MV-CLIP: Multi-View CLIP for Zero-shot 3D Shape Recognition

Dan Song[1], Xinwei Fu[1], Weizhi Nie[1], Wenhui Li[1], Lanjun Wang[1], You Yang[2], Anan Liu[1] *
[1]Tianjin University
[2]Huazhong University of Science and Technology

ABSTRACT

Large-scale pre-trained models have demonstrated impressive performance in vision and language tasks within open-world scenarios. Due to the lack of comparable pre-trained models for 3D shapes, recent methods utilize language-image pre-training to realize zero-shot 3D shape recognition. However, due to the modality gap, pretrained language-image models are not confident enough in the generalization to 3D shape recognition. Consequently, this paper aims to improve the confidence with view selection and hierarchical prompts. Leveraging the CLIP model as an example, we employ view selection on the vision side by identifying views with high prediction confidence from multiple rendered views of a 3D shape. On the textual side, the strategy of hierarchical prompts is proposed for the first time. The first layer prompts several classification candidates with traditional class-level descriptions, while the second layer refines the prediction based on function-level descriptions or further distinctions between the candidates. Remarkably, without the need for additional training, our proposed method achieves impressive zero-shot 3D classification accuracies of 84.44%, 91.51%, and 66.17% on ModelNet40, ModelNet10, and ShapeNet Core55, respectively. Furthermore, we will make the code publicly available to facilitate reproducibility and further research in this area.

CCS CONCEPTS

• **Computing methodologies** → **Artificial intelligence**; **Computer vision**; **Computer vision tasks**; **Visual content-based indexing and retrieval**.

KEYWORDS

3D shape recognition, Zero-shot recognition, Multi-view representation, Multi-modal pretrained models

1 INTRODUCTION

With the extensive applications of 3D models in computer-aided design (CAD), autonomous driving and virtual reality/augmented reality (VR/AR), together with the rapid advancements in 3D scanning and reconstruction technologies, the number of 3D shapes has experienced an explosive increase. How to effectively identify and manage these unlabeled 3D data has become a challenging problem. Zero-shot 3D shape recognition aims to classify unseen 3D shapes without explicit training, which has become a hot topic in computer vision with significant benefits such as identifying novel objects and alleviating labor-intensive annotations.

Traditional zero-shot methods [5, 6, 30] rely on a limited distribution of "seen" 3D shape data, resulting in insufficient generalization to new "unseen" categories. Additionally, the hand-crafted semantic attributes designed for "seen" data cannot cover the characteristics of "unseen" data and mapping high-level shape features to these attributes is difficult. Pre-trained large-scale models have demonstrated strong generalization capabilities, making them highly favored for zero-shot tasks. Due to the absence of comparable pre-trained models specifically designed for 3D shapes, recent methods utilize vision-language models (e.g., CLIP [26]) to realize zero-shot 3D shape recognition, which can be classified into training-based and non-training-based methods. Training-based methods [13, 35] create multi-modal datasets for 3D shapes and perform multi-modal contrasts. Integrating 3D data into the pre-training stage greatly enhances the capabilities of zero-shot 3D recognition. However, such approach comes with challenges such as the need for large-scale 3D data, extensive pre-processing requirements, and high computational training costs. Non-training-based methods [28, 41, 47] render 3D shapes into images which are later encoded by the visual encoder of CLIP and compared with the textual encoding of category labels. Our approach falls into the non-training paradigm and raises concerns about the reliability of CLIP when applied to 3D shape recognition, considering both the visual and textual prompt aspects.

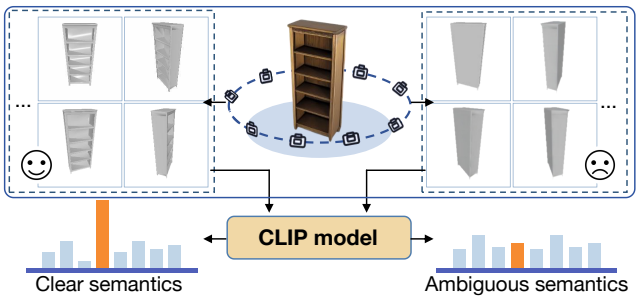

Figure 1: Improve CLIP's confidence at zero-shot 3D shape recognition in the visual aspect: select images with clear semantics.

Visual aspect. PointCLIP [41] projected 3D point clouds to sparsely distributed points in a depth map, which for the first time leverage CLIP for zero-shot 3D shape recognition. To improve the image quality according to CLIP's preference, PointCLIP V2 [47] transformed point clouds into voxels to generate much smoother projection values and DiffCLIP [28] enhanced the style of depth map closely to natural photos with diffusion model. CLIP2Point [14] enhanced the projection between points and pixels and refined the issue of excessive blank area in rendered depth images. However, as shown in Figure 1, the view images with ambiguous semantics confuse CLIP and will hinder the performance of shape recognition.

Prompt aspect. PointCLIP [41] adopted a hand-crafted template "`point cloud depth map of a [class]`" as prompt. Similarly, DiffCLIP [28] used "`a 3D rendered image of a [class]`". To

*Corresponding Author

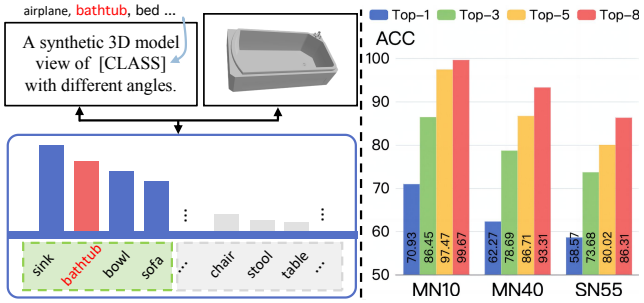

Figure 2: Improve CLIP's confidence at zero-shot 3D shape recognition in the textual prompt: refine the prediction with hierarchical prompts. Left: an example of bathtub that is mis-classified into sink. Right: Statistical zero-shot top-k accuracy on three popular datasets, which is obtained by the CLIP model under the settings of 12 pre-defined online-rendered views and hand-crafted prompts of first layer.

design a more detailed prompt in 3D perspective, PointCLIP V2 fed 3D command into GPT-3 like "`Give a caption of a table depth map`" and obtained 3D-specific prompt such as "`A height map of a table with a top and several legs`". By matching the visual feature and textual feature encoded by pre-trained model, the class that owns the highest similarity score becomes the predicted result. However, based on our observations as shown in Figure 2, sometimes the visual encoding does not perfectly match the ground-truth textual encoding, but usually ranks at the forefront. Directly adopting the top-1 result as prediction will limit further optimization.

To improve the confidence of pre-trained vision-language models towards the task of zero-shot 3D shape recognition, the proposed MV-CLIP (Multi-View CLIP) is equipped with view selection and hierarchical prompts. On the visual side, we select the view images with clear semantics based on the prediction entropy and fuse the classification results of the selected views as the final prediction. On the textual side, we propose a novel prompt mechanism named hierarchical prompts. For the first layer, we design a hand-crafted prompt as "`a synthetic 3D model view of [class] with different angles`". By matching the encoding of selected views and hand-crafted prompts via MV-CLIP, we acquire top-k results as candidates for further consideration. The second layer for refinement is designed based on current large language model (GPT-3.5 [3]). Specifically towards the candidate classes, we feed sentences like "`describe the visual characteristics and functional features of [candidate classes]`" or "`what is the difference between [candidate classes] in visual characteristics and functional features`". The concise answer generated by GPT is then used as the textual prompts for these candidates. View selection improves CLIP's confidence by filtering confusing views while hierarchical prompts give MV-CLIP a second chance with more specific descriptions. Consequently, without any training the proposed method achieves impressive zero-shot 3D classification accuracies of 84.44%, 91.51%, and 66.17% on ModelNet40, ModelNet10, and ShapeNet Core55, respectively. With the performance improvement of vision-language models, the proposed lightweight method can be further extended to the latest models to enhance zero-shot shape recognition performance.

In summary, the contributions are as follows:

- We propose MV-CLIP for directly extracting multi-view features. We utilize the pre-trained image encoder as the backbone of the multi-view network, achieving state-of-the-art performance in zero-shot 3D shape recognition.
- We introduce a view selection module to evaluate the quality of each view based on the principle of entropy minimization. This allows us to identify views that have a positive impact on MV-CLIP.
- We propose a novel hierarchical prompts strategy to improve the matching between MV-CLIP's view representations and textual prompts. With the candidates voted by the first-layer classification, LLMs-powered prompts towards candidates contribute more accurate second-layer matching.

2 RELATED WORK

2.1 Zero-shot Learning in 3D Shape Recognition

In recent years, Vision-Language Models (VLMs) such as CLIP [26], which employs large-scale image-text contrastive pre-training, have achieved remarkable success in the realm of 2D visual task. Many works [13, 14, 20, 27, 32, 35, 38, 39, 41] have explored to apply VLMs in 3D learning, thereby attaining the capability of zero-shot recognition.

Some methods directly apply VLMs to the 3D domain. PointCLIP [41] stands as the pioneering effort in applying VLMs to 3D recognition. It directly projects point clouds into multi-view depth maps and utilizes a frozen CLIP model for zero-shot classification. With this advancement, PointCLIP V2 [47] introduces more realistic shape projection and utilizes LLMs-assisted 3D prompts to effectively mitigate the 2D-3D domain gap. Meanwhile, CLIP2Point [14] fixes CLIP and additionally trains a depth encoder using 3D dataset with initialization of CLIP's visual encoder.

In addition, other methods explore extending the multi-modal learning between images and language to 3D modalities [13, 20, 35, 36, 38], achieving impressive zero-shot 3D model recognition performance. ULIP [35] learns a unified representation between language, images, and point clouds, and it significantly enhances the recognition capability of 3D backbone. Furthermore, ULIP-2 [36] specifically focuses on the scalability and comprehensiveness of the language modality. The more recent work of Openshape [20] also adopts a multi-modality contrastive learning framework, which improves the ability of open-world 3D shape understanding by improving aspects such as data scalability, text quality, 3D backbone scaling, and data resampling.

In this paper, we focus on lightweight zero-shot 3D recognition without the necessity for 3D pre-training, resulting in outstanding zero-shot classification performance, comparable to the state-of-the-art results achieved by multi-modal contrast methods that need fine-tuning.

2.2 Multi-View in 3D Shape Recognition

For 3D shape recognition, multi-view representation stands as one of the most classical types, which employs 2D views from multiple perspectives to represent 3D shapes. The work proposed by

Bradski et al. [2] was the trailblazer in employing multiple views to depict 3D shapes. Subsequently, with the advancement of deep learning, MVCNN [29] uses 2D CNN to extract features from a set of predefined views and aggregates them into a descriptor that could effectively represent 3D shapes. View-GCN [33] utilizes dynamic graph convolutional networks for hierarchical learning, leveraging inter-view relationship information to aggregate multi-view features. MVTN [12] combines differentiable rendering techniques for predicting the optimal viewing angles in multi-view setups, which enhances the robustness and recognition performance of multi-view networks. Additionally, some recent studies [9, 40, 41, 43] process 3D point clouds in the form of multiple depth images, which further emphasizes the importance of multi-view in 3D shape recognition.

However, it is essential to ensure the effectiveness of each view in the multiple views for subsequent tasks. In this study, we exploit the extensive semantic insights provided by VLMs [26] to assess the efficacy of multiple views. We select a subset of views for each 3D shape, prioritizing those that offer semantic clarity and excluding those that introduce ambiguous information.

2.3 Prompt Learning in Vision

The concept of prompts is initially introduced in the field of Natural Language Processing (NLP) [17, 21, 31], and has found increasingly flexible and widespread application in pre-trained language models such as BERT [8] and the GPT [3] series. Inspired by the success of prompts in NLP, the practice of prompt engineering has also been adopted in 2D vision [1, 10, 16, 19, 26, 45, 46]. Some of them employ a few learnable prompts either in the encoder input [45, 46] or within the transformer layers [19] to adapt the model for enhanced alignment between text and images. Others introduce visual prompting to apply in the pixel space [1, 16] or embeddings of input images [10] without extensive retraining or fine-tuning.

Additionally, in vision approaches related to CLIP [26], notable advancements have been achieved by integrating LLMs to refine the prompts. CuPL [24] and CaFo [42] utilize GPT-3 [3] to improve the downstream capabilities of CLIP in handling a variety of 2D datasets. Meanwhile, CHiLS [23] employs GPT-3 [3] to generate subclass labels that form mutually mapping hierarchical label sets, which are utilized to produce the final prediction. PointCLIP V2 [47] prompts GPT-3 [3] to enhance performance in open-world 3D tasks using 3D-oriented commands. Concurrently, ULIP-2 [36] and OpenShape [20] leverage LLMs to improve the quality of 3D prompts, aiming to a more effective alignment across various modalities. Although LLMs facilitate the design of prompts, the single-step top-1 prediction limits further improvement.

In this paper, we propose a novel strategy of hierarchical prompts. At the first layer, candidate categories are voted via matching hand-crafted prompts. At the second layer, we further utilize GPT-3.5 [3] to generate 3D-specific prompts for these candidates in aspects of function and difference. The design of hierarchical prompts enables more accurate category prediction without the necessity for training.

3 METHODOLOGY

The overview of Multi-View CLIP (MV-CLIP) for zero-shot 3D shape recognition is illustrated in Figure 3. In Sec. 3.1, we give a brief introduction to multi-view rendering and visual feature extraction. Sec. 3.2 explains how semantic information indicated by CLIP [26] is utilized to filter ambiguous views. Furthermore, in Sec. 3.3, we design hierarchical prompts to firstly propose candidates and then refine the prediction.

3.1 View Rendering and Feature Extraction

In order to fully validate and implement the proposed view selection module, we employ a multi-view online renderer $\mathbf{R}$ [11]. In our approach, we initially render a total of M views $X = \{x_i\}_{i=1}^{M}$ for a 3D shape based on fixed view configurations. We adopt three different view configurations, among which the circular [29] aligns view-points on a circle around the object, the spherical [18, 33] aligns equally spaced view-points on a sphere surrounding the object, and the random selects randomly view-points around the object. In addition, we replace 2D CNNs in the traditional multi-view convolutional neural network [29] with pre-trained visual encoder, which is referred to as MV-CLIP for zero-shot 3D recognition network. The feature extraction for multiple views is formulated as:

$$\{f\}_{i=1}^{M} = \mathbf{E}_V\left(\mathbf{R}(\mathbf{S})\right) \tag{1}$$

where $\mathbf{E}_V$ denotes the visual encoder of MV-CLIP, $\mathbf{R}$ represents the online renderer, and $\mathbf{S}$ stands for an arbitrary 3D shape.

3.2 View Selection

The objective of this module is to select views that capture clear semantic features of the 3D shape from multiple pre-defined views. As entropy reflects the prediction uncertainty, we utilize it to evaluate the prediction confidence of pre-trained models. The views with higher prediction confidence are selected as representative views. In the following contents, we will first elaborate the prediction process via CLIP [26] and then compute entropy for view selection.

Specifically, we evaluate the semantic representation of each view by matching the straightforward and hand-crafted prompts. Based on Eq. 1, multiple views are passed through MV-CLIP to obtain corresponding visual features, denoted as $\{f\}_{i=1}^{M} \in \mathbb{R}^{M\times C}$. For textual prompts, we design a pre-defined template: "`a synthetic 3D model view of [class] with different angles .`" to generate hand-crafted prompts containing K categories and encode their textual features as $W_t \in \mathbb{R}^{K\times C}$. Subsequently, the prediction of each view is calculated separately,

$$logits_i = f_i W_t^T, for\ i = 1, \ldots, M \tag{2}$$

where each bit of $logits_i$ represents the similarity score between the i^{th} view and each category.

Then the entropy of $logits_i$ is computed as:

$$H(logits_i) = -\sum_{j=1}^{K} P(logits_{i,j}) \log_2 P(logits_{i,j}) \tag{3}$$

where $H(\cdot)$ denotes the entropy of $logits_i$, which is the information summation over all possible categories. Besides, $P(logits_{i,j})$ represents the probability of the occurrence of the j^{th} category, and the term $\log_2 P(logits_{i,j})$ is used to quantify the information content of the probability. Lower entropy signifies higher information quality compared with other views of the same 3D shape.

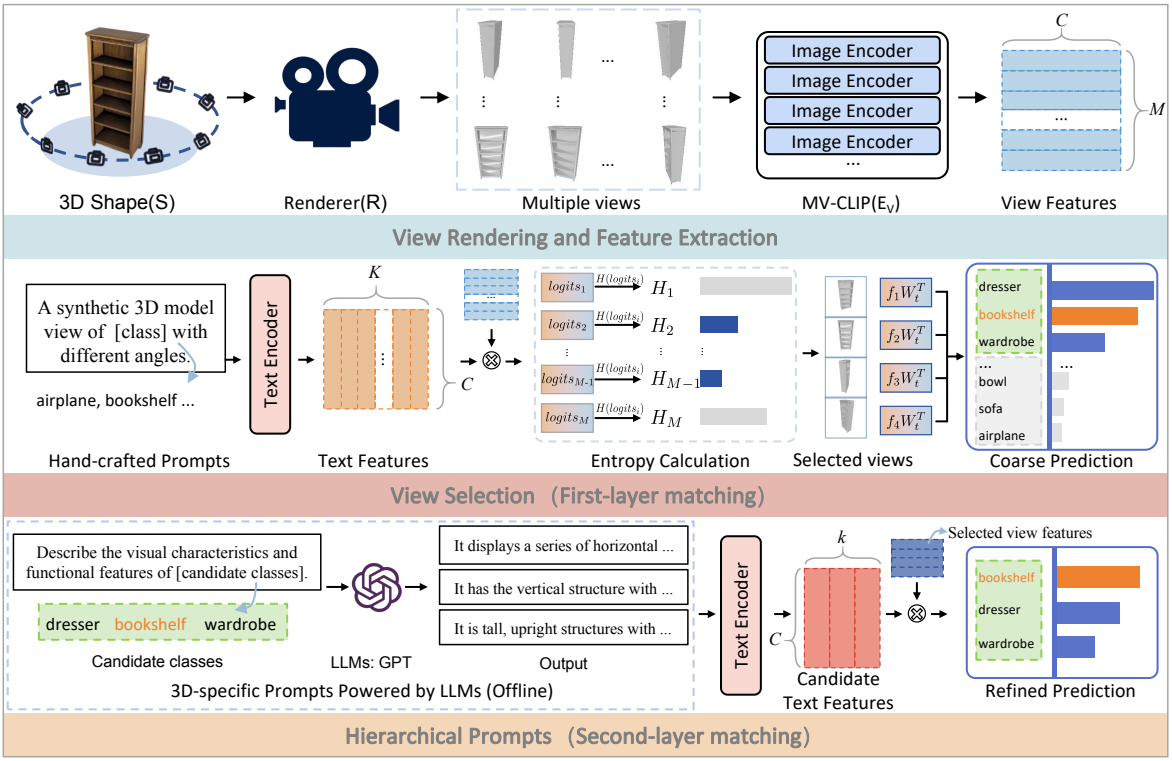

Figure 3: Overview of the proposed MV-CLIP for zero-shot 3D shape recognition. Firstly, multiple view images are obtained via a render R and the corresponding visual features are extracted via the visual encoder of CLIP [26]. Secondly, visual features are matched with textual features encoded by CLIP with hand-crafted prompts, and we select representative views according the prediction confidence. By aggregating the representative predictions, several candidates with the top classification probability are kept for the second matching. Finally, by matching the prompts powered by LLMs for these candidates, the prediction result is refined.

We rank all views by the entropy of prediction and select a subset of views with clearer semantics, i.e., smaller entropy, to represent the 3D shape. The selected views are denoted as:

$$X_{selec} = \{x_i | Rank(H(logits_i)) <= M_{selec}\} \tag{4}$$

where the $Rank(\cdot)$ is used to sort the information quality in ascending order, and M_{selec} represents the number of selected views.

3.3 Hierarchical Prompts

3.3.1 3D-Specific Prompts Powered by LLMs. As previously shown in Figure 2, due to modality gap, the prediction via CLIP is not confident enough. Therefore, we propose the hierarchical prompts where for the first layer hand-crafted prompts with clear category indication are used to vote several class candidates and for the second layer we utilize the prompts generated by powerful LLMs for these candidate classes. The prediction result gets refined by twice matching the view feature with hierarchical prompts.

With the consideration that the pre-trained models is trained using a collection of image-text pairs obtained from the Internet, besides focusing on the difference between candidates, we enhance the richness of prompts for augmenting the functional attributes. Specifically, we employ a pre-defined question template and utilize the GPT-3.5 [3] to generate 3D textual description: "`Describe the visual characteristics and functional features of [candidate classes]'s rendering view.`"

Take the candidate classes as [dresser, bookshelf, wardrobe] for example, GPT-3.5 produces:

- `It displays a series of horizontal shelves designed to hold books to provide storage for reading materials.`
- `It has the vertical structure with multiple drawers used for storing clothes or personal items.`
- `It is tall, upright structures with multiple compartments or shelves for storing clothes and accessories.`

Subsequently, the specific prompts are encoded by the pre-trained textual encoder $\mathbf{E}_T$ into text features, which are then utilized for second-layer classification. Formally, the textual features for candidate classes are represented as:

$$\{t\}_{i=1}^{k} = \mathbf{E}_T(\mathbf{LLM}(Q_i)) \tag{5}$$

where Q_i is the question template with the i^{th} candidate class and k represents the number of candidate classes.

3.3.2 Hierarchical Prompts Matching. The potential of the matching capability of 3D shape using hand-crafted prompts has been overlooked in zero-shot 3D learning with CLIP. Under the hand-crafted prompts, the first-layer classification score via MV-CLIP is computed by aggregating the scores of the selected views. Formally, denote the prediction of a 3D shape in the first layer as:

$$logits^I = \sum_{i=1}^{M_{selec}} logits_i \tag{6}$$

where $logits_i$ is computed as Eq. 2, and M_{selec} denotes the number of selected views.

If the prediction output by the first layer is not confident enough, e.g., the maximum probability within $logits^I$ is lower than a threshold δ, we choose the *top-k* labels as candidate classes for the second matching. Subsequently, at the second-layer, we employ the specific prompts generated by LLMs for re-matching to attain optimal classification outcomes.

Suppose the textual features in the second layer towards candidate classes are $W_{t_k} = \{t_1; t_2; t_3; \cdots; t_k\} \in \mathbb{R}^{k \times C}$ where t is encoded as Eq. 5, the final prediction is computed as:

$$logits^{II} = \sum_{i=1}^{M_{selec}} f_i W_{t_k}^T \tag{7}$$

$$y_{pre} = Argmax(logits^{II}) \tag{8}$$

Rather than directly classifying within the initial categories, we combine hand-crafted and LLMs generated prompts into a novel strategy of hierarchical prompts. For less certain samples, MV-CLIP will further focus on category details within a limited set of candidate labels, thereby achieving more accurate zero-shot recognition results.

4 EXPERIMENTS

4.1 Dataset

We evaluate the performance of zero-shot 3D shape classification on three popular datasets **ModelNet10**[34], **ModelNet40**[34] and **ShapeNet Core55**[4], using the complete test set without any pre-training on 3D training set.

ModelNet10 and ModelNet40 are two synthetic 3D model datasets commonly used for shape recognition and classification tasks. The former one selects 10 most common categories from ModelNet [34] and provides a simplified dataset. It contains approximately 4,899 3D models from 10 different categories, with 908 3D models used for testing. In contrast, the ModelNet40 includes a greater number of categories and exhibits a richer variety of 3D shapes. It consists of 40 categories from ModelNet [34] and comprises a total of 12,311 3D models, with 2,468 samples for testing. ShapeNet Core55 is a more diverse and challenging synthetic 3D dataset, widely used for 3D model analysis and understanding. It selects 55 commonly seen categories from ShapeNet [4], comprising approximately 51,300 shapes in total, with 10,265 shapes for testing.

4.2 Experimental Setting and Details

Our framework is built entirely on the PyTorch and experiments are executed on the NVIDIA RTX 3090 GPU. In terms of multi-view processing, we transform the MVCNN architecture by incorporating the publicly available pre-trained visual encoders, OpenCLIP[15] and CLIP[26], specifically utilizing the ViT/B-16 as the backbone network for extracting view features. Regarding the hierarchical prompts, we employ the existing large-scale language model GPT-3.5[3] to generate the 3D-specific prompts of candidate classes. To maintain the conciseness of the prompts, we enforce the prompt length constraint of 40.

In terms of experimental details, we designate the batch size as 4 and establish the confidence threshold parameter δ at 0.96. The rendering of multi-views is executed using the MV-pytorch [11] framework. Following the default settings of the online renderer, we uniformly set the rendering color to gray and the background color to white. The radial distance between the centroid of model and the camera is fixed at a value of 2, with an incorporation of random lighting conditions. The dimensions of the output views are set to 224×224 pixels. For the experimental results, unless explicitly stated otherwise, we concentrate on rendering views from a circular perspective. Except addressed particularly, we set the total number of views to 20 and the selected number of views to 4 for analysis.

4.3 Zero-shot 3D recognition

In Table 1, we evaluate the performance of zero-shot 3D shape recognition on ModelNet10 and ModelNet40 using the complete test set without any pre-training on 3D training set. Training-based methods utilize large-scale 3D datasets for multi-modal contrastive learning. CG3D [13], ULIP [35] and ULIP2 [36] employ an upgraded version of CLIP named SLIP [22], while OpenShape [20] uses the best model of OpenCLIP [15]. On the other hand, non-training-based methods like PointCLIP V2 [47] and DiffCLIP [14] use CLIP-ViT/B-16 [26]. For MV-CLIP, we use ViT/B-16 from both CLIP [26] and OpenCLIP [15] for our experiments. Besides, our approach achieves a zero-shot classification accuracy of 91.51% and 84.44% on the ModelNet10 and ModelNet40, respectively. The results are among the best in zero-shot 3D shape classification without any pre-training on 3D data, and they are comparable to the state-of-the-art results based on multi-modal contrast.

Although in the absence of 3D dataset training, our approach could still achieve competitive experimental results by fully leveraging the potential of the pre-trained model for 3D shape analysis. Our approach selects views with clear semantics from original multiple views, which significantly enhances the discriminative power of MV-CLIP and avoids the interference of semantically ambiguous views. Furthermore, under the first matching with hand-crafted prompts, we leverage LLMs-generated prompts that describe the characteristics within the scope of candidate labels for refined matching, effectively improving the zero-shot performance with hierarchical prompts.

4.4 Ablation Study

4.4.1 Effectiveness of the designed modules. As shown in Table 2, we conduct ablation study for individual modules and different versions of CLIP. We observe that with the addition of key components and the upgrade of CLIP versions, there is a consistent improvement in zero-shot 3D shape recognition performance.

We initially analyze the benefits brought by different moudles based on the baseline that uses OpenCLIP's ViT/B-16 as the backbone. By incorporating only the hierarchical prompts, the accuracy

Table 1: Zero-shot 3D shape classification performance. We compare the experimental results of existing zero-shot 3D learning methods using their best-performing settings on ModelNet10 and ModelNet40.

Method	CLIP version	Pre-training source	Zero-shot performance	
			ModelNet40	ModelNet10
CG3D[13]+PointTransformer[44]	SLIP[22]	ShapeNet[4]	50.6	-
ULIP[35]+PointBERT[37]		ShapeNet[4]	60.4	-
ULIP2[36]+Point-BERT[37]		ShapeNet[4]	66.4	-
ULIP2[36]+Point-BERT[37]		Objaverse[7]	74	-
OpenShape[20]+PointBERT[37]	OpenCLIP[15]	ShapeNet[4]	72.9	-
OpenShape[20]+PointBERT[37]		Ensembled(no LVIS)[20]	**85.3**	-
CLIP2Point[14]	CLIP[26]	ShapeNet[4]	49.38	66.63
Recon[25]			61.7	75.6
PointCLIP[41]	CLIP[26]	×	20.18	30.23
PointCLIP v2[47]		×	64.22	73.13
DiffCLIP[28]		×	49.7	80.6
Ours	CLIP[22]	×	65.92	77.53
	OpenCLIP[15]	×	84.44	**91.51**

Table 2: Performance comparison with different components. We conduct ablation study on ModelNet10, ModelNet40 and ShapeNet Core55 to explore the impact of individual designed modules on the experimental results, respectively.

CLIP (ViT\B-16)	OpenCLIP (ViT\B-16)	View selection	Hierarchical prompts	Zero-shot 3D classification		
				ModelNet40	ModelNet10	ShapeNet Core55
√	×	×	×	61.93	70.70	57.90
√	×	×	√	63.85 (↑ 1.92)	71.37 (↑ 0.67)	58.70 (↑ 0.8)
√	×	√	×	64.18 (↑ 2.25)	76.34 (↑ 5.64)	60.80 (↑ 2.9)
√	×	√	√	65.92 (↑ 3.99)	77.53 (↑ 6.83)	61.70 (↑ 3.8)
×	√	×	×	78.03	86.45	60.62
×	√	×	√	80.22 (↑ 2.19)	87.35 (↑ 0.9)	61.77 (↑ 1.15)
×	√	√	×	83.32 (↑ 5.29)	90.41 (↑ 3.96)	64.89 (↑ 4.27)
×	√	√	√	**84.44** (↑ 6.41)	**91.51** (↑ 5.06)	**66.17** (↑ 5.55)

improvements of 2.19%, 0.9%, and 1.15% are achieved on ModelNet40, ModelNet10 and ShapeNet Core55, respectively. It validates the effectiveness of being tolerate with several candidates and giving a further refined matching. By incorporating only the view selection, higher improvements are achieved, i.e., 5.29%, 3.96%, and 4.27% on three datasets. It shows that the selected views possess clearer semantic information and effectively mitigate the adverse impact of ambiguous views. Furthermore, it indicates that the first-layer of class-level hand-crafted prompts can effectively eliminate interfering categories, allowing the functional-level prompts of second-layer to have good candidate classes. When both modules are employed, we achieve the best results, with improvements of 6.41%, 5.06%, and 5.55% against baseline. The significant improvements demonstrates that the key modules could facilitate each other, both enhancing MV-CLIP's confidence towards 3D shape recognition.

In addition, we also conduct ablation experiments on different CLIP versions. By altering the backbone network in MV-CLIP from CLIP [26] to OpenCLIP [15], we observe the gains of 16.10%, 15.75%, and 2.72% on ModelNet40, ModelNet10 and ShapeNet Core55 respectively. With CLIP as backbone, the proposed modules bring the gains of 3.99%, 6.83%, and 3.8%. It implies that our method has potentials to adapt to various pre-trained vision-language models and will further lift the performance of zero-shot 3D shape recognition with rising advanced pre-trained models.

4.4.2 Discussions on the view selection module. Table 3, 4 and 5 show the classification results on ModelNet40 for different number of views, view rendering configurations and multi-view aggregation types, respectively. Additionally, we show the superiority of selected views with the average accuracy of individual view image and further discuss the variance of selected views, as shown in Table 6 and Figure 4.

(1) Different number of views. In Table 3, we find that in the case of multi-view setting without selection, once the number of views exceeds 12, the performance of 3D shape classification improves slowly (almost no gains). With the adoption of view selection, the zero-shot classification performance is steadily improved, but selecting too many views will bring redundant information and slightly hinder the performance.

(2) Different configurations of rendering. Table 4 shows not only the importance of view quality, but also the effectiveness of view

Table 3: View selection ablation with different number of views on ModelNet40.

	M	M_{selec}	First-layer	Second-layer
$w/$ Vs	20	12	82.62	83.43
		8	83.21	84.27
		4	**83.32**	**84.44**
w/o Vs		20	78.03	79.65
		12	78.02	80.22
		8	76.41	77.47
		4	64.38	61.91

Table 4: View selection ablation with different configurations of rendering on ModelNet40.

View configuration		First-layer	Second-layer
Random	w/o Vs	57.90	58.02
	$w/$ Vs	75.08 (↑ 17.18)	75.72 (↑ 17.7)
Spherical	w/o Vs	64.02	61.79
	$w/$ Vs	69.20 (↑ 5.18)	69.43 (↑ 7.64)
Circular	w/o Vs	78.03	80.22
	$w/$ Vs	**83.32** (↑ 5.29)	**84.44** (↑ 4.22)

Table 5: View selection ablation with different aggregation types on ModelNet40.

Aggregation type	Acc
$w/$ Mean pooling	66.89
$w/$ Max pooling	62.27
w/o Pooling	**84.44**

Table 6: Average accuracy for individual view on ModelNet40 (only first layer involved).

Dataset	All views (20)	Selected views (4)
MN40	57.54	79.97
MN10	66.37	86.71
SN55	51.23	64.07

Table 7: Hierarchical prompts ablation with different prompt settings on ModelNet40. All results are based on an accuracy of 83.32% at the first layer.

Prompts setting	Acc
Only visual characteristics prompts of candidate classes	83.72
Only functional features prompts of candidate classes	83.75
Fusion of visual characteristics and functional features prompts	84.31
The prompts of difference between candidate classes	84.42
The visual characteristics and functional features of candidate classes	**84.44**

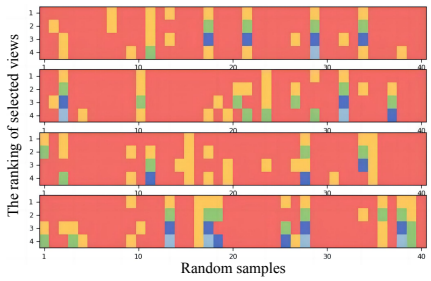

Figure 4: The decisions of the selected 4 views of 40×4 randomly chosen samples on ModelNet40.

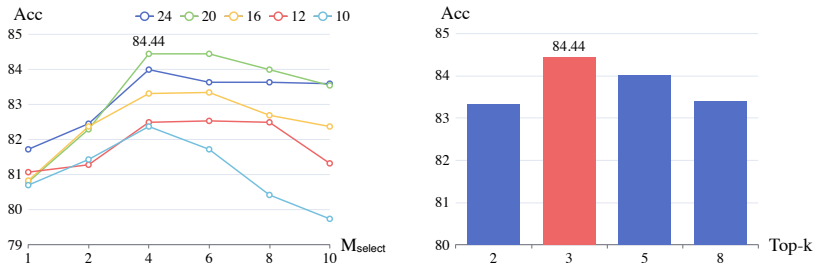

Figure 5: Sensitivity analysis of view selection and hierarchical prompts on ModelNet40.

selection to filter low-quality views. Furthermore, the gains caused by view selection will increase when the view sets contain poorer viewpoints. Please note that the camera positions of different configurations are shown in the supplement and Figure 7 gives an example of captured views under these configurations.

(3) Different multi-view aggregation types. In Table 5, given that the backbone network of MV-CLIP is a 2D pre-trained visual encoder, aggregating the features of multiple views negatively affects the alignment between views and texts. Therefore, unlike MVCNN, MV-CLIP fuses the prediction instead of any pooling of features.

(4) Average accuracy for individual view. Table 6 illustrates the average view prediction accuracy of the selected views and the total views on three datasets. We find that the views within the selected set have much higher accuracy than the rest, which shows that view selection choose views with high prediction confidence from multiple views, and filters the negative influence of views with ambiguous semantics on the final decision.

(5) Variance within the selected views. Figure 4 shows the variance within the selected view decisions, where red patch indicates the right decision and different colors in each column show the disagreement. We find that the views selected in most samples have similar decisions, and we regard the similar decision as a kind of weighting where the fused decision of selected views is impacted by majority voting. Furthermore, we tried another variant of view selection by keeping more diversity in decisions of selected views. Along the increase of entropy, we select 4 views with different decisions, but the accuracy only reaches 74.68% (84.44% for the proposed method). It indicates that incorporating views with contrary decisions may introduce ambiguity into the final decision-making process.

4.4.3 *Discussions on the hierarchical prompts module.*

(1) Different prompt settings. Several variants for prompting have been tried, with emphasis on only visual characteristics, only functional features, fusion of the both, difference between candidates and a combination of visual and functional features. Table 7 illustrates the results of different variants, which shows that both visual and functional characteristics are important and each aspect contributes comparatively. Putting emphasis on the difference between candidates is also an alternative way for the second matching.

(2) The correction capability of hierarchical prompts. As shown in Figure 6, we visualize some successful and failed cases caused by the second-layer matching based on hierarchical prompts. Sometimes the second matching changes the decision of first-layer matching. Statistically, the number of successful corrections by second-layer matching is approximately 2.3 times that of failure cases. Additionally, we find that in most of the successful cases, the unique function and shape characteristics compared to other categories are captured in the second match. For example, bathtub is used for bathing, dresser has multiple drawers for storing items, toilet has a water tank, and monitor has a rectangular screen. We also summarize the failed reasons as: (1) Visual similarity, e.g., vase and cup. (2) Limited rendering quality, e.g., views of glass_box do not display transparency which is an important attribute in the second layer prompt. (3) Prompt quality, e.g., desk has more explicit descriptions than table. (4) Co-existence of multiple objects, e.g., flower_pot and plant.

4.5 Sensitivity Analysis

We discuss the prediction accuracy affected by different views settings of view selection and different numbers of candidate classes, as shown in Figure 5. According to our observations, selecting too few views results in insufficient information, while selecting too many views reduces the benefits of view selection under a defined total number of views, which are not desirable for zero-shot 3D recognition. Furthermore, if the total number of views is excessively high, it can lead to high similarity among different views, resulting in the selected views being similar to each other and lack of diversity.

Additionally, we test on the classification results under the different numbers of candidate classes, as shown in Figure 5. We can observe that when the number of candidate labels equals to 3, it leads to the maximum gain in hierarchical prompts. It reflects that the LLMs powered prompts in the second layer, which describe fine-grained characteristics of categories, performs better within a limited number of classes.

4.6 Visualization

As shown in Figure 7, we visualize a subset of the selected views. It is observable that the chosen views typically encompass more comprehensive category features and have relatively clearer semantic information. In contrast, views with fewer features or even lacking semantic contents, like the bottom of a cup, the back of a piano or the side of a bookshelf, are not selected. Consequently, the view selection based on entropy minimization effectively reduces the redundancy present in the rendered multiple views of a 3D shape.

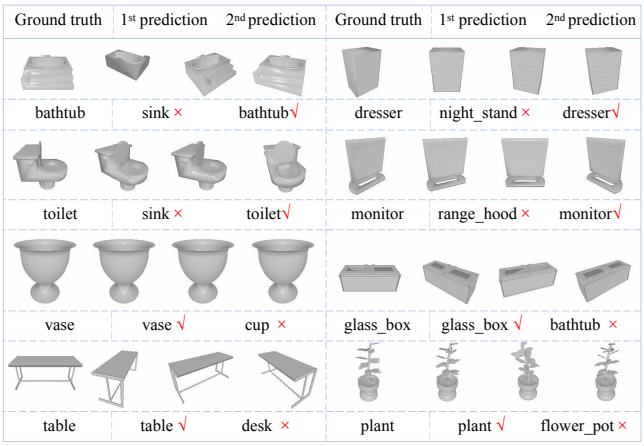

Figure 6: Visualization of successful and failed cases for second matching. Note that the text below the selected views represents the true label, the first-layer prediction, and the second-layer prediction, respectively.

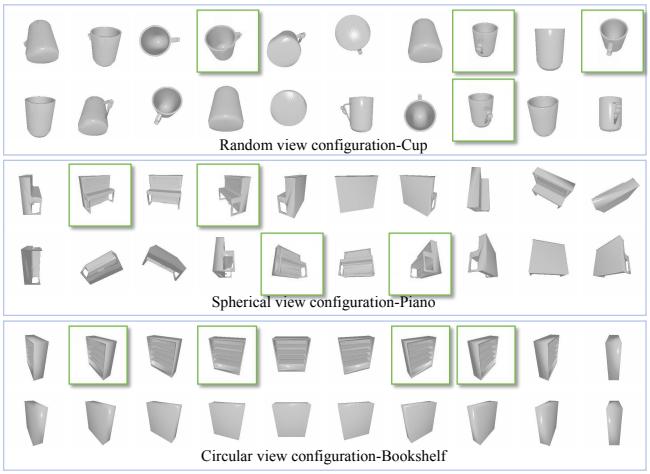

Figure 7: Visualization of multiple views from the models in ModelNet40. Note that the selected views are indicated by green boxes.

5 CONCLUSION

We design a zero-shot 3D recognition pipeline based on MV-CLIP to fully leverage the large-scale pre-trained models. We utilize the pre-trained visual encoder to evaluate the prediction confidence of the rendered multiple views, and explicitly select views with clearer semantics. In addition, we combine hand-crafted prompts with 3D-specific prompts powered by LLMs to form hierarchical prompts, which refine the first-layer prediction and achieve more accurate zero-shot performance via the second-layer matching. The experimental results demonstrate that we obtain superior performance of zero-shot 3D shape recognition by directly utilizing pre-trained models without any pre-training on 3D dataset, and this paper also discovers some interesting findings in prompt engineering.

REFERENCES

[1] Hyojin Bahng, Ali Jahanian, Swami Sankaranarayanan, and Phillip Isola. 2022. Visual prompting: Modifying pixel space to adapt pre-trained models. *arXiv preprint arXiv:2203.17274* 3 (2022), 11–12.

[2] Gary Bradski and Stephen Grossberg. 1994. Recognition of 3-d objects from multiple 2-d views by a self-organizing neural architecture. In *From Statistics to Neural Networks: Theory and Pattern Recognition Applications*. Springer, 349–375.

[3] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems* 33 (2020), 1877–1901.

[4] Angel X. Chang, Thomas A. Funkhouser, Leonidas J. Guibas, Pat Hanrahan, Qi-Xing Huang, Zimo Li, Silvio Savarese, Manolis Savva, Shuran Song, Hao Su, Jianxiong Xiao, Li Yi, and Fisher Yu. 2015. ShapeNet: An Information-Rich 3D Model Repository. *CoRR* abs/1512.03012 (2015). arXiv:1512.03012 http://arxiv.org/abs/1512.03012

[5] Ali Cheraghian, Shafin Rahman, Townim F Chowdhury, Dylan Campbell, and Lars Petersson. 2022. Zero-shot learning on 3d point cloud objects and beyond. *International Journal of Computer Vision* 130, 10 (2022), 2364–2384.

[6] Ali Cheraghian, Shafin Rahman, and Lars Petersson. 2019. Zero-shot learning of 3d point cloud objects. In *2019 16th International Conference on Machine Vision Applications (MVA)*. IEEE, 1–6.

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

[8] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805* (2018).

[9] Ankit Goyal, Hei Law, Bowei Liu, Alejandro Newell, and Jia Deng. 2021. Revisiting point cloud shape classification with a simple and effective baseline. In *International Conference on Machine Learning*. PMLR, 3809–3820.

[10] Zoey Guo, Yiwen Tang, Ray Zhang, Dong Wang, Zhigang Wang, Bin Zhao, and Xuelong Li. 2023. ViewRefer: Grasp the Multi-view Knowledge for 3D Visual Grounding. (2023), 15372–15383.

[11] Abdullah Hamdi, Faisal AlZahrani, Silvio Giancola, and Bernard Ghanem. 2022. MVTN: Learning Multi-View Transformations for 3D Understanding. arXiv:2212.13462 [cs.CV]

[12] Abdullah Hamdi, Silvio Giancola, and Bernard Ghanem. 2021. Mvtn: Multi-view transformation network for 3d shape recognition. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 1–11.

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

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

[15] Gabriel Ilharco, Mitchell Wortsman, Ross Wightman, Cade Gordon, Nicholas Carlini, Rohan Taori, Achal Dave, Vaishaal Shankar, Hongseok Namkoong, John Miller, et al. [n. d.]. Openclip, July 2021. *If you use this software, please cite it as below* 2, 4 ([n. d.]), 5.

[16] Menglin Jia, Luming Tang, Bor-Chun Chen, Claire Cardie, Serge Belongie, Bharath Hariharan, and Ser-Nam Lim. 2022. Visual prompt tuning. In *European Conference on Computer Vision*. Springer, 709–727.

[17] Zhengbao Jiang, Frank F Xu, Jun Araki, and Graham Neubig. 2020. How can we know what language models know? *Transactions of the Association for Computational Linguistics* 8 (2020), 423–438.

[18] Asako Kanezaki, Yasuyuki Matsushita, and Yoshifumi Nishida. 2018. Rotationnet: Joint object categorization and pose estimation using multiviews from unsupervised viewpoints. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 5010–5019.

[19] Muhammad Uzair Khattak, Hanoona Rasheed, Muhammad Maaz, Salman Khan, and Fahad Shahbaz Khan. 2023. Maple: Multi-modal prompt learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 19113–19122.

[20] Minghua Liu, Ruoxi Shi, Kaiming Kuang, Yinhao Zhu, Xuanlin Li, Shizhong Han, Hong Cai, Fatih Porikli, and Hao Su. 2023. OpenShape: Scaling Up 3D Shape Representation Towards Open-World Understanding. *arXiv preprint arXiv:2305.10764* (2023).

[21] Pengfei Liu, Weizhe Yuan, Jinlan Fu, Zhengbao Jiang, Hiroaki Hayashi, and Graham Neubig. 2023. Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing. *Comput. Surveys* 55, 9 (2023), 1–35.

[22] Norman Mu, Alexander Kirillov, David Wagner, and Saining Xie. 2022. Slip: Self-supervision meets language-image pre-training. In *European Conference on Computer Vision*. Springer, 529–544.

[23] Zachary Novack, Julian McAuley, Zachary Chase Lipton, and Saurabh Garg. 2023. Chils: Zero-shot image classification with hierarchical label sets. In *International Conference on Machine Learning*. PMLR, 26342–26362.

[24] Sarah Pratt, Ian Covert, Rosanne Liu, and Ali Farhadi. 2023. What does a platypus look like? generating customized prompts for zero-shot image classification. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 15691–15701.

[25] Zekun Qi, Runpei Dong, Guofan Fan, Zheng Ge, Xiangyu Zhang, Kaisheng Ma, and Li Yi. 2023. Contrast with Reconstruct: Contrastive 3D Representation Learning Guided by Generative Pretraining. *arXiv preprint arXiv:2302.02318* (2023).

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

[27] Aditya Sanghi, Pradeep Kumar Jayaraman, Arianna Rampini, Joseph G. Lambourne, Hooman Shayani, Evan Atherton, and Saeid Asgari Taghanaki. 2023. Sketch-A-Shape: Zero-Shot Sketch-to-3D Shape Generation. *CoRR* abs/2307.03869 (2023). https://doi.org/10.48550/ARXIV.2307.03869 arXiv:2307.03869

[28] Sitian Shen, Zilin Zhu, Linqian Fan, Harry Zhang, and Xinxiao Wu. 2023. DiffCLIP: Leveraging Stable Diffusion for Language Grounded 3D Classification. *arXiv preprint arXiv:2305.15957* (2023).

[29] Hang Su, Subhransu Maji, Evangelos Kalogerakis, and Erik Learned-Miller. 2015. Multi-view convolutional neural networks for 3d shape recognition. In *Proceedings of the IEEE international conference on computer vision*. 945–953.

[30] Yuting Su, Jiayu Li, Wenhui Li, Zan Gao, Haipeng Chen, Xuanya Li, and An-An Liu. 2022. Semantically guided projection for zero-shot 3D model classification and retrieval. *Multimedia Systems* 28, 6 (2022), 2437–2451.

[31] Eric Wallace, Shi Feng, Nikhil Kandpal, Matt Gardner, and Sameer Singh. 2019. Universal adversarial triggers for attacking and analyzing NLP. *arXiv preprint arXiv:1908.07125* (2019).

[32] Haowei Wang, Jiji Tang, Jiayi Ji, Xiaoshuai Sun, Rongsheng Zhang, Yiwei Ma, Minda Zhao, Lincheng Li, Zeng Zhao, Tangjie Lv, et al. 2023. Beyond First Impressions: Integrating Joint Multi-modal Cues for Comprehensive 3D Representation. In *Proceedings of the 31st ACM International Conference on Multimedia*. 3403–3414.

[33] Xin Wei, Ruixuan Yu, and Jian Sun. 2020. View-gcn: View-based graph convolutional network for 3d shape analysis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 1850–1859.

[34] Zhirong Wu, Shuran Song, Aditya Khosla, Fisher Yu, Linguang Zhang, Xiaoou Tang, and Jianxiong Xiao. 2015. 3d shapenets: A deep representation for volumetric shapes. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 1912–1920.

[35] Le Xue, Mingfei Gao, Chen Xing, Roberto Martín-Martín, Jiajun Wu, Caiming Xiong, Ran Xu, Juan Carlos Niebles, and Silvio Savarese. 2023. ULIP: Learning a unified representation of language, images, and point clouds for 3D understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 1179–1189.

[36] Le Xue, Ning Yu, Shu Zhang, Junnan Li, Roberto Martín-Martín, Jiajun Wu, Caiming Xiong, Ran Xu, Juan Carlos Niebles, and Silvio Savarese. 2023. ULIP-2: Towards Scalable Multimodal Pre-training For 3D Understanding. *arXiv preprint arXiv:2305.08275* (2023).

[37] Xumin Yu, Lulu Tang, Yongming Rao, Tiejun Huang, Jie Zhou, and Jiwen Lu. 2022. Point-bert: Pre-training 3d point cloud transformers with masked point modeling. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 19313–19322.

[38] Yihan Zeng, Chenhan Jiang, Jiageng Mao, Jianhua Han, Chaoqiang Ye, Qingqiu Huang, Dit-Yan Yeung, Zhen Yang, Xiaodan Liang, and Hang Xu. 2023. CLIP2: Contrastive Language-Image-Point Pretraining from Real-World Point Cloud Data. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 15244–15253.

[39] Junbo Zhang, Runpei Dong, and Kaisheng Ma. 2023. Clip-fo3d: Learning free open-world 3d scene representations from 2d dense clip. *arXiv preprint arXiv:2303.04748* (2023).

[40] Renrui Zhang, Ziyu Guo, Peng Gao, Rongyao Fang, Bin Zhao, Dong Wang, Yu Qiao, and Hongsheng Li. 2022. Point-m2ae: multi-scale masked autoencoders for hierarchical point cloud pre-training. *Advances in neural information processing systems* 35 (2022), 27061–27074.

[41] Renrui Zhang, Ziyu Guo, Wei Zhang, Kunchang Li, Xupeng Miao, Bin Cui, Yu Qiao, Peng Gao, and Hongsheng Li. 2022. PointCLIP: Point Cloud Understanding by CLIP. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2022, New Orleans, LA, USA, June 18-24, 2022*. IEEE, 8542–8552. https://doi.org/10.1109/CVPR52688.2022.00836

[42] Renrui Zhang, Xiangfei Hu, Bohao Li, Siyuan Huang, Hanqiu Deng, Yu Qiao, Peng Gao, and Hongsheng Li. 2023. Prompt, generate, then cache: Cascade of foundation models makes strong few-shot learners. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 15211–15222.

[43] Renrui Zhang, Liuhui Wang, Yu Qiao, Peng Gao, and Hongsheng Li. 2023. Learning 3d representations from 2d pre-trained models via image-to-point masked autoencoders. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 21769–21780.

[44] Hengshuang Zhao, Li Jiang, Jiaya Jia, Philip HS Torr, and Vladlen Koltun. 2021. Point transformer. In *Proceedings of the IEEE/CVF international conference on computer vision*. 16259–16268.

[45] Kaiyang Zhou, Jingkang Yang, Chen Change Loy, and Ziwei Liu. 2022. Conditional prompt learning for vision-language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 16816–16825.

[46] Kaiyang Zhou, Jingkang Yang, Chen Change Loy, and Ziwei Liu. 2022. Learning to prompt for vision-language models. *International Journal of Computer Vision* 130, 9 (2022), 2337–2348.

[47] Xiangyang Zhu, Renrui Zhang, Bowei He, Ziyu Guo, Ziyao Zeng, Zipeng Qin, Shanghang Zhang, and Peng Gao. 2023. Pointclip v2: Prompting clip and gpt for powerful 3d open-world learning. (2023), 2639–2650.