lr = base_lr \times \text{batchsize}/256$. The Text Encoder and the Image Encoder are frozen using OpenCLIP ViT-bigG-14 [5], and, as in Liu et al. [21], simple layers are added afterward to optimize the model. For dataset generation, we used five Nvidia TITAN RTX GPUs. The generation parameters followed the default settings of Point-E: the output point cloud size was 4,096, the guidance scale ω was set to [3.0, 0.0], and 50 diffusion steps were performed to obtain the final output.

5.2. Exploratory Experiments

This section analyzes the effect of key parameters in TeGA. Specifically, we conducted two exploratory experiments: (i) consistency filtering and (ii) guidance scale on Point-E. For each exploratory experimental baseline, we evaluated the performance of MixCon3D with a combined dataset of

Table 3. Comparison with zero-shot 3D classification performance on three representative benchmark datasets. Best scores at MixCon3D are shown in underlined bold.

Method	Training data	Objaverse-LVIS				ScanObjectNN				ModelNet40			
		Top1	Top1-C	Top3	Top5	Top1	Top1-C	Top3	Top5	Top1	Top1-C	Top3	Top5
MixCon3D	ShapeNet	9.4	5.7	16.7	20.4	46.5	41.6	67.9	79.6	64.6	60.3	83.0	87.1
MixCon3D **(Ours)**	ShapeNet + TeGA	**12.4**	**10.8**	**22.0**	**27.4**	**51.1**	**49.5**	**69.7**	**80.2**	**73.3**	**68.7**	**88.8**	**93.6**

10,000 synthetic data samples and ShapeNet.

Consistency Filtering (see Table 1). The consistency filtering of the proposed TeGA is an important process because it removes the unaligned data from the generated data that adversely affects training. To verify the effectiveness of TeGA, we conducted a comparative experiment under two scenarios: one with consistency filtering applied and one without it. For this experiment, we use two versions of our synthetic dataset: one filtered by TeGA and one unfiltered, and combine each with the ShapeNet dataset. Then, these combined datasets are used for training. In Table 1, we compare the Top1 accuracy with and without consistency filtering on Objaverse-LVIS, ScanObjectNN, and ModelNet40. The results show that the concatenated dataset with consistency filtering outperforms the unfiltered one by 0.4% points on Objaverse-LVIS and 3.3% points on ScanObjectNN, while it decreases 2.1% points on ModelNet40. These results indicate that the proposed consistency filtering of TeGA provides effective synthetic training for zero-shot 3D classification.

Effect of Guidance Scale of Point-E (see Table 2). The guidance scale of Point-E is a critical parameter that controls the fidelity of the 3D model to the input text prompt. With a high guidance scale, the 3D model adheres more closely to the text prompt, though this reduces the diversity of geometric shapes. Conversely, a low guidance scale results in a 3D model that is less aligned with the text prompt but exhibits greater geometric diversity. Referring to the fact that the default guidance scale in Point-E is set to 3.0, this experiment evaluates varying guidance scale in $\{0.3, 3.0, 30\}$.

In Table 1, we compare the Top1 accuracy with each guidance scale on Objaverse-LVIS, ScanObjectNN, and ModelNet40. Experimental results show that the guidance scale of 3.0 gives the best performance. When the guidance scale is set to 0.3, the generated point cloud is expected to have minimal reliance on the input text, leading to weaker alignment between modalities. Conversely, at a guidance scale of 30, the point cloud strongly reflects the input text; however, since all samples are generated using the same text, this likely results in a loss of diversity. This trade-off between alignment and diversity suggests that the highest accuracy is achieved with the standard guidance scale of 3.0.

Table 4. Ablation studies of the balance of real and synthetic data. We denote the replacement ratio of ShapeNet with synthetic data as Point-E (PE) / ShapeNet (SN)-%.

PE/SN-%	O-LVIS Top1	S-Object Top1	M40 Top1
0	9.01	50.3	63.7
25	9.47	47.4	67.7
50	8.93	43.8	67.6
100	0.3	14.0	9.4

5.3. Comparison with Conventional Methods

In Table 3, we compare the zero-shot 3D classification results of MixCon3D. One model is trained with both ShapeNet and the synthetic dataset generated by TeGA, and the other is solely trained with ShapeNet. We generate a synthetic dataset with the same number of samples as ShapeNet. As a result, our proposed method effectively doubles the amount of training data compared to the original ShapeNet. Table 3 shows that our approach outperforms the baseline MixCon3D, which was trained solely on the original ShapeNet, across all benchmark datasets and evaluation metrics. Specifically, our method demonstrates substantial improvements, achieving gains of 3.0% on Objaverse-LVIS, 4.4% on ScanObjectNN, and 8.8% on ModelNet40 in zero-shot performance. These results clearly demonstrate that TeGA is an effective method for dataset expansion in zero-shot 3D classification, even when using synthetic 3D data. Furthermore, the ability to scale the dataset while maintaining or improving model performance opens up new possibilities for addressing data scarcity in 3D vision tasks.

5.4. Ablation Studies

This section analyzes the factors that contribute to performance improvements in dataset expansion using TeGA. Specifically, we investigate (i) the mixing ratio of synthetic 3D data and (ii) the scalability of synthetic 3D data.

Mixing ratio of synthetic 3D data (see Table 4 and Figure 5). In this experiment, we investigate whether synthetic data can serve as a substitute for real data. While keeping the total number of samples the same as the original ShapeNet, we replace a portion of the real data with synthetic data generated by TeGA. Table 4 presents the Top1 performance in zero-shot 3D classification with varying proportions of synthetic 3D data. Surprisingly, it shows that the best performance is achieved when 25% of the orig-

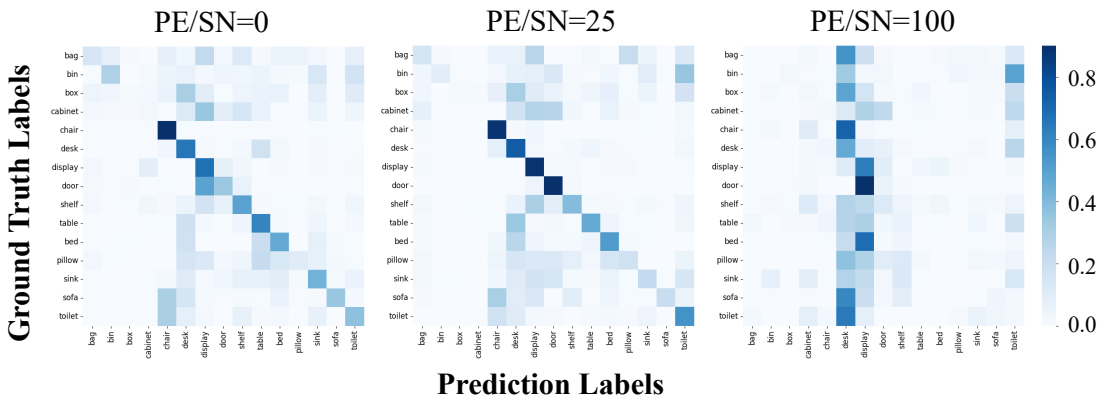

Figure 5. Confusion matrics when varying PE/SP. When the proportion of real data decreases, the model's predictions increasingly skew towards desk and display, leading to an overall deterioration in prediction accuracy.

Table 5. Ablation studies of varying scale of synthetic data.

Scaling	O-LVIS Top1	S-Object Top1	M40 Top1
×0.1	11.1	48.2	71.0
×1	11.2	47.1	73.4
×2	12.1	45.1	73.8

inal ShapeNet is replaced with synthetic 3D data, with a performance improvement over using ShapeNet alone. However, as the proportion of synthetic 3D data increases beyond 25%, performance gradually deteriorates; when only synthetic data is used, training fails and classification performance significantly drops. This decline is likely due to the learning strategy of MixCon3D, which extracts knowledge from CLIP models, making it difficult for the model to learn from the synthetic 3D data generated by Point-E due to domain gaps.

We also analyze why the model fails to learn when trained solely on synthetic data. In this experiment, we construct a mixture matrix at PE / SN = {0, 25, 100}, where real 3D data is progressively replaced by synthetic 3D data, to examine how the model misclassifies. As shown in Figure 4, for PE / SN = {0, 25, 100}, the categories predicted by the model exhibit similar trends. However, when PE/SN = 100, i.e., when the model is trained on synthetic 3D data, the predicted labels are biased towards desk or display.

Scalability of synthetic 3D data. In this experiment, we examine how scaling the amount of synthetic data added to ShapeNet influences the performance of MixCon3D. Specifically, we assess performance changes when Point-E generated data is scaled to 0.1x, 1x and 2x the original ShapeNet data. The scaling is based on ShapeNet's default dataset size of 53,470. Table 5 presents the Top1 performance with varying the amount of generated data. It shows that scaling plays a critical role in MixCon3D's training and that synthetic data effectively contributes to this scaling. According to these results, accuracy improves with the addition of scaled synthetic data for Objaverse-LVIS and ModelNet40. In contrast, the accuracy decreases with scaling for ScanObjectNN. This decline is likely due to ScanObjectNN's scores being more sensitive to noise.

6. Conclusion

To fix the problem of 3D data scarcity in zero-shot 3D classification, we propose TeGA (Text-guided Geometric Augmentation), a language-image-3D dataset expansion method with high-quality synthetic 3D data. TeGA plays a significant role in the expansion of existing datasets. To further enhance the quality of 3D data, we apply consistency filtering to remove unaligned data that could negatively impact training. In practice, by combining the generated synthetic data with existing 3D datasets, we demonstrated improved performance in zero-shot 3D classification. Based on these results, we believe text-to-3D models have the potential to alleviate challenges associated with 3D data collection and, when leveraged for data expansion, can support the development of generalizable vision models for 3D vision.

Limitations. In this paper, we adopted Point-E as a text-to-3D model. However, Point-E is a generative model trained on data that is not open to the public. Since it is difficult to generate out-of-distribution data not included in the training dataset, our framework heavily relies on the generative model. Therefore, we plan to focus on improving the text-to-3D model in the future. Additionally, when incorporating synthetic 3D data generated by Point-E into the training dataset, there is concern that societal biases or other unintended elements may be inadvertently introduced. It is recommended to take these risks into account and incorporate the findings from this study into the model improvement process. Furthermore, we look forward to the development of clean Text-to-3D models, like Stable Diffusion, which are built on publicly available datasets.

References

[1] Moshe Bar. Visual objects in context. *Nature Reviews Neuroscience*, 5(8):617–629, 2004. 3

[2] Fausto Bernardini, Joshua Mittleman, Holly Rushmeier, Claudio Silva, and Gabriel Taubin. The ball-pivoting algorithm for surface reconstruction. *IEEE Transactions on Visualization and Computer Graphics*, 5(4):349–359, 1999. 5

[3] Angel X Chang, Thomas Funkhouser, Leonidas Guibas, Pat Hanrahan, Qixing Huang, Zimo Li, Silvio Savarese, Manolis Savva, Shuran Song, Hao Su, et al. Shapenet: An information-rich 3d model repository. *arXiv preprint arXiv:1512.03012*, 2015. 6

[4] Rui Chen, Yongwei Chen, Ningxin Jiao, and Kui Jia. Fantasia3d: Disentangling geometry and appearance for high-quality text-to-3d content creation. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 22246–22256, 2023. 2, 3

[5] Mehdi Cherti, Romain Beaumont, Ross Wightman, Mitchell Wortsman, Gabriel Ilharco, Cade Gordon, Christoph Schuhmann, Ludwig Schmidt, and Jenia Jitsev. Reproducible scaling laws for contrastive language-image learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2818–2829, 2023. 3, 6, 11

[6] Jasmine Collins, Shubham Goel, Kenan Deng, Achleshwar Luthra, Leon Xu, Erhan Gundogdu, Xi Zhang, Tomas F Yago Vicente, Thomas Dideriksen, Himanshu Arora, et al. Abo: Dataset and benchmarks for real-world 3d object understanding. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 21126–21136, 2022. 6

[7] Matt Deitke, Dustin Schwenk, Jordi Salvador, Luca Weihs, Oscar Michel, Eli VanderBilt, Ludwig Schmidt, Kiana Ehsani, Aniruddha Kembhavi, and Ali Farhadi. Objaverse: A universe of annotated 3d objects. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition(CVPR)*, pages 13142–13153, 2023. 2, 6

[8] Matt Deitke, Ruoshi Liu, Matthew Wallingford, Huong Ngo, Oscar Michel, Aditya Kusupati, Alan Fan, Christian Laforte, Vikram Voleti, Samir Yitzhak Gadre, et al. Objaverse-xl: A universe of 10m+ 3d objects. *Advances in Neural Information Processing Systems*, 36, 2024. 2

[9] Huan Fu, Rongfei Jia, Lin Gao, Mingming Gong, Binqiang Zhao, Steve Maybank, and Dacheng Tao. 3d-future: 3d furniture shape with texture. *International Journal of Computer Vision*, 129:3313–3337, 2021. 6

[10] Yipeng Gao, Zeyu Wang, Wei-Shi Zheng, Cihang Xie, and Yuyin Zhou. Sculpting holistic 3d representation in contrastive language-image-3d pre-training. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition(CVPR)*, pages 22998–23008, 2024. 2, 3

[11] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial networks. *Communications of the ACM*, 63(11):139–144, 2020. 3

[12] Yuze He, Yushi Bai, Matthieu Lin, Wang Zhao, Yubin Hu, Jenny Sheng, Ran Yi, Juanzi Li, and Yong-Jin Liu. T3 bench: Benchmarking current progress in text-to-3d generation. *arXiv preprint arXiv:2310.02977*, 2023. 5, 6, 11

[13] Deepti Hegde, Jeya Maria Jose Valanarasu, and Vishal Patel. Clip goes 3d: Leveraging prompt tuning for language grounded 3d recognition. In *Proceedings of the IEEE/CVF International Conference on Computer Vision(CVPR)*, pages 2028–2038, 2023. 2, 3

[14] Tianyu Huang, Bowen Dong, Yunhan Yang, Xiaoshui Huang, Rynson WH Lau, Wanli Ouyang, and Wangmeng Zuo. Clip2point: Transfer clip to point cloud classification with image-depth pre-training. In *Proceedings of the IEEE/CVF International Conference on Computer Vision(CVPR)*, pages 22157–22167, 2023. 2, 3

[15] Phillip Isola. When faking your data actually helps – learning vision from GANs, NeRFs, and noise, 2022. BMVC Keynote talk. 2, 3

[16] Heewoo Jun and Alex Nichol. Shap-e: Generating conditional 3d implicit functions. *arXiv preprint arXiv:2305.02463*, 2023. 2

[17] Roozbeh Kiani, Hossein Esteky, Koorosh Mirpour, and Keiji Tanaka. Object category structure in response patterns of neuronal population in monkey inferior temporal cortex. *Journal of neurophysiology*, 97(6):4296–4309, 2007. 3

[18] Diederik P Kingma. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013. 3

[19] Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. In *International conference on machine learning*, pages 19730–19742. PMLR, 2023. 5

[20] Chen-Hsuan Lin, Jun Gao, Luming Tang, Towaki Takikawa, Xiaohui Zeng, Xun Huang, Karsten Kreis, Sanja Fidler, Ming-Yu Liu, and Tsung-Yi Lin. Magic3d: High-resolution text-to-3d content creation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 300–309, 2023. 2, 3

[21] Minghua Liu, Ruoxi Shi, Kaiming Kuang, Yinhao Zhu, Xuanlin Li, Shizhong Han, Hong Cai, Fatih Porikli, and Hao Su. Openshape: Scaling up 3d shape representation towards open-world understanding. *Advances in Neural Information Processing Systems*, 36, 2024. 2, 3, 6

[22] Ruoshi Liu, Rundi Wu, Basile Van Hoorick, Pavel Tokmakov, Sergey Zakharov, and Carl Vondrick. Zero-1-to-3: Zero-shot one image to 3d object. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 9298–9309, 2023. 2

[23] I Loshchilov. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017. 6

[24] Ilya Loshchilov and Frank Hutter. Sgdr: Stochastic gradient descent with warm restarts. *arXiv preprint arXiv:1608.03983*, 2016. 6

[25] Alex Nichol, Prafulla Dhariwal, Aditya Ramesh, Pranav Shyam, Pamela Mishkin, Bob McGrew, Ilya Sutskever, and Mark Chen. Glide: Towards photorealistic image generation and editing with text-guided diffusion models. *arXiv preprint arXiv:2112.10741*, 2021. 2

[26] Alex Nichol, Heewoo Jun, Prafulla Dhariwal, Pamela Mishkin, and Mark Chen. Point-e: A system for generating 3d point clouds from complex prompts. *arXiv preprint arXiv:2212.08751*, 2022. 2, 3, 4

[27] Maxime Oquab, Timothee Darcet, Theo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, et al. Dinov2: Learning robust visual features without supervision. *arXiv preprint arXiv:2304.07193*, 2023. 3

[28] Ben Poole, Ajay Jain, Jonathan T Barron, and Ben Mildenhall. Dreamfusion: Text-to-3d using 2d diffusion. *arXiv preprint arXiv:2209.14988*, 2022. 2, 3

[29] Zekun Qi, Runpei Dong, Guofan Fan, Zheng Ge, Xiangyu Zhang, Kaisheng Ma, and Li Yi. Contrast with reconstruct: Contrastive 3d representation learning guided by generative pretraining. In *International Conference on Machine Learning(ICML)*, pages 28223–28243. PMLR, 2023. 2, 3

[30] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning(ICML)*, pages 8748–8763. PMLR, 2021. 1, 3

[31] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Bjorn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition(CVPR)*, pages 10684–10695, 2022. 2, 3

[32] Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily L Denton, Kamyar Ghasemipour, Raphael Gontijo Lopes, Burcu Karagol Ayan, Tim Salimans, et al. Photorealistic text-to-image diffusion models with deep language understanding. *Advances in neural information processing systems*, 35:36479–36494, 2022. 2

[33] Mert Bulent Sariyildiz, Karteek Alahari, Diane Larlus, and Yannis Kalantidis. Fake it till you make it: Learning transferable representations from synthetic imagenet clones. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8011–8021, 2023. 2, 3

[34] Christoph Schuhmann, Romain Beaumont, Richard Vencu, Cade Gordon, Ross Wightman, Mehdi Cherti, Theo Coombes, Aarush Katta, Clayton Mullis, Mitchell Wortsman, et al. Laion-5b: An open large-scale dataset for training next generation image-text models. *Advances in Neural Information Processing Systems*, 35:25278–25294, 2022. 1

[35] Yonglong Tian, Lijie Fan, Kaifeng Chen, Dina Katabi, Dilip Krishnan, and Phillip Isola. Learning vision from models rivals learning vision from data. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition(CVPR)*, pages 15887–15898, 2024. 3

[36] Yonglong Tian, Lijie Fan, Phillip Isola, Huiwen Chang, and Dilip Krishnan. Stablerep: Synthetic images from text-to-image models make strong visual representation learners. *Advances in Neural Information Processing Systems*, 36, 2024. 3

[37] Christina Tsalicoglou, Fabian Manhardt, Alessio Tonioni, Michael Niemeyer, and Federico Tombari. Textmesh: Generation of realistic 3d meshes from text prompts. In *2024 International Conference on 3D Vision (3DV)*, pages 1554–1563. IEEE, 2024. 2

[38] Mikaela Angelina Uy, Quang-Hieu Pham, Binh-Son Hua, Thanh Nguyen, and Sai-Kit Yeung. Revisiting point cloud classification: A new benchmark dataset and classification model on real-world data. In *Proceedings of the IEEE/CVF International Conference on computer vision(CVPR)*, pages 1588–1597, 2019. 2, 6

[39] Zhengyi Wang, Cheng Lu, Yikai Wang, Fan Bao, Chongxuan Li, Hang Su, and Jun Zhu. Prolificdreamer: High-fidelity and diverse text-to-3d generation with variational score distillation. *Advances in Neural Information Processing Systems*, 36, 2024. 2

[40] Zhirong Wu, Shuran Song, Aditya Khosla, Fisher Yu, Linguang Zhang, Xiaoou Tang, and Jianxiong Xiao. 3d shapenets: A deep representation for volumetric shapes. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition(CVPR)*, pages 1912–1920, 2015. 2, 6

[41] Le Xue, Ning Yu, Shu Zhang, Artemis Panagopoulou, Junnan Li, Roberto Martin-Martjn, Jiajun Wu, Caiming Xiong, Ran Xu, Juan Carlos Niebles, et al. Ulip-2: Towards scalable multimodal pre-training for 3d understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition(CVPR)*, pages 27091–27101, 2024. 2, 3

[42] Junsheng Zhou, Jinsheng Wang, Baorui Ma, Yu-Shen Liu, Tiejun Huang, and Xinlong Wang. Uni3d: Exploring unified 3d representation at scale. *arXiv preprint arXiv:2310.06773*, 2023. 3

A. Details of TeGA

A.1. Details of the generated data by TeGA

We provide samples of data generated by TeGA (Fig. A and Fig. B). The generated data which pass consistency filtering are shown in Fig. A and the generated data which are filtered out are shown in Fig. B. Tab. A compares some of the prompts used in the generation with the percentage that passed through the filter.

Classes that pass consistency filtering in large numbers tend to have clear class names. On the other hand, those that frequently fail consistency filtering often have ambiguous class names or simple shapes that make correspondence difficult to capture. For example, the 'birdhouse' has no fixed shape and could be a mesh or a wooden box. In such cases, it is difficult to generate and evaluate. Conversely, the 'bicycle' has fixed the shape which is a vehicle with two wheels, a frame, handlebars for steering, and pedals for propulsion. Some classes, despite having clear class names and not being simple in shape, are still prone to failing consistency filtering.

Class Name	Count (Filtered/Generated)	Percentage (%)
airplane	3777/4045	93.38%
bicycle	449/831	54.03%
birdhouse	0/73	0.00%
bookshelf	59/452	13.05%
bottle	433/498	86.95%
chair	6473/6778	95.05%
clock	44/651	6.76%
display	8/1093	7.32%
lamp	1052/2318	45.38%
telephone	42/1089	3.86%

Table A. ShapeNet class name and the percentage that passed the filtering of the generation. A high percentage of the class names that are ambiguous are caught in the filtering process.

A.2. Details of the consistency filtering by TeGA

In Tab. B and Tab. C, Tab. D, we show an example of a part of TeGA using GPT-4. This evaluation is instructed by pre-defined prompt. The prompt is based on T3Bench [12]. In T3Bench, the prompt for generation assumes a concrete long sentence so there needs only to perform sense-level comparisons between caption and prompt. However, in our evaluation, since the prompt for generation is the class name of ShapeNet, which is a word, T3Bench always returns low score. For these reasons, it is necessary to account for conceptual similarities and we redesigned the prompt for evaluation.

In the generated captions, key information is sometimes placed at the end of the sentence, and low scores are output by GPT-4 in such cases. For example, in the case of Tab. C, despite the presence of the same classmate word 'sofa', a low score is output. In such cases, performing word-level matching outputs a high score and is able to complement GPT-4's shortcomings.

Text-PC	Image-PC	Text-3D	ModelNet-40
			Top1
✓	✗	✗	56.3
✗	✓	✗	35.9
✗	✗	✓	0.4
✗	✓	✓	60.3
✓	✓	✗	42.2
✓	✓	✓	**73.3**

Table E. Comparison of the final accuracy achieved when MixCon3D's contrastive learning is limited. A check mark indicates an active contrastive learning setting, while a cross mark denotes an inactive setting.

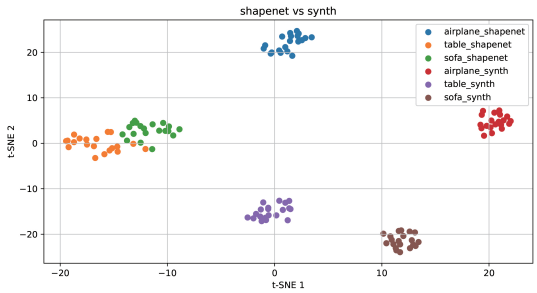

Figure C. Visualization of features. Features with the same class name are clustered close to each other. However, real data and synthetic data are not clustered close to each other.

B. Additional Experimental Results

Multi-modal Pretraining. We investigate which modality interactions contribute to training for language-image-3D tasks using synthetic data. Specifically, it compares the final accuracy achieved when MixCon3D's contrastive learning is applied to only a subset of the modalities. Tab. E presents the results. A check mark indicates an active contrastive learning setting, while a cross mark denotes an inactive setting. Note that the text-image contrastive learning is always active due to knowledge distillation from OpenCLIP [5].

Surprisingly, high performance is achieved even without utilizing text-point cloud contrastive learning. Additionally, although training succeeds without text-3D contrastive learning, the accuracy is higher when it is included, indicating that text-3D contrastive learning definitely contributes to improving performance.

Visualize Feature. Fig. C shows the visualization of the predicted feature by MixCon3D. The features of the three

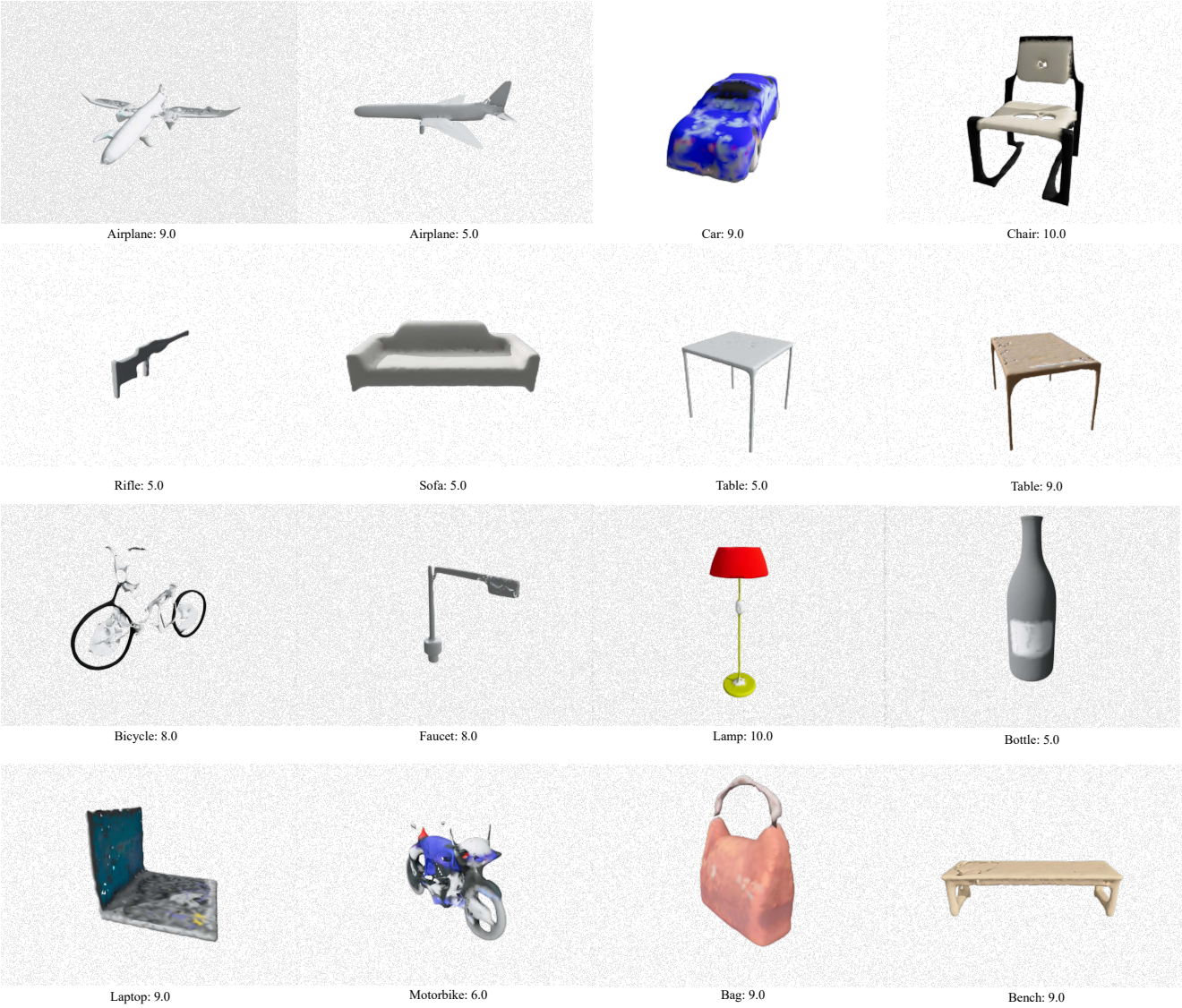

Figure A. Samples of data generated by TeGA.

classes—airplane, table, and sofa—are extracted from both ShapeNet data and synthetic data, and dimensionality reduction is performed using t-SNE. Surprisingly, not only are different classes separated, but even the ShapeNet data and synthetic data of the same class are distinctly separated.

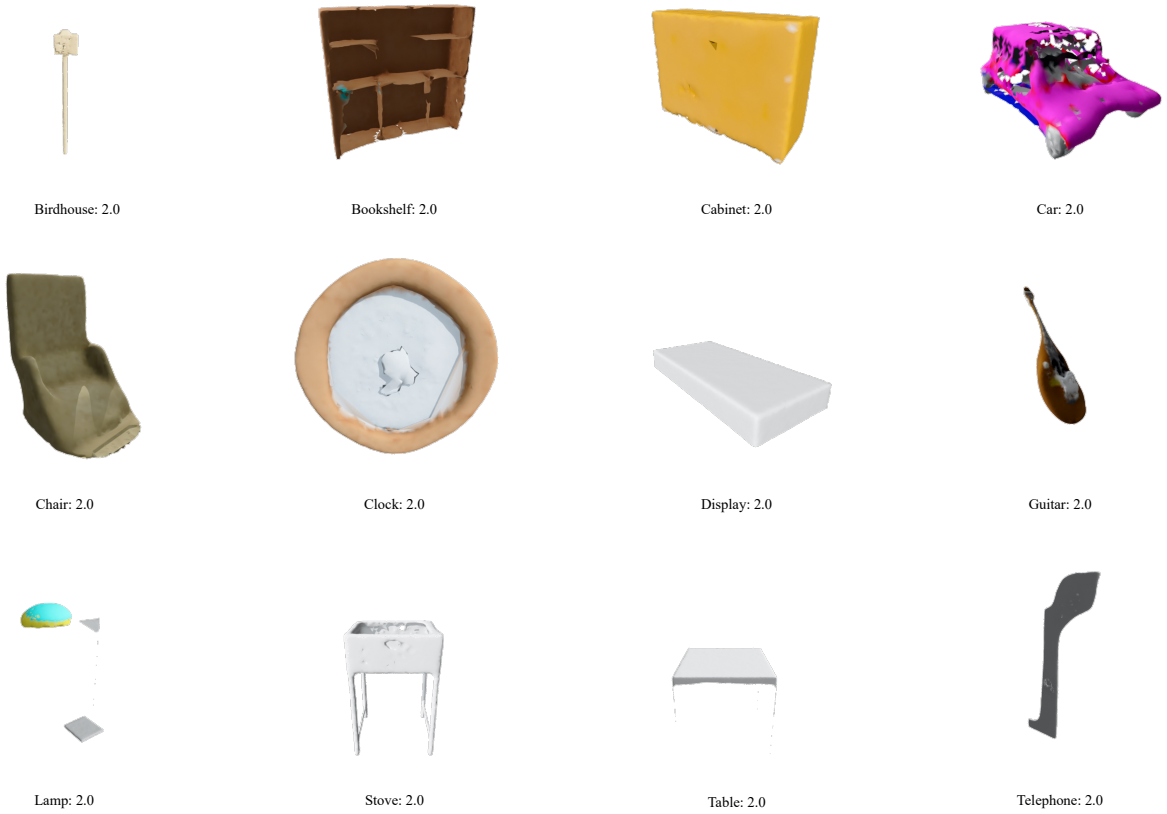

Figure B. Samples of data generated but filtered out in TeGA.

Prompt	You are an assessment expert responsible for prompt-prediction pairs. Your task is to score the prediction according to the following requirements: 1. Evaluate the recall, or how well the prediction covers the information in the prompt. If the prediction contains information that does not appear in the prompt, it should not be considered as bad. 2. Assign a score between 1 and 5, with 5 being the highest. Do not provide a complete answer; give the score in the format: 3 3. add points if the prediction and prompt are conceptually close (e.g. similar in appearance). (e.g., bike and bycicle and table and chair are close) 4. since the prompt is at the word level, it is inevitable that some detailed information is missing, so exclude it from the point deduction. prompt: car prediction: A 3D rendering of a car with a pink and white exterior and a pink interior with red streaks
Answer:	5
Word matching	5
Total	10 (safe)

Table B. The process of consistency filtering in TeGA. Captions and prompts are evaluated by GPT-4 and by word-matching. Both of evaluation are judged to be high.

Prompt	You are an assessment expert responsible for prompt-prediction pairs. Your task is to score the prediction according to the following requirements: 1. Evaluate the recall, or how well the prediction covers the information in the prompt. If the prediction contains information that does not appear in the prompt, it should not be considered as bad. 2. Assign a score between 1 and 5, with 5 being the highest. Do not provide a complete answer; give the score in the format: 3 3. add points if the prediction and prompt are conceptually close (e.g. similar in appearance). (e.g., bike and bycicle and table and chair are close) 4. since the prompt is at the word level, it is inevitable that some detailed information is missing, so exclude it from the point deduction. prompt: sofa prediction: A modern, cream-colored sofa
GPT-4 Answer	1
Word matching	5
Total	6 (safe)

Table C. The process of consistency filtering in TeGA. Captions and prompts are evaluated by GPT-4 and by word-matching. The prompts appear in the caption, but are judged low by GPT-4, while word-matching is judged high. On the other hand, the word-matching score is high.

Prompt	You are an assessment expert responsible for prompt-prediction pairs. Your task is to score the prediction according to the following requirements: 1. Evaluate the recall, or how well the prediction covers the information in the prompt. If the prediction contains information that does not appear in the prompt, it should not be considered as bad. 2. Assign a score between 1 and 5, with 5 being the highest. Do not provide a complete answer; give the score in the format: 3 3. add points if the prediction and prompt are conceptually close (e.g. similar in appearance). (e.g., bike and bycicle and table and chair are close) 4. since the prompt is at the word level, it is inevitable that some detailed information is missing, so exclude it from the point deduction. prompt: birdhouse prediction: A black and white artistic object
GPT-4 Answer	2
Word matching	1
Total	3 (ng)

Table D. The process of consistency filtering in TeGA. Captions and prompts are evaluated by GPT-4 and by word-matching. Both of evaluation are judged to be low.