

AiDE: Improving 3D Open-Vocabulary Semantic Segmentation by Aligned Vision-Language Learning

Yimu Wang Krzysztof Czarnecki*
 University of Waterloo
 {yimu.wang, k2czarne}@uwaterloo.ca

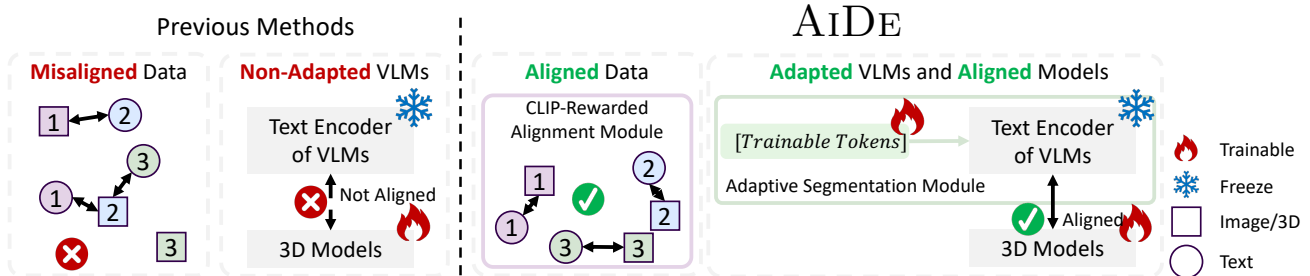


Figure 1. Previous methods use misaligned paired data (e.g., image/point cloud 1 is closest to text 2) and freeze the text encoder trained on 2D benchmark datasets, leading to misalignment between text encoders and 3D models and suboptimal performance. Our proposed AiDE aligns text encoders with 3D models by (i) generating better aligned 3D-image-to-text training data (CLIP-Rewarded Alignment Module) and (ii) optimizing deep trainable prompt tokens (Adaptive Segmentation Module).

Abstract

3D open-vocabulary semantic segmentation aims at recognizing countless categories beyond the limited set of annotations used in traditional settings. Due to the lack of large-scale 3D-vision-language segmentation data, instead of training models from scratch, the current solutions distill knowledge from pre-trained 2D vision-language models (VLMs) into 3D models. However, this distillation is supervised by misaligned 3D-scene-image-to-text data pairs, consequently leading to suboptimal performance. Moreover, as 2D VLMs are trained on 2D datasets, text encoders of VLMs, which serve as the bridge between 3D models and an unbounded set of categories, lack 3D semantics. In this paper, to address these issues and improve generalization performance, we propose an **Aligned 3D Open-Vocabulary Semantic Segmentation** framework, called AiDE, with two novel modules. To collect high-quality and well-aligned 3D-scene-image-to-text pairs, our CLIP-rewarded alignment module (i) generates diverse captions of multi-view images of 3D scenes to capture details by varying the temperatures and then (ii) samples captions based on their similarity to corresponding images for rich and accurate associations. Next, to adapt 2D VLMs to 3D contexts, our adaptive segmentation module introduces (iii) trainable tokens

within the input space and each layer of the text encoder, while freezing the text encoder to avoid catastrophic forgetting. Extensive experiments show that AiDE outperforms previous methods by a large margin on three representative benchmarks, demonstrating its effectiveness.

1. Introduction

3D semantic segmentation has attracted extensive research attention [16, 43, 64, 78, 83]. To handle real-world scenarios where new categories frequently emerge [46, 56, 61], researchers explore the open-vocabulary setting [23, 32, 79], which requires models to recognize novel classes beyond the typical limited training label space.

Due to the lack of large-scale 3D-image-text pairs, instead of training a 3D-language model from scratch, recent works [13, 23, 57] propose to transfer the knowledge encoded in pretrained 2D vision-language foundation models (VLMs), e.g., CLIP [59] and SAM [42], to 3D semantic segmentation models by aligning them with generated 3D-scene-image-to-text data. After the alignment, the 3D model with the VLM text encoder can handle unlimited categories. As a pioneering work, PLA [23] distills knowledge through captioning multi-view images [19, 68] of 3D scenes, allowing explicit association between 3D

* Corresponding author.

point clouds and captions. Following PLA, RegionPLC [79] boosts the performance by using dense object-level associations for better alignment.

However, these methods suffer from several major issues, making them suboptimal. First, the generated low-quality captions do not accurately match the corresponding image data of the 3D scenes semantically. As shown in Tab. 1a, when view images are used to retrieve captions using CLIP, recall@10 is lower than 0.034, indicating that less than 3.4% of images have their corresponding generated captions ranked in the top 10. This misalignment between captions and their corresponding images of 3D scenes—possibly stemming from the distribution shifts between 3D datasets [2,20] and 2D datasets [37,50] used for training captioners—hinders precise alignment between 3D models and text encoders. It further affects hIoU, mIoU^B, and mIoU^N as they are proportional to recall@10 (*i.e.*, the quality of data). Second, as datasets [17, 21, 37, 50] used for training VLMs differ from 3D datasets [2, 7, 20, 63], the alignment between 3D models and text encoders of VLMs exhibit excessive sensitivity to the prompts used for adapting text encoders, further leading to the unreliable performance as shown in Tab. 1b.

To address these issues, we propose a novel **Aligned 3D Open-Vocabulary SEMantic Segmentation** framework, called **AIDE**. Our key idea is to collect high-quality well-aligned data and adapt the text encoder to 3D contexts for better alignment between 3D models and text encoders, further improving the performance. Specifically, to generate well-aligned 3D-scene-image-to-text data pairs, our CLIP-rewarded alignment module first generates a variety of rich captions using different temperatures for enhancing caption diversity and capturing rich details in the 2D images of the 3D scenes (temperature-based generation)¹. Then, to encourage rich associations between 3D and text, we propose the CLIP-rewarded sampling method, which samples captions based on their similarity to the 3D-scene image in each training iteration. Next, drawing inspiration from previous methods [23,79], we employ a hierarchical alignment strategy, enabling alignment from scene-level (coarse-grained) to entity-level (fine-grained). Furthermore, to adapt text encoders for 3D semantics, the adaptive segmentation module extends beyond the popular visual prompt tuning methods [31, 38, 86] by incorporating trainable tokens not only in the input space but also across each transformer layer within text encoders, thereby enhancing its flexibility and effectiveness.

In summary, our contributions are as follows,

- We identify two significant challenges within existing methods, *i.e.*, the misalignment in 3D-scene-image-to-text data pairs and the need to adapt text encoders into

¹Higher temperatures lead to more diverse captions, while lower temperatures produce more deterministic ones.

| Captioner | R@10 ↑ | hIoU ↑ | mIoU ^B ↑ | mIoU ^N ↑ |
|----------------------|--------------|-------------|---------------------|---------------------|
| OFA [68] | 0.034 | 65.6 | 68.3 | 63.1 |
| ViT-GPT2 [19] | 0.004 | 65.3 | 68.3 | 62.4 |
| Best | 0.129 | 68.9 | 69.6 | 68.2 |
| Ours-Sampling | 0.151 | 70.3 | 69.9 | 70.6 |

(a) Impact of different captioning methods.

| Prompt Templates | hIoU ↑ | mIoU ^B ↑ | mIoU ^N ↑ |
|------------------|-------------|---------------------|---------------------|
| Identity | 65.3 | 68.3 | 62.4 |
| Simple | 64.3 | 67.7 | 61.3 |
| Full-ImageNet | 64.6 | 68.1 | 61.5 |
| Ours | 66.3 | 70.2 | 62.8 |

(b) Impact of different prompt templates.

Table 1. Performance of Semantic Segmentation on ScanNet (B15/N4 Split) [20] using PLA [23]. Metrics include harmonic IoU (hIoU), mIoU on base categories (mIoU^B), and mIoU on novel categories (mIoU^N), where base categories are annotated during training but novel categories are not. Table 1a illustrates that segmentation performance correlates with the recall at 10 (R@10) metric in view image-to-caption retrieval, which measures the portion of images with corresponding captions ranked in the top 10. “Best” indicates the selection of most-aligned captions generated by OFA and ViT-GPT2, based on cosine similarity. “Ours-Sampling” uses captions generated by our CLIP-rewarded alignment module. Table 1b highlights the importance of prompts for adapting 2D VLMs, with “Ours” being our proposed adaptive segmentation module. Details of templates are in Appendix D.

the 3D setting. To address these challenges, we propose AIDE, including the CLIP-rewarded alignment and adaptive segmentation modules.

- In the CLIP-rewarded alignment module, we generate high-quality 3D-scene-image-to-text pair data by varying the temperatures and sample captions based on their similarity to 3D-scene images to facilitate precise alignment. The adaptive segmentation module adapts the text encoder by integrating learnable prompts across the input space and each layer of text encoder.
- Extensive experiments on three representative benchmarks, *i.e.*, ScanNet [20], S3DIS [2], and one outdoor dataset (nuScenes [7]), demonstrate the superiority of AIDE compared to existing approaches across various metrics, highlighting its robustness and versatility.

2. Related Work

3D Open-Vocabulary Recognition. While vision-language models (VLMs) [39, 42, 47, 52, 59, 62, 70–73, 82] have achieved remarkable results in zero-shot or few-shot learning of 2D images by utilizing web-scale image-text data [8, 21, 50], the availability of such large scale data for 3D point clouds is limited. To extend this zero-

shot ability with VLMs in 3D point clouds, pioneering work [85, 89] has been working on converting point clouds into CLIP-recognizable images and aligning the point cloud features with the language features. Following these works, CLIP2Point [36] proposes to align the projected depth map with CLIP’s image space by a trainable depth encoder. This line of research focuses on aligning the 3D features with text and image features within CLIP’s representation space, further enhancing both the zero-shot and standard 3D recognition capability.

3D Zero-Shot and Open-Vocabulary Semantic Segmentation. This task aims to recognize novel classes that are not annotated in the training data. Early attempts [1, 15, 33, 54, 69, 80] extend effective 2D zero-shot methods [4, 6] into 3D scenarios for zero-shot 3D segmentation. 3DGenZ [54] shows that generated data can be used to boost the performance of zero-shot segmentation. Later, inspired by the remarkable advances in 2D open-vocabulary segmentation [12, 24, 26, 54, 55, 84, 88], Ding *et al.* [23] first propose PLA to distill knowledge encoded in VLMs using generated 3D-scene-image-to-caption data, allowing explicit associations between 3D and captions. Most of the recent works [22, 23, 32, 57, 79] have focused on improving the alignment between 3D and text representations with generated 3D-scene-image-to-text pair data using off-the-shelf image captioning methods [19, 48, 68].

Following these methods, our work also focuses on aligning 3D models and text encoders using 3D-scene-image-to-text pair data. However, our preliminary results in Tab. 1a show that the pairs are mismatched, which hinders previous methods from achieving a precise alignment between 3D models and text encoders. To address this, we introduce a novel temperature-based caption generation method and a similarity-based selection method (the CLIP-rewarded alignment module), which significantly enhances the quality of the pairs and, consequently, the model’s generalization capabilities. Furthermore, as the text encoders of 2D VLMs are trained on 2D datasets, as shown in Tab. 1b, they are not suitable for 3D scenarios. To mitigate this issue, the adaptive segmentation module is proposed to integrate trainable tokens within both the input space and each layer of the text encoder, thereby adapting it to 3D scenarios.

3. AIDE

3.1. Problem Definition

3D open-vocabulary semantic segmentation targets recognizing unseen categories, *i.e.*, those unannotated during training. Each 3D scene is represented by $(P, Y) = (\{\mathbf{p}_i\}_{i \in [N]}, \{\mathbf{y}_i\}_{i \in [N]})$, where N is the number of points, \mathbf{p} is a point, $\mathbf{y} \in \mathcal{Y}$ represents the corresponding label, and \mathcal{Y} contains all possible categories. Additionally, we have camera images $X = \{\mathbf{x}_i\}_{i \in [N_{\text{img}}]}$ for each 3D scene, where

N_{img} is the number of images available for the scene P . \mathcal{Y} is divided into base and novel classes, *i.e.*, \mathcal{Y}_B and \mathcal{Y}_N , respectively. During training, all points P are available, while only the labels Y_B in the base classes \mathcal{Y}_B are accessible. Meanwhile, the point annotations by novel classes and the names of these novel classes remain unknown. During inference, the model is provided with the names of all classes and is required to classify points belonging to them.

3.2. Preliminaries, Problems, and Solutions

Following previous works [23, 79], in AIDE, point-wise features $f_{3D}(P) \in \mathcal{R}^{N \times D}$ are extracted by a 3D backbone $f_{3D}(\cdot)$, where D represents feature dimensions. Then, a semantic segmentation classifier $f_{\text{seg}}(\cdot)$ is employed, producing the point-wise segmentation results $f_{\text{seg}}(f_{3D}(P))$. AIDE incorporates a text encoder $f_{\text{text}}(\cdot)$, *i.e.*, the text encoder of CLIP [59], to generate embeddings for captions and classes. To align 3D and text encoders, paired 3D-scene-image-to-text data are generated by an off-the-shelf captioner $f_{\text{cap}}(\cdot)$, *e.g.*, OFA [68], detailed in Sec. 3.3. The model is illustrated in Fig. 2.

Open-vocabulary segmentation [23, 32, 57]. By leveraging a classifier $f_{\text{seg}}(\cdot)$ with class-wise embedding $C \in \mathcal{R}^{|\mathcal{Y}| \times D}$ as weights, we obtain point-wise semantic segmentation prediction \hat{Y} ,

$$\hat{Y} = f_{\text{seg}}(f_{3D}(P)) = f_{3D}(P)C^T \in \mathcal{R}^{N \times |\mathcal{Y}|}, \quad (1)$$

where C is generated by the text encoder $f_{\text{text}}(\cdot)$ with prompts, *e.g.*, “a photo of a [CLASS]”, and class names, *e.g.*, table and chair. During training, the embedding C only contains the embeddings of the base categories \mathcal{Y}_B . We employ the Cross-Entropy loss $CE(\cdot, \cdot)$ to train the model,

$$\ell_{\text{seg}} = \sum_{i \in [N]} \mathcal{I}(\mathbf{y}_i \in \mathbf{Y}_B) CE(\hat{Y}_i, \mathbf{y}_i), \quad (2)$$

where $\mathcal{I}(\text{cond})$ is the indicator function ($\mathcal{I}(\text{cond}) = 1$ when cond is true) and \hat{Y}_i is the predicted label for the i -th point.

Confidence-based calibration. To counteract the tendency of models to exhibit overconfidence in base categories while ignoring novel categories [14, 23, 79], a confidence-based calibration branch $f_{\text{conf}}(\cdot)$ is employed. It dynamically balances the confidence level by predicting whether a point falls in the base categories. During training, we employ the binary cross-entropy loss $\text{BCE}(\cdot, \cdot)$ as,

$$\ell_{\text{conf}} = \text{BCE}(f_{\text{conf}}(f_{3D}(P)), Y_{\text{conf}}), \quad (3)$$

where Y_{conf} is the binary label (1 represents the point belonging to the base categories and vice versa). At inference, confidence calibration is applied to balance predictions on base and novel categories as $\hat{Y} = [f_{\text{conf}}(f_{3D}(P))\hat{Y}_B, (1 - f_{\text{conf}}(f_{3D}(P)))\hat{Y}_N]$, where \hat{Y}_B and \hat{Y}_N represent the predictions on base and novel categories, respectively.

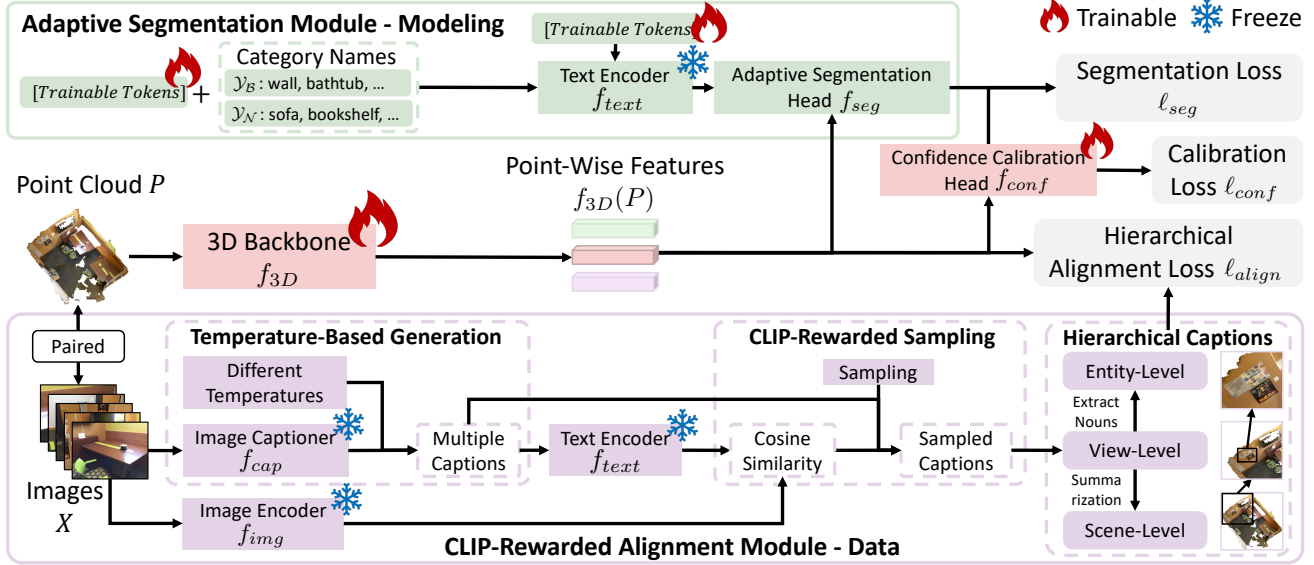


Figure 2. The illustration of our AIDE with two proposed modules, *i.e.*, CLIP-rewarded alignment module (Sec. 3.3) for enhancing the quality of 3D-text data pairs and adaptive segmentation module (Sec. 3.4) for adapting text encoders. **CLIP-Rewarded Alignment Module.** We first generate numerous captions using the temperature-based generation strategy, then sample captions based on their similarity to the images (CLIP-rewarded sampling), and finally align 3D point clouds and captions hierarchically. **Adaptive Segmentation Module.** To automatically find the most suitable prompt for adapting text encoders into 3D scenarios, AIDE extends prompt tuning [38] by incorporating learnable tokens in the input space and each layer of the text encoder f_{text} . A confidence calibration head is employed for balancing the confidence on base and novel categories, as models tend to overestimate their certainty in base categories [23, 28, 49, 53].

In this paper, we identify two problems in the current open-vocabulary segmentation pipeline [23, 32, 57] and propose corresponding solutions to mitigate them.

Problem 1 (data): Our preliminary results (Tab. 1a) show that 3D-scene-image-text paired data for 3D scenes are mismatched, hurting alignment of text encoders and 3D models. **Our solution:** To generate aligned data, we propose the CLIP-rewarded alignment module in Sec. 3.3 with temperature-based caption generation and CLIP-rewarded sampling.

Problem 2 (modelling): Our preliminary results (Tab. 1b) show that the alignment between text encoders and 3D models is highly sensitive to prompts for generating C , possibly due to the mismatch between 2D datasets that text encoders are trained on and 3D datasets. **Our solution:** To solve this issue and adapt text encoders by automatically finding the most suitable prompt, we propose the adaptive segmentation module elaborated in Sec. 3.4.

3.3. CLIP-Rewarded Alignment—Data Generation

Previous methods [13, 23, 57] align text encoders with 3D models using generated 3D-scene-image-to-text data, enabling 3D models to recognize novel categories. However, as shown in Tab. 1a, the images and the generated captions are mismatched, which leads to unsatisfying performance across both base and novel categories. A straightforward solution to this issue is to fine-tune captioners. However, the

lack of existing large-scale 3D-vision-text data makes it impractical. To solve this issue and generate high-quality data, we introduce a novel CLIP-rewarded alignment module as shown in Fig. 2. Specifically, we first generate several captions per image by the proposed **temperature-based generation**, which adjusts the temperature to encourage image captioners [11, 18, 19, 34, 68, 76] to produce diverse captions to capture details in the images. Next, to enable rich associations between 3D and text, we propose a novel **CLIP-rewarded sampling** method, which selects captions based on their similarity to the paired image data. Finally, inspired by previous methods [9, 23, 41, 52, 62, 79, 87], to enable coarse-to-fine alignment, we adopt hierarchical 3D-text alignment [23, 32].

Temperature-based caption generation. A widely accepted principle of text or caption generation [3, 65, 68] is that temperature controls the fidelity of the generated text. For image captioners, the i -th word w_i of generated captions are sampled from the probability $\sigma_{v \in \text{Vocab}}(P(v | I, w_{1:i-1})/\gamma)$, where γ is the temperature, w_i is the i -th word, I is the input image, $\sigma(\cdot)$ is the softmax function, $w_{1:i-1}$ represents the words before the i -th word, Vocab is the whole vocabulary, and $P(v | I, w_{1:i-1})$ is the prediction of v . As γ approaches 0, caption generation becomes more deterministic, and as γ increases, it becomes more stochastic. By leveraging this principle, the temperature-based generation varies temperatures γ during captioning to

foster caption diversity and reduce the mismatch in image-text data pairs. Specifically, we fix the captioner $f_{\text{cap}}(\cdot)$ and vary the temperature γ to get a set of captions $T_i = \{\text{caption}_j\}_{j \in [N_{\text{cap}}]}$ for the i -th view-image of the point cloud P , where N_{cap} is the number of captions.

CLIP-rewarded sampling. To avoid confusing models by matching one image with multiple captions, we need to obtain one caption. A simple way to get that caption is to select the best-aligned caption or summarize captions into one. However, previous studies [40, 51, 68] suggest that different captions represent different view perspectives of the image. To fully utilize multiple perspective captions, instead of using one perspective, we propose the CLIP-rewarded sampling method. It samples one caption in each iteration based on the similarity to the image/3D data, to capture details in all captions and highlight the details that exhibit high similarity. Specifically, we first calculate the similarity $s_{c2i,i}$ between the image \mathbf{x}_i and the set of captions T_i as $s_{c2i,i} = [\text{cos}(f_{\text{img}}(\mathbf{x}_i), f_{\text{text}}(\text{caption}_1)), \dots, \text{cos}(f_{\text{img}}(\mathbf{x}_i), f_{\text{text}}(\text{caption}_{N_{\text{cap}}}))]$, where $\text{cos}(\cdot, \cdot)$ is the cosine similarity and $f_{\text{img}}(\cdot)$ is the image encoder of the utilized VLM. Next, in each iteration, we sample one caption $T^{\text{view},i}$ from T_i based on the similarity $s_{c2i,i}$ and obtain the image-caption data pair (P, X, T^{view}) , where $T^{\text{view}} = \{T^{\text{view},i}\}_{i \in [N_{\text{img}}]}$ contains the sampled captions for all the view images in the point cloud P .

Hierarchical point cloud-text alignment. As suggested by previous 2D VLMs work [9, 23, 41, 52, 62, 87], hierarchical alignment, *e.g.*, region-level, image-level, and pixel-level alignment, is essential to cross-modal learning. To this end, we employ the hierarchical 3D-text alignment [23] to enable rich cross-modal associations with the hierarchical alignment loss ℓ_{align} from coarse-grained (scene-level) to fine-grained (entity-level). Details are deferred to Appendix B.1.

3.4. Adaptive Segmentation—Text Modeling

As shown in Tab. 1b, due to the domain shift between 3D datasets [2, 7, 20, 63] and the datasets [17, 21, 37, 50] 2D VLMs trained on, the performance of 3D open-vocabulary semantic segmentation models is highly sensitive to the prompt used for adapting the VLMs. One possible solution is fine-tuning VLMs to handle 3D data for better alignment between text encoders of VLMs and 3D models. However, due to catastrophic forgetting [35, 38, 66], where the encoder loses its prior knowledge while attempting to adapt to the new data distribution, fine-tuning VLMs is infeasible.

Drawing inspiration from visual prompt tuning [10, 38, 60, 74, 75, 81, 86], we focus on tailoring VLMs for 3D applications while freezing the model’s parameters to avoid compromising the integrity of VLMs’ knowledge, as shown in Fig. 2. To better adapt the text encoder to 3D scenarios, we introduce a small number of learnable tokens

$TOKENS$ at the input and every transformer layer in the text encoder $f_{\text{text}}(\cdot)$. Specifically, at the input layer, we directly concatenate the trainable tokens with the text as the input of the text encoder. Sequentially, for each transformer layer, trainable tokens are merged with the output of the previous layer as the input of the current layer, *i.e.*, $\text{concat}([TOKENS_i, F_{\text{text}, i-1}])$, where concat is the concatenation operation, $TOKENS_i$ is the i -th layer’s trainable token, and $F_{\text{text}, i-1}$ represents the output of the $(i-1)$ -th layer. The trainable tokens are updated using the overall training objective (Eq. (4)). During inference, we use the trainable tokens and the category names as the input of the text encoder $f_{\text{text}}(\cdot)$ to generate the category embedding C for the adaptive semantic segmentation head $f_{\text{seg}}(\cdot)$ as shown in Eq. (1).

3.5. Training Objective

The training objective of our proposed AIDE is a weighted linear combination of segmentation loss (Eq. (2)), confidence calibration loss (Eq. (3)), and hierarchical alignment loss (Eq. (5)) as follows:

$$\mathcal{L} = \beta^{\text{seg}} \ell_{\text{seg}} + \beta^{\text{align}} \ell_{\text{align}} + \beta^{\text{conf}} \ell_{\text{conf}}, \quad (4)$$

where β^{seg} , β^{align} , and β^{conf} are weight parameters.

4. Experiments

4.1. Benchmarks, Baselines, and Implementation

Benchmarks and category partitions. To validate the effectiveness of AIDE, we conducted extensive experiments on three popular 3D benchmarks: ScanNet [20], S3DIS [2], and one outdoor dataset (nuScenes [7]). For ScanNet, we split it into three base/novel partitions, *i.e.*, B15/N4, B12/N7, and B10/N9, where Bx/Ny refers to x base and y novel categories. As for S3DIS, we split it into 2 base/novel splits, *i.e.*, B8/N4, B6/N6. For nuScenes [7], we ignore the “otherflat” class and randomly divide the rest 15 classes into B12/N3 and B10/N5. Due to space limitations, the details of benchmarks and partitions are deferred to Appendix C.1.

Evaluation metrics. Following previous methods [23, 77, 79], we employ the commonly used 3D segmentation metric mean Intersection over Union for both base and novel categories (mIoU^{B} and mIoU^{N}), alongside the harmonic mean IoU (hIoU , $\text{hIoU} = (2 * \text{mIoU}^{\text{B}} * \text{mIoU}^{\text{N}}) / (\text{mIoU}^{\text{B}} + \text{mIoU}^{\text{N}})$) for evaluating base, novel categories and their harmonic mean.

Baselines. We compare AIDE with LSeg-3D [45], 3DGenZ [54], 3DTZSL [15], PLA [23], OpenScene [57], RegionPLC [79], and 3DPC-GZSL [80].

Implementation details. Following PLA [23], which is also our baseline, we employ the sparse-convolution-based UNet [29] with a base hidden dimension of 16 as our 3D backbone f_{3D} . We use CLIP (ViT-B16) [59] for the text and

| Methods | Venue | ScanNet | | | | | | | | |
|-------------------------------|---------|-------------|-------------------|-------------------|-------------|-------------------|-------------------|-------------|-------------------|-------------------|
| | | B15/N4 | | | B12/N7 | | | B10/N9 | | |
| | | hIoU | mIoU ^B | mIoU ^N | hIoU | mIoU ^B | mIoU ^N | hIoU | mIoU ^B | mIoU ^N |
| LSeg-3D [45] | ICLR'22 | 0.0 | 64.4 | 0.0 | 0.9 | 55.7 | 0.1 | 1.8 | 68.4 | 0.9 |
| 3DTZSL [15] | WACV'20 | 10.5 | 36.7 | 6.1 | 3.8 | 36.6 | 2.0 | 7.8 | 55.5 | 4.2 |
| 3DPC-GZSL [80] | ICCV'23 | 20.2 | 32.8 | 7.7 | - | - | - | - | - | - |
| 3DGenZ [54] | 3DV'21 | 20.6 | 56.0 | 12.6 | 19.8 | 35.5 | 13.3 | 12.0 | 63.6 | 6.6 |
| OpenScene [†] [57] | CVPR'23 | 67.1 | 68.8 | 62.8 | 56.8 | 61.5 | 51.7 | 55.7 | 71.8 | 43.6 |
| RegionPLC [79] | CVPR'24 | 69.4 | 68.2 | 70.7 | 68.2 | 69.9 | 66.6 | 64.3 | 76.3 | 55.6 |
| PLA (Baseline) [23] | CVPR'23 | 65.3 | 68.3 | 62.4 | 55.3 | 69.5 | 45.9 | 53.1 | 76.2 | 40.8 |
| AiDE | | 72.8 | 71.9 | 73.8 | 69.8 | 70.1 | 69.6 | 65.0 | 77.5 | 56.0 |
| Fully-Supervised [‡] | | 73.3 | 68.4 | 79.1 | 70.6 | 70.0 | 71.8 | 69.9 | 75.8 | 64.9 |

Table 2. Results on ScanNet. [†] and [‡] refer to numbers copied from He *et al.* [32] and Ding *et al.* [23]. Best in **Bold**.

| Methods | Venue | S3DIS | | | | | | nuScenes | | | | | |
|-------------------------------|---------|-------------|-------------------|-------------------|-------------|-------------------|-------------------|-------------|-------------------|-------------------|-------------|-------------------|-------------------|
| | | B8/N4 | | | B6/N6 | | | B12/N3 | | | B10/N5 | | |
| | | hIoU | mIoU ^B | mIoU ^N | hIoU | mIoU ^B | mIoU ^N | hIoU | mIoU ^B | mIoU ^N | hIoU | mIoU ^B | mIoU ^N |
| LSeg-3D [‡] [45] | ICLR'22 | 0.1 | 49.0 | 0.1 | 0 | 30.1 | 0 | 0.6 | 74.4 | 0.3 | 0 | 71.5 | 0 |
| 3DGenZ [‡] [54] | 3DV'21 | 8.8 | 50.3 | 4.8 | 9.4 | 20.3 | 6.1 | 1.6 | 53.3 | 0.8 | 1.9 | 44.6 | 1.0 |
| 3DTZSL [‡] [15] | WACV'20 | 8.4 | 43.1 | 4.7 | 3.5 | 28.2 | 1.9 | 1.2 | 21.0 | 0.6 | 6.4 | 17.1 | 3.9 |
| PLA (Baseline) [23] | CVPR'23 | 34.6 | 59.0 | 24.5 | 38.5 | 55.5 | 29.4 | 47.7 | 73.4 | 35.4 | 24.3 | 73.1 | 14.5 |
| AiDE | | 42.2 | 60.3 | 32.4 | 42.5 | 58.3 | 34.4 | 62.2 | 75.8 | 52.7 | 48.4 | 66.4 | 38.1 |
| Fully-Supervised [‡] | | 67.5 | 61.4 | 75.0 | 65.4 | 59.9 | 72.0 | 73.7 | 76.6 | 71.1 | 74.8 | 76.8 | 72.8 |

Table 3. Results on S3DIS and nuScenes. [‡] refers to numbers copied from Ding *et al.* [23]. Best in **Bold**.

image encoders (f_{text} and f_{img}) and BART [44] for text summarization. For a fair comparison, we use ViT-GPT2 [19] as the image captioner f_{cap} . Four trainable tokens are used in adapting text encoders. The weight parameters are adapted from our baseline [23].

4.2. Quantative Results

In this part, we present the results on ScanNet, S3DIS, and nuScenes in Tabs. 2 and 3, respectively.

ScanNet (Tab. 2). AiDE demonstrates superior performance across all metrics and splits, achieving hIoU of 72.8, 69.8, and 65.0 for B15/N4, B12/N7, and B10/N9 splits, respectively. This represents a significant improvement over PLA (Baseline), with increases of 7.5, 14.5, and 11.9 in hIoU. Notably, AiDE also narrows the gap with the fully supervised model, which represents the upper bound based on the backbone and training strategies.

S3DIS and nuScenes (Tab. 3). Notably, AiDE achieves the highest hIoU of 42.2 and 42.5 for B8/N4 and B6/N6 splits among various zero-shot learning-based and open-vocabulary segmentation methods, respectively, on S3DIS. Compared to our baseline, PLA, AiDE improves hIoU by 7.6 and 4.0 for each split. Improvements can also be observed on the outdoor dataset, nuScenes, as AiDE improves the hIoU from 47.7 and 24.3 to 62.2 and 48.4 on two different splits, showing the superiority of AiDE.

4.3. Ablation Studies

In this part, we present the ablation studies on the effects of two proposed modules (Tab. 4), hyperparameters (Tabs. 5 and 6), and the choice of text encoders (Tab. 7). Due to the space limitation, ablation studies on the choice of temperatures (Tab. 12 and Fig. 4), and the importance of aligning with image space (Tab. 16) are deferred to the Appendix. All experiments are conducted on ScanNet (B15/N4).

Impact of proposed modules (Tab. 4). Notably, the introduction of adaptive segmentation module alone improves hIoU from 65.3 to 66.3, and mIoU^B from 68.3 to 70.2, illustrating the efficacy in adapting the text encoders. ‘‘Caption Best’’ and ‘‘Caption Sampling’’ refer to using temperature-based generation and then selecting the best-aligned captions or captions sampled by the CLIP-rewarded sampling (the CLIP-rewarded alignment module). Specifically, while ‘‘Caption Best’’ improves hIoU to 68.9, ‘‘Caption Sampling’’ further boosts hIoU to 70.3 from 65.3. It underscores the importance of well-designed captioning techniques in improving alignment with the text encoder. Combining two modules together reaches the highest results. Specifically, combining the adaptive segmentation module with ‘‘Caption Best’’ achieves a hIoU of 70.4, while integrating it with ‘‘Caption Sampling’’ remarkably advances hIoU, mIoU^B, and mIoU^N to 72.8, 71.9, and 73.8, respectively.

Numbers of learnable tokens. To understand how the length of (deep) learnable tokens impacts performance, we

| Adaptive Segmentation | Caption | | ScanNet(B15/N4) | | |
|-----------------------|---------|----------|-----------------|-------------------|-------------------|
| | Best | Sampling | hIoU | mIoU ^B | mIoU ^N |
| | | | 65.3 | 68.3 | 62.4 |
| ✓ | | | 66.3 | 70.2 | 62.8 |
| | ✓ | | 68.9 | 69.6 | 68.2 |
| | | ✓ | 70.3 | 69.9 | 70.6 |
| ✓ | ✓ | | 70.4 | 71.0 | 69.9 |
| ✓ | | ✓ | 72.8 | 71.9 | 73.8 |
| Fully-Supervised | | | 73.3 | 68.4 | 79.1 |

Table 4. Ablation studies on different modules of AiDE. “Adaptive Segmentation” refers to the adaptive segmentation module. “Caption Selection” and “Caption Sampling” refer to using temperature-based generation and then selecting the best-aligned captions or sampling captions based on their similarity to the images (CLIP-rewarded alignment module).

| # of Learnable Prompts | ScanNet (B15/N4) | | |
|------------------------|------------------|-------------------|-------------------|
| | hIoU | mIoU ^B | mIoU ^N |
| 0 (Baseline) | 65.3 | 68.3 | 62.4 |
| 2 | 71.9 | 70.9 | 72.9 |
| 4 | 72.8 | 71.9 | 73.8 |
| 8 | 71.8 | 72.2 | 71.0 |
| 16 | 71.2 | 72.4 | 70.0 |
| Fully-Supervised | | | |
| | 73.3 | 68.4 | 79.1 |

Table 5. Ablation studies on different numbers of learnable tokens of AiDE on ScanNet (B15/N4).

conduct a series of experiments as shown in Tab. 5. It is clear that increasing the number of learnable tokens from the baseline (zero prompts) significantly enhances the model’s performance across all metrics (hIoU, mIoU^B, and mIoU^N). The optimal point is achieved with four learnable tokens, with the highest hIoU and mIoU^N of 72.8 and 73.8. However, further increasing the number of prompts to 8 or 16 leads to a slight performance decline, which might be due to potential overfitting on seen categories.

Number of captions N_{ca} . While the temperature used in caption generation affects the diversity of the generated captions (Tab. 12), the number of captions (samples) generated for each temperature also matters. Thus, to understand the impact, we vary the number of samples for each temperature and present the results in Tab. 6. Notably, the peak performance is observed at 30 captions, with hIoU, mIoU^B, and mIoU^N of 72.8, 71.9, and 73.8. Also, we observe consistent improvement when increasing the number of samples from 1 to 30, underscoring the value of leveraging more descriptive and diverse captions to enhance performance. However, it also illustrates a diminishing return beyond this optimal point, as evidenced by a decrease in all metrics when the number of captions is further increased to 50. Noisy caption generation might be the reason behind

| # of Generated Captions N_{cap} | ScanNet (B15/N4) | | |
|--|------------------|-------------------|-------------------|
| | hIoU | mIoU ^B | mIoU ^N |
| 1 (Baseline) | 65.3 | 68.3 | 62.4 |
| 10 | 71.9 | 71.3 | 72.5 |
| 20 | 72.7 | 72.5 | 73.0 |
| 30 | 72.8 | 71.9 | 73.8 |
| 50 | 70.2 | 69.6 | 70.8 |
| Fully-Supervised | | | |
| | 73.3 | 68.4 | 79.1 |

Table 6. Ablation studies on different numbers of captions of AiDE for each temperature.

| Text Encoder | ScanNet (B15/N4) | | |
|------------------------|------------------|-------------------|-------------------|
| | hIoU | mIoU ^B | mIoU ^N |
| AiDE w. CLIP [59] | 72.8 | 71.9 | 73.8 |
| AiDE w. ImageBind [27] | 71.0 | 70.4 | 71.7 |
| AiDE w. PointBind [30] | 70.0 | 70.7 | 69.4 |
| Fully-Supervised | | | |
| | 73.3 | 68.4 | 79.1 |

Table 7. Ablation studies on using different text encoders of AiDE on ScanNet (B15/N4).

this phenomenon. As generating over 30 captions per temperature will result in a total of more than 120 captions and the training epochs are 128, the majority of these captions cannot be effectively utilized during training. It suggests that an optimal balance between the diversity and quality of captions is crucial.

Choice of text encoder. In this part, we compare the performance using three text encoders from multimodal foundational models, *e.g.*, CLIP [59], ImageBind [27], and PointBind [30], as shown in Tab. 7. The results indicate that the CLIP text encoder outperforms others across all metrics, with the highest hIoU (72.8), mIoU^B (71.9), and mIoU^N (73.8). This superiority likely stems from CLIP’s adeptness at aligning visual and textual representations, a critical factor in open-vocabulary segmentation tasks.

4.4. Qualitative Results—Generalization

To illustrate the effectiveness of our proposed AiDE, we present qualitative results on analyzing the open-vocabulary ability to segment point clouds with the synonyms and hypernyms of classes (Tab. 8) and the zero-shot domain transfer ability (Tab. 9). The interpretation of learnable prompts in the input space (Tab. 15), the quality of generated captions (Fig. 5), and generalization on instance segmentation (Tab. 17) are deferred to the Appendix.

Generalization on segmenting with synonyms and hypernyms of classes. VLMs have demonstrated a remarkable capacity for associating semantically similar words. To understand how well this ability is transferred to 3D models, we evaluate AiDE and baseline’s performance in recognizing synonyms and hypernyms of original class names, as

| Methods | hIoU | mIoU ^B | mIoU ^N | IoU on Base Categories | | | | IoU on Novel | |
|-----------------------------|--------------|-------------------|-------------------|------------------------|--------------|--------------|--------------|--------------|--------------|
| | | | | floor | bed | window | sink | desk | toilet |
| <i>Original Class Names</i> | | | | | | | | | |
| Baseline | 0.645 | 0.679 | 0.615 | 0.950 | 0.808 | 0.617 | 0.571 | 0.449 | 0.594 |
| AiDE | 0.728 | 0.719 | 0.738 | 0.979 | 0.818 | 0.658 | 0.656 | 0.513 | 0.862 |
| <i>Synonyms</i> | | | | | | | | | |
| Baseline | 0.289 | 0.207 | 0.479 | 0.138 | 0.566 | 0.346 | 0.000 | 0.242 | 0.464 |
| AiDE | 0.340 | 0.243 | 0.568 | 0.352 | 0.739 | 0.326 | 0.000 | 0.304 | 0.746 |
| <i>Hypernyms</i> | | | | | | | | | |
| Baseline | 0.311 | 0.230 | 0.478 | 0.000 | 0.401 | 0.344 | 0.000 | 0.425 | 0.143 |
| AiDE | 0.364 | 0.264 | 0.588 | 0.000 | 0.637 | 0.378 | 0.000 | 0.474 | 0.464 |

Table 8. Open-vocabulary semantic segmentation results on ScanNet (B15/N4) using the original class names, their synonyms, and hypernyms. The full table is presented in Tab. 14, while the synonyms and hypernyms of class names are presented in Tab. 13.

| Train Dataset | Metrics (Baseline/AiDE) | | |
|--------------------------------|-------------------------|---------------------------|-------------------|
| | hIoU | mIoU ^B | mIoU ^N |
| Test Dataset: S3DIS (B8/N4) | | | |
| ScanNet (B15/N4) | 32.1/ 35.9 | 31.6/ 39.9 | 32.6/ 33.8 |
| ScanNet (B12/N7) | 22.2/ 25.8 | 25.0 /23.3 | 19.9/ 28.9 |
| ScanNet (B10/N9) | 24.7/ 31.0 | 30.5/ 38.9 | 20.7/ 25.7 |
| Test Dataset: ScanNet (B15/N4) | | | |
| S3DIS (B8/N4) | 10.5/ 11.9 | 15.0 / 15.0 | 8.1/ 9.9 |
| S3DIS (B6/N6) | 5.9/ 7.5 | 7.1/ 7.6 | 5.1/ 7.5 |

Table 9. Zero-shot transfer ability of baseline and AiDE. We train models on ScanNet (B15/N4) or S3DIS (B8/N4) and test them on another. Best in **bold**.

| | # of Trainable Parameters | # of Parameters | Throughput (Scene/s) |
|----------|---------------------------|----------------------|----------------------|
| Baseline | 11,001,346 | 74,138,114 | 7.58 |
| AiDE | 11,027,970 (0.2% ↑) | 74,164,738 (0.03% ↑) | 7.58 |

Table 10. Comparison of parameters and throughput between the baseline and AiDE.

shown in Tab. 8. Results show that AiDE consistently surpasses the baseline across various metrics for most of the classes. When using synonyms and hypernyms, both methods experience a drop in performance compared to original class names, which is reasonable due to the increased difficulty in matching these broader or alternative terms with points precisely. On the other side, AiDE still maintains a lead over the baseline, demonstrating its robustness to variations in vocabulary. The drop is mostly in the synonyms category, especially with base categories, e.g., “floor”, “curtain”, and “sink”. It also highlights the challenge of segmentation with hypernyms, as the baseline and AiDE have (near-)zero IoU for several categories, indicating difficulty in generalizing to broader category terms.

Generalization on zero-shot domain transfer. To understand the zero-shot domain transfer ability of AiDE, we conduct experiments where models are trained on either ScanNet (B15/N4) or S3DIS (B8/N4) and tested on the other, as shown in Tab. 9. In every setting, AiDE consistently outperforms the baseline on hIoU, underscoring its superior generalization capability. Notably, when training on ScanNet (B15/N4) and testing on S3DIS (B8/N4), AiDE

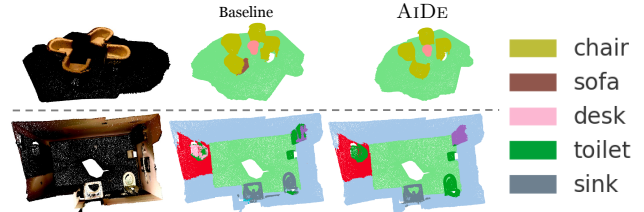


Figure 3. Qualitative results of segmentation compared between baseline and AiDE.

achieves significant improvements in all metrics, with hIoU, mIoU^B, and mIoU^N increasing from 32.1, 31.6, and 32.6 to 35.9, 39.9, and 33.8, respectively. Similar patterns are observed with huge gaps between the baseline and AiDE in different settings. These results underscore the importance of the CLIP-rewarded alignment and adaptive segmentation modules in enhancing open-vocabulary segmentation models’ transferability to novel categories and scenarios.

Visualization. To better understand how our AiDE excels at segmenting seen and unseen objects, we visualize segmentation results in Figs. 3 and 6. It is obvious that, on both the seen and unseen classes (chair and toilet), AiDE better segments them from other objects compared with the baseline.

Parameters and throughput comparison. We also include a parameter and throughput analysis in Tab. 10. We notice that, as AiDE only introduces sets of trainable tokens to adapt the text encoder, the additional parameters and latency are marginal compared with our baseline model.

5. Conclusion

We introduced AiDE to collect well-aligned 3D-vision-and-text multimodal data and efficiently adapt 2D VLMs for 3D semantic segmentation, thereby enhancing the models’ generalization capabilities. AiDE has two key components: (i) the CLIP-rewarded alignment module, using temperature-based caption generation combined with CLIP-rewarded sampling to generate well-aligned 3D-vision-and-text data, and (ii) adaptive segmentation module, adding a small set of learnable tokens in both the input space and each layer of text encoder to adapt the VLM text encoder to the 3D setting. Our experimental results demonstrated AiDE’s superiority over previous methods, indicating the importance of high-quality data generation and the adaptation of text encoders.

Limitation. While we employ scene-level, view-level, and entity-level alignments, an even more fine-grained alignment could achieve better alignment [9, 41, 52, 62, 87]. Currently, the point-wise classification method is limited to cosine similarity matching. In the future, an advanced text-3D fusion method should be applied to integrate multimodal information for segmentation.

References

- [1] Ahmed Abdelreheem, Ivan Skorokhodov, Maks Ovsjanikov, and Peter Wonka. SATR: Zero-Shot Semantic Segmentation of 3D Shapes. In *IEEE/CVF International Conference on Computer Vision*, pages 15120–15133, 2023. [3](#)
- [2] Iro Armeni, Ozan Sener, Amir R. Zamir, Helen Jiang, Ioannis Brilakis, Martin Fischer, and Silvio Savarese. 3D Semantic Parsing of Large-Scale Indoor Spaces. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1534–1543, 2016. [2](#), [5](#), [14](#)
- [3] Akari Asai, Zeqiu Wu, Yizhong Wang, Avirup Sil, and Hannaneh Hajishirzi. Self-RAG: Learning to Retrieve, Generate, and Critique through Self-Reflection, 2023. arXiv:2310.11511 [cs]. [4](#)
- [4] Donghyeon Baek, Youngmin Oh, and Bumsub Ham. Exploiting a Joint Embedding Space for Generalized Zero-Shot Semantic Segmentation. In *IEEE/CVF International Conference on Computer Vision*, pages 9536–9545, 2021. [3](#)
- [5] Steven Bird, Ewan Klein, and Edward Loper. *Natural language processing with Python: analyzing text with the natural language toolkit*. ” O’Reilly Media, Inc.”, 2009. [13](#)
- [6] Maxime Bucher, Tuan-Hung VU, Matthieu Cord, and Patrick Pérez. Zero-Shot Semantic Segmentation. In *Advances in Neural Information Processing Systems*, 2019. [3](#)
- [7] Holger Caesar, Varun Bankiti, Alex H. Lang, Sourabh Vora, Venice Erin Liong, Qiang Xu, Anush Krishnan, Yu Pan, Giancarlo Baldan, and Oscar Beijbom. nuScenes: A Multimodal Dataset for Autonomous Driving. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 11618–11628. IEEE, 2020. [2](#), [5](#), [14](#)
- [8] Holger Caesar, Jasper R. R. Uijlings, and Vittorio Ferrari. COCO-Stuff: Thing and Stuff Classes in Context. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1209–1218, 2018. [2](#)
- [9] Zhaowei Cai, Gukyeong Kwon, Avinash Ravichandran, Erhan Bas, Zhuowen Tu, Rahul Bhotika, and Stefano Soatto. X-DETR: A Versatile Architecture for Instance-wise Vision-Language Tasks, 2022. arXiv:2204.05626 [cs]. [4](#), [5](#), [8](#), [13](#)
- [10] Rui Cao, Roy Ka-Wei Lee, Wen-Haw Chong, and Jing Jiang. Prompting for multimodal hateful meme classification. In Yoav Goldberg, Zornitsa Kozareva, and Yue Zhang, editors, *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 321–332, Abu Dhabi, United Arab Emirates, Dec. 2022. Association for Computational Linguistics. [5](#)
- [11] David M. Chan, Austin Myers, Sudheendra Vijayanarasimhan, David A. Ross, and John Canny. \$IC^3\$: Image Captioning by Committee Consensus, 2023. arXiv:2302.01328 [cs]. [4](#)
- [12] Jun Chen, Deyao Zhu, Guocheng Qian, Bernard Ghanem, Zhicheng Yan, Chenchen Zhu, Fanyi Xiao, Sean Chang Culatana, and Mohamed Elhoseiny. Exploring Open-Vocabulary Semantic Segmentation from CLIP Vision Encoder Distillation Only. In *IEEE/CVF International Conference on Computer Vision*, pages 699–710, 2023. [3](#)
- [13] Runnan Chen, Youquan Liu, Lingdong Kong, Xinge Zhu, Yuexin Ma, Yikang Li, Yuenan Hou, Yu Qiao, and Wenping Wang. CLIP2Scene: Towards label-efficient 3D scene understanding by CLIP. In *IEEE/CVF conference on computer vision and pattern recognition (CVPR)*, pages 7020–7030, 2023. [1](#), [4](#)
- [14] Zhimin Chen, Longlong Jing, Yingwei Li, and Bing Li. Bridging the domain gap: Self-supervised 3D scene understanding with foundation models. In *Thirty-seventh conference on neural information processing systems*, 2023. [3](#)
- [15] Ali Cheraghian, Shafin Rahman, Dylan Campbell, and Lars Petersson. Transductive Zero-Shot Learning for 3D Point Cloud Classification. In *IEEE Winter Conference on Applications of Computer Vision*, pages 912–922, 2020. [3](#), [5](#), [6](#), [14](#)
- [16] Christopher Choy, JunYoung Gwak, and Silvio Savarese. 4D Spatio-Temporal ConvNets: Minkowski Convolutional Neural Networks. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3070–3079, 2019. [1](#)
- [17] Tat-Seng Chua, Jinhui Tang, Richang Hong, Haojie Li, Zhiping Luo, and Yantao Zheng. NUS-WIDE: a real-world web image database from National University of Singapore. In *CIVR*, pages 1–9, 2009. [2](#), [5](#)
- [18] Sanghyuk Chun, Wonjae Kim, Song Park, Minsuk Chang, and Seong Joon Oh. ECCV Caption: Correcting False Negatives by Collecting Machine-and-Human-verified Image-Caption Associations for MS-COCO. In *Computer Vision – ECCV 2022*, volume 13668, pages 1–19, 2022. [4](#)
- [19] NLP Connect. vit-gpt2-image-captioning, 2022. [1](#), [2](#), [3](#), [4](#), [6](#)
- [20] Angela Dai, Angel X. Chang, Manolis Savva, Maciej Halber, Thomas Funkhouser, and Matthias Niessner. ScanNet: Richly-Annotated 3D Reconstructions of Indoor Scenes. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 2432–2443. IEEE, 2017. [2](#), [5](#), [14](#)
- [21] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. ImageNet: A large-scale hierarchical image database. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 248–255, 2009. [2](#), [5](#)
- [22] Runyu Ding, Jihan Yang, Chuhui Xue, Wenqing Zhang, Song Bai, and Xiaojuan Qi. Lowis3D: Language-driven open-world instance-level 3D scene understanding, 2023. arXiv: 2308.00353 [cs.CV]. [3](#)
- [23] Runyu Ding, Jihan Yang, Chuhui Xue, Wenqing Zhang, Song Bai, and Xiaojuan Qi. PLA: Language-Driven Open-Vocabulary 3D Scene Understanding. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7010–7019, 2023. [1](#), [2](#), [3](#), [4](#), [5](#), [6](#), [13](#), [14](#), [17](#), [18](#)
- [24] Xiaoyi Dong, Jianmin Bao, Yinglin Zheng, Ting Zhang, Dongdong Chen, Hao Yang, Ming Zeng, Weiming Zhang, Lu Yuan, Dong Chen, Fang Wen, and Nenghai Yu. MaskCLIP: Masked Self-Distillation Advances Contrastive Language-Image Pretraining. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10995–11005, 2023. [3](#)
- [25] Christiane Fellbaum. *WordNet: An Electronic Lexical Database*. Bradford Books, 1998. [16](#)
- [26] Golnaz Ghiasi, Xiuye Gu, Yin Cui, and Tsung-Yi Lin. Scaling Open-Vocabulary Image Segmentation with Image-Level Labels, 2022. arXiv:2112.12143 [cs]. [3](#)

- [27] Rohit Girdhar, Alaaeldin El-Nouby, Zhuang Liu, Mannat Singh, Kalyan Vasudev Alwala, Armand Joulin, and Ishan Misra. ImageBind: One Embedding Space To Bind Them All, 2023. arXiv:2305.05665 [cs]. [7](#)
- [28] Parker Glenn, Cassandra L. Jacobs, Marvin Thielk, and Yi Chu. The viability of best-worst scaling and categorical data label annotation tasks in detecting implicit bias. In Gavin Abercrombie, Valerio Basile, Sara Tonelli, Verena Rieser, and Alexandra Uma, editors, *Proceedings of the 1st Workshop on Perspectivist Approaches to NLP @LREC2022*, pages 32–36, Marseille, France, June 2022. European Language Resources Association. [4](#)
- [29] Benjamin Graham, Martin Engelcke, and Laurens Van Der Maaten. 3D Semantic Segmentation with Submanifold Sparse Convolutional Networks. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9224–9232, 2018. [5](#)
- [30] Ziyu Guo, Renrui Zhang, Xiangyang Zhu, Yiwen Tang, Xianzheng Ma, Jiaming Han, Kexin Chen, Peng Gao, Xianzhi Li, Hongsheng Li, and Pheng-Ann Heng. Point-Bind & Point-LLM: Aligning Point Cloud with Multi-modality for 3D Understanding, Generation, and Instruction Following, 2023. arXiv:2309.00615 [cs]. [7](#)
- [31] Jiale Han, Shuai Zhao, Bo Cheng, Shengkun Ma, and Wei Lu. Generative prompt tuning for relation classification. In Yoav Goldberg, Zornitsa Kozareva, and Yue Zhang, editors, *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 3170–3185, Abu Dhabi, United Arab Emirates, Dec. 2022. Association for Computational Linguistics. [2](#)
- [32] Qingdong He, Jinlong Peng, Zhengkai Jiang, Kai Wu, Xiaozhong Ji, Jiangning Zhang, Yabiao Wang, Chengjie Wang, Mingang Chen, and Yunsheng Wu. UniM-OV3D: Uni-Modality Open-Vocabulary 3D Scene Understanding with Fine-Grained Feature Representation, 2024. arXiv:2401.11395 [cs]. [1](#), [3](#), [4](#), [6](#), [13](#), [17](#), [18](#)
- [33] Shuting He, Xudong Jiang, Wei Jiang, and Henghui Ding. Prototype Adaption and Projection for Few- and Zero-Shot 3D Point Cloud Semantic Segmentation. *IEEE Transactions on Image Processing*, 32:3199–3211, 2023. [3](#)
- [34] MD. Zakir Hossain, Ferdous Sohel, Mohd Fairuz Shiratuddin, and Hamid Laga. A Comprehensive Survey of Deep Learning for Image Captioning. *ACM Computing Surveys*, 51(6), 2019. [4](#)
- [35] Siteng Huang, Biao Gong, Yulin Pan, Jianwen Jiang, Yiliang Lv, Yuyuan Li, and Donglin Wang. VoP: Text-Video Co-Operative Prompt Tuning for Cross-Modal Retrieval. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6565–6574, 2023. [5](#)
- [36] Tianyu Huang, Bowen Dong, Yunhan Yang, Xiaoshui Huang, Rynson W.H. Lau, Wanli Ouyang, and Wangmeng Zuo. CLIP2Point: Transfer CLIP to point cloud classification with image-depth pre-training. In *IEEE/CVF international conference on computer vision (ICCV)*, pages 22157–22167, 2023. [3](#)
- [37] Mark J. Huiskes and Michael S. Lew. The MIR Flickr Retrieval Evaluation. In *ACM International Conference on Multimedia Information Retrieval*, pages 39–43, 2008. [2](#), [5](#)
- [38] Menglin Jia, Luming Tang, Bor-Chun Chen, Claire Cardie, Serge Belongie, Bharath Hariharan, and Ser-Nam Lim. Visual Prompt Tuning, 2022. arXiv:2203.12119 [cs]. [2](#), [4](#), [5](#)
- [39] Xiangru Jian and Yimu Wang. InvGC: Robust cross-modal retrieval by inverse graph convolution. In Houda Bouamor, Juan Pino, and Kalika Bali, editors, *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 836–865, Singapore, Dec. 2023. Association for Computational Linguistics. [2](#)
- [40] Ming Jiang, Qiuyuan Huang, Lei Zhang, Xin Wang, Pengchuan Zhang, Zhe Gan, Jana Diesner, and Jianfeng Gao. TIGER: Text-to-image grounding for image caption evaluation. In Kentaro Inui, Jing Jiang, Vincent Ng, and Xiaojun Wan, editors, *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 2141–2152, Hong Kong, China, Nov. 2019. Association for Computational Linguistics. [5](#)
- [41] Wonjae Kim, Bokyung Son, and Ildoo Kim. ViLT: Vision-and-Language Transformer Without Convolution or Region Supervision. In *International Conference on Machine Learning*, pages 5583–5594, 2021. [4](#), [5](#), [8](#), [13](#)
- [42] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C. Berg, Wan-Yen Lo, Piotr Dollár, and Ross Girshick. Segment Anything, 2023. arXiv:2304.02643 [cs]. [1](#), [2](#)
- [43] Xin Lai, Yukang Chen, Fanbin Lu, Jianhui Liu, and Jiaya Jia. Spherical Transformer for LiDAR-Based 3D Recognition. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 17545–17555, 2023. [1](#)
- [44] Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In Dan Jurafsky, Joyce Chai, Natalie Schluter, and Joel Tetreault, editors, *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7871–7880, Online, July 2020. Association for Computational Linguistics. [6](#), [13](#)
- [45] Boyi Li, Kilian Q. Weinberger, Serge Belongie, Vladlen Koltun, and Rene Ranftl. Language-driven Semantic Segmentation. In *International Conference on Learning Representations*, 2022. [5](#), [6](#), [14](#)
- [46] Jiale Li, Hang Dai, Hao Han, and Yong Ding. Mseg3d: Multi-modal 3d semantic segmentation for autonomous driving. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 21694–21704, 2023. [1](#)
- [47] Junnan Li, Dongxu Li, Caiming Xiong, and Steven Hoi. BLIP: Bootstrapping Language-Image Pre-training for Unified Vision-Language Understanding and Generation, 2022. arXiv:2201.12086 [cs]. [2](#)
- [48] Wei Li, Linchao Zhu, Longyin Wen, and Yi Yang. DeCap: Decoding CLIP Latents for Zero-Shot Captioning via Text-Only Training. 2023. [3](#)

- [49] Yan Li, Ethan X. Fang, Huan Xu, and Tuo Zhao. Implicit Bias of Gradient Descent based Adversarial Training on Separable Data. 2019. [4](#)
- [50] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. Microsoft COCO: Common Objects in Context. In *Computer Vision – ECCV 2014*, pages 740–755, 2014. [2](#), [5](#)
- [51] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual Instruction Tuning. 2023. [5](#)
- [52] Jiasen Lu, Dhruv Batra, Devi Parikh, and Stefan Lee. ViLBERT: Pretraining Task-Agnostic Visiolinguistic Representations for Vision-and-Language Tasks. In *Advances in Neural Information Processing Systems*, 2019. [2](#), [4](#), [5](#), [8](#), [13](#)
- [53] Kaifeng Lyu and Jian Li. Gradient Descent Maximizes the Margin of Homogeneous Neural Networks. In *International Conference on Learning Representations*, 2019. [4](#)
- [54] Björn Michele, Alexandre Boulch, Gilles Puy, Maxime Bucher, and Renaud Marlet. Generative Zero-Shot Learning for Semantic Segmentation of 3D Point Clouds. In *International Conference on 3D Vision*, pages 992–1002, 2021. [3](#), [5](#), [6](#), [14](#)
- [55] Jie Oin, Jie Wu, Pengxiang Yan, Ming Li, Ren Yuxi, Xuefeng Xiao, Yitong Wang, Rui Wang, Shilei Wen, Xin Pan, and Xingang Wang. FreeSeg: Unified, Universal and Open-Vocabulary Image Segmentation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 19446–19455, 2023. [3](#)
- [56] Youngmin Park, Vincent Lepetit, and Woontack Woo. Multiple 3d object tracking for augmented reality. In *2008 7th IEEE/ACM International Symposium on Mixed and Augmented Reality*, pages 117–120, 2008. [1](#)
- [57] Songyou Peng, Kyle Genova, Chiyu “Max” Jiang, Andrea Tagliasacchi, Marc Pollefeys, and Thomas Funkhouser. OpenScene: 3D Scene Understanding With Open Vocabularies. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 815–824, 2023. [1](#), [3](#), [4](#), [5](#), [6](#), [17](#), [18](#)
- [58] Charles R. Qi, Hao Su, Kaichun Mo, and Leonidas J. Guibas. PointNet: Deep Learning on Point Sets for 3D Classification and Segmentation. *arXiv*, 2017. *arXiv:1612.00593 [cs]*. [14](#)
- [59] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning Transferable Visual Models From Natural Language Supervision. In *International Conference on Machine Learning*, pages 8748–8763, 2021. [1](#), [2](#), [3](#), [5](#), [7](#), [14](#)
- [60] Yongming Rao, Wenliang Zhao, Guangyi Chen, Yansong Tang, Zheng Zhu, Guan Huang, Jie Zhou, and Jiwen Lu. DenseCLIP: Language-Guided Dense Prediction with Context-Aware Prompting. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 18061–18070, 2022. [5](#)
- [61] Zhu Rui and Zhao Yongjia. Real-time plane segmentation for scene understanding in robot navigation. In *2017 10th International Congress on Image and Signal Processing, BioMedical Engineering and Informatics (CISP-BMEI)*, pages 1–5, 2017. [1](#)
- [62] Amanpreet Singh, Ronghang Hu, Vedanuj Goswami, Guillaume Couairon, Wojciech Galuba, Marcus Rohrbach, and Douwe Kiela. FLAVA: A Foundational Language And Vision Alignment Model. *arXiv:2112.04482 [cs]*, 2021. *arXiv:2112.04482*. [2](#), [4](#), [5](#), [8](#), [13](#)
- [63] Pei Sun, Henrik Kretzschmar, Xerxes Dotiwalla, Aurelien Chouard, Vijaysai Patnaik, Paul Tsui, James Guo, Yin Zhou, Yuning Chai, Benjamin Caine, Vijay Vasudevan, Wei Han, Jiquan Ngiam, Hang Zhao, Aleksei Timofeev, Scott Etinger, Maxim Krivokon, Amy Gao, Aditya Joshi, Yu Zhang, Jonathon Shlens, Zhifeng Chen, and Dragomir Anguelov. Scalability in perception for autonomous driving: Waymo open dataset. In *IEEE/CVF conference on computer vision and pattern recognition*, 2020. [2](#), [5](#)
- [64] Hugues Thomas, Charles R. Qi, Jean-Emmanuel Deschaud, Beatriz Marcotegui, Francois Goulette, and Leonidas Guibas. KPConv: Flexible and Deformable Convolution for Point Clouds. In *IEEE/CVF International Conference on Computer Vision*, pages 6410–6419, 2019. [1](#)
- [65] Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, Dan Bikel, Lukas Blecher, Cristian Canton Ferrer, Moya Chen, Guillem Cucurull, David Esiobu, Jude Fernandes, Jeremy Fu, Wenyin Fu, Brian Fuller, Cynthia Gao, Vedanuj Goswami, Naman Goyal, Anthony Hartshorn, Saghar Hosseini, Rui Hou, Hakan Inan, Marcin Kardas, Viktor Kerkez, Madian Khabsa, Isabel Kloumann, Artem Korenev, Punit Singh Koura, Marie-Anne Lachaux, Thibaut Lavril, Jenya Lee, Diana Liskovich, Yinghai Lu, Yuning Mao, Xavier Martinet, Todor Mihaylov, Pushkar Mishra, Igor Molybog, Yixin Nie, Andrew Poulton, Jeremy Reizenstein, Rashi Rungta, Kalyan Saladi, Alan Schelten, Ruan Silva, Eric Michael Smith, Ranjan Subramanian, Xiaoqing Ellen Tan, Binh Tang, Ross Taylor, Adina Williams, Jian Xiang Kuan, Puxin Xu, Zheng Yan, Iliyan Zarov, Yuchen Zhang, Angela Fan, Melanie Kambadur, Sharan Narang, Aurelien Rodriguez, Robert Stojnic, Sergey Edunov, and Thomas Scialom. Llama 2: Open Foundation and Fine-Tuned Chat Models, 2023. *arXiv:2307.09288 [cs]*. [4](#)
- [66] Lifu Tu, Caiming Xiong, and Yingbo Zhou. Prompt-tuning can be much better than fine-tuning on cross-lingual understanding with multilingual language models. In Yoav Goldberg, Zornitsa Kozareva, and Yue Zhang, editors, *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 5478–5485, Abu Dhabi, United Arab Emirates, Dec. 2022. Association for Computational Linguistics. [5](#)
- [67] Thang Vu, Kookhoi Kim, Tung M. Luu, Thanh Nguyen, and Chang D. Yoo. SoftGroup for 3D instance segmentation on point clouds. In *IEEE/CVF conference on computer vision and pattern recognition*, pages 2708–2717, 2022. [17](#)
- [68] Peng Wang, An Yang, Rui Men, Junyang Lin, Shuai Bai, Zhikang Li, Jianxin Ma, Chang Zhou, Jingren Zhou, and Hongxia Yang. OFA: Unifying Architectures, Tasks, and Modalities Through a Simple Sequence-to-Sequence Learning Framework. In *International Conference on Machine Learning*, pages 23318–23340, 2022. [1](#), [2](#), [3](#), [4](#), [5](#)

- [69] Yuanbin Wang, Shaofei Huang, Yulu Gao, Zhen Wang, Rui Wang, Kehua Sheng, Bo Zhang, and Si Liu. Transferring CLIP’s knowledge into zero-shot point cloud semantic segmentation. In *ACM international conference on multimedia*, pages 3745–3754, 2023. [3](#)
- [70] Yimu Wang, Xiangru Jian, and Bo Xue. Balance act: Mitigating hubness in cross-modal retrieval with query and gallery banks. In Houda Bouamor, Juan Pino, and Kalika Bali, editors, *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 10542–10567, Singapore, Dec. 2023. Association for Computational Linguistics. [2](#)
- [71] Yimu Wang and Peng Shi. Video-text retrieval by supervised sparse multi-grained learning. In Houda Bouamor, Juan Pino, and Kalika Bali, editors, *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 633–649, Singapore, Dec. 2023. Association for Computational Linguistics. [2](#)
- [72] Yimu Wang, Bo Xue, Quan Cheng, Yuhui Chen, and Lijun Zhang. Deep unified cross-modality hashing by pairwise data alignment. In Zhi-Hua Zhou, editor, *International Joint Conference on Artificial Intelligence*, pages 1129–1135, 2021. [2](#)
- [73] Yimu Wang, Shuai Yuan, Xiangru Jian, Wei Pang, Mushi Wang, and Ning Yu. Havtr: Improving video-text retrieval through augmentation using large foundation models. *CoRR*, abs/2404.05083, 2024. [2](#)
- [74] Jingyuan Wen, Yutian Luo, Nanyi Fei, Guoxing Yang, Zhiwu Lu, Hao Jiang, Jie Jiang, and Zhao Cao. Visual prompt tuning for few-shot text classification. In Nicoletta Calzolari, Chu-Ren Huang, Hansaem Kim, James Pustejovsky, Leo Wanner, Key-Sun Choi, Pum-Mo Ryu, Hsin-Hsi Chen, Lucia Donatelli, Heng Ji, Sadao Kurohashi, Patrizia Paggio, Nianwen Xue, Seokhwan Kim, Younggyun Hahm, Zhong He, Tony Kyungil Lee, Enrico Santus, Francis Bond, and Seung-Hoon Na, editors, *Proceedings of the 29th International Conference on Computational Linguistics*, pages 5560–5570, Gyeongju, Republic of Korea, Oct. 2022. International Committee on Computational Linguistics. [5](#)
- [75] Chen Henry Wu, Saman Motamed, and Shaunak Srivastava. Generative Visual Prompt: Unifying Distributional Control of Pre-Trained Generative Models. [5](#)
- [76] Jianzong Wu, Xiangtai Li, Henghui Ding, Xia Li, Guangliang Cheng, Yunhai Tong, and Chen Change Loy. Betrayed by Captions: Joint Caption Grounding and Generation for Open Vocabulary Instance Segmentation. In *IEEE/CVF International Conference on Computer Vision*, pages 21938–21948, 2023. [4](#)
- [77] Yongqin Xian, Subhabrata Choudhury, Yang He, Bernt Schiele, and Zeynep Akata. Semantic Projection Network for Zero- and Few-Label Semantic Segmentation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8248–8257, 2019. [5](#)
- [78] Mutian Xu, Runyu Ding, Hengshuang Zhao, and Xiaojuan Qi. PAConv: Position Adaptive Convolution with Dynamic Kernel Assembling on Point Clouds. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3172–3181, 2021. [1](#)
- [79] Jihan Yang, Runyu Ding, Zhe Wang, and Xiaojuan Qi. RegionPLC: Regional Point-Language Contrastive Learning for Open-World 3D Scene Understanding, 2023. arXiv:2304.00962 [cs]. [1](#), [2](#), [3](#), [4](#), [5](#), [6](#), [13](#)
- [80] Yuwei Yang, Munawar Hayat, Zhao Jin, Hongyuan Zhu, and Yinjie Lei. Zero-Shot Point Cloud Segmentation by Semantic-Visual Aware Synthesis. In *IEEE/CVF International Conference on Computer Vision*, pages 11552–11562, 2023. [3](#), [5](#), [6](#)
- [81] Hantao Yao, Rui Zhang, and Changsheng Xu. Visual-Language Prompt Tuning with Knowledge-Guided Context Optimization. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6757–6767, 2023. [5](#)
- [82] Qiyang Yu, Yang Liu, Yimu Wang, Ke Xu, and Jingjing Liu. Multimodal federated learning via contrastive representation ensemble. In *The Eleventh International Conference on Learning Representations*, 2023. [2](#)
- [83] Cheng Zhang, Haocheng Wan, Xinyi Shen, and Zizhao Wu. PVT: Point-Voxel Transformer for Point Cloud Learning, 2022. arXiv:2108.06076 [cs]. [1](#)
- [84] Junbo Zhang, Runpei Dong, and Kaisheng Ma. CLIP-FO3D: Learning free open-world 3D scene representations from 2D dense CLIP. In *IEEE/CVF international conference on computer vision workshops (ICCVW)*, pages 2040–2051, 2023. [3](#)
- [85] Renrui Zhang, Ziyu Guo, Wei Zhang, Kunchang Li, Xupeng Miao, Bin Cui, Yu Qiao, Peng Gao, and Hongsheng Li. PointCLIP: Point Cloud Understanding by CLIP. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8542–8552, 2022. [3](#)
- [86] Kaiyang Zhou, Jingkang Yang, Chen Change Loy, and Ziwei Liu. Conditional prompt learning for vision-language models. In *IEEE/CVF conference on computer vision and pattern recognition (CVPR)*, 2022. [2](#), [5](#)
- [87] Mingyang Zhou, Luowei Zhou, Shuohang Wang, Yu Cheng, Linjie Li, Zhou Yu, and Jingjing Liu. UC2: Universal Cross-lingual Cross-modal Vision-and-Language Pre-training. *CoRR*, abs/2104.00332, 2021. arXiv: 2104.00332. [4](#), [5](#), [8](#), [13](#)
- [88] Muzhi Zhu, Hengtao Li, Hao Chen, Chengxiang Fan, Weian Mao, Chenchen Jing, Yifan Liu, and Chunhua Shen. Seg-Prompt: Boosting Open-World Segmentation via Category-Level Prompt Learning. In *IEEE/CVF International Conference on Computer Vision*, pages 999–1008, 2023. [3](#)
- [89] Xiangyang Zhu, Renrui Zhang, Bowei He, Ziyu Guo, Ziyao Zeng, Zipeng Qin, Shanghang Zhang, and Peng Gao. Point-CLIP V2: Prompting CLIP and GPT for powerful 3D open-world learning. In *IEEE/CVF international conference on computer vision*, pages 2639–2650, 2023. [3](#)

Appendix of AIDE: Improving 3D Open-Vocabulary Semantic Segmentation by Aligned Vision-Language Learning

In the Appendix, we present additional details on the model and experimental results in Appendix C including ablation studies, qualitative results (e.g., generated captions), and generalization to instance segmentation. Prompt templates used for producing Tab. 1b are detailed in Appendix D.

A. Societal Impact

In this paper, we propose a novel 3D open-vocabulary segmentation framework aimed at recognizing countless classes of objects to improve the generalization of segmentation models. We did not identify any obvious negative societal impacts. Instead, we hope our model can contribute to the development of reliable and generalizable machine learning models and advance progress in this area.

B. Model Details

B.1. Hierarchical Point Cloud-Text Alignment

After obtaining the paired data, we are ready to align text encoders with 3D models. As suggested by previous 2D VLMs work [9, 23, 41, 52, 62, 87], hierarchical alignment, e.g., region-level, image-level, and pixel-level alignment, is essential to cross-modal learning. To this end, we employ hierarchical 3D-text alignment [23, 32] to enable rich cross-modal association. Specifically, we have scene-, view-, and entity-level alignments. For **scene-level alignment** (coarse-grained), we first obtain a comprehensive scene-level caption $\mathbf{t}^{\text{scene}}$ with a text summarization model [44] to aggregate view-level captions T^{view} . Then, we directly maximize the similarity between the entire point clouds and the scene-level caption as,

$$\ell_{\text{align}}^{\text{scene}} = \ell_{\text{pdc}}(f_{3D}(P), f_{\text{text}}(\mathbf{t}^{\text{scene}})),$$

where ℓ_{pdc} is the point-discriminative contrastive loss detailed below (Eq. (6)). For **view-level alignment**, we maximize the similarity between the points \hat{P}_i^{view} visible in an image (view) and its corresponding view-level caption T_i^{view} ,

$$\ell_{\text{align}}^{\text{view}} = \sum_{i \in [|T^{\text{view}}|]} \ell_{\text{pdc}}(f_{3D}(\hat{P}_i^{\text{view}}), f_{\text{text}}(T_i^{\text{view}})).$$

For **entity-level alignment**, following PLA [23], we get nouns as the entity-level captions T^{entity} of each view by employing NLTK [5] on the view-level captions T^{view} . Next, to associate a specific set of points (instead of the view-level set of points) with the entity-level captions, by the set differences and intersections of two adjacent views i and j , we

obtain two entity-level pairs,

$$\begin{aligned} & (\hat{P}_{i \setminus j}^{\text{entity}}, \hat{T}_{i \setminus j}^{\text{entity}}), (\hat{P}_{i \cap j}^{\text{entity}}, \hat{T}_{i \cap j}^{\text{entity}}), \forall i, j \in [|T^{\text{view}}|], \\ & \text{s.t.}, \sigma_{\min} < |\hat{P}_{i * j}^{\text{entity}}| < \sigma_{\max} \min(|\hat{P}_i^{\text{view}}|, |\hat{P}_j^{\text{view}}|), \\ & |\hat{T}_{i * j}^{\text{entity}}| > 0, \forall * \in \{\setminus, \cap\}, \end{aligned}$$

where \setminus and \cap represent the set difference and intersection, σ_{\min} and σ_{\max} are two hyperparameters, $\hat{P}_{i * j}^{\text{entity}} = \hat{P}_i^{\text{view}} * \hat{P}_j^{\text{view}}$, and $\hat{T}_{i * j}^{\text{entity}} = T_i^{\text{entity}} * T_j^{\text{entity}}$. In total, we have $2 \binom{|T^{\text{view}}|}{2}$ entity-level associations, and now, we are ready to perform the entity-level alignment as,

$$\ell_{\text{align}}^{\text{entity}} = \sum_{i \in [2 \binom{|T^{\text{view}}|}{2}]} \ell_{\text{pdc}}(f_{3D}(\hat{P}_i^{\text{entity}}), f_{\text{text}}(\hat{T}_i^{\text{entity}})).$$

Summarizing three losses, we have the hierarchical 3D-text alignment loss as,

$$\ell_{\text{align}} = \alpha^{\text{scene}} \ell_{\text{align}}^{\text{scene}} + \alpha^{\text{view}} \ell_{\text{align}}^{\text{view}} + \alpha^{\text{entity}} \ell_{\text{align}}^{\text{entity}}, \quad (5)$$

where α^{scene} , α^{view} , and α^{entity} are weighted parameters.

B.2. Point-Discriminative Contrastive Learning

To maximize the similarity between the sets of point features and caption features, following previous studies [23, 79], we employ the point-discriminative contrastive loss [79] ℓ_{pdc} instead of the vanilla contrastive loss. Specifically, to encourage point-wise alignment, instead of grouping different-level point cloud features by mean pooling and applying contrastive loss, we first calculate point-wise contrastive-style activations and then group these activations,

$$\ell_{\text{pdc}}(F^P, F^T) = \frac{-\sum_{i \in [N_{FP}]} \ln \frac{\exp(F_i^P F^{T\top})}{\exp(F_i^P F^{T\top}) + \sum_j \exp(F_i^P \hat{F}_j^{T\top})}}{N_{FP}}, \quad (6)$$

where F_i^P represents the i -th point's feature, N_{FP} is the number of points in $F^P \in \mathcal{R}^{N_{FP} \times D}$, and $\hat{F}^T \in \mathcal{R}^{1 \times D}$ represents captions in the same training batch.

C. Experiments

C.1. Benchmarks, Category Partition, and Baselines

Benchmarks. Here we present the details of two representative benchmarks we use in our experiments.

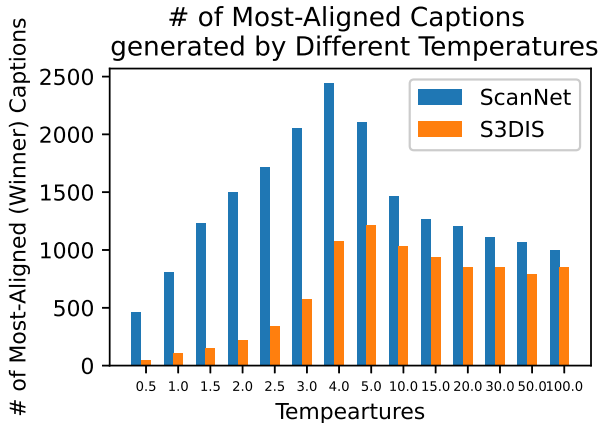


Figure 4. The number of most-aligned captions generated by different temperatures for each image. We use temperatures of 0.5, 1.0 (standard temperature), 1.5, 2.0, 2.5, 3.0, 4.0, 5.0, 10.0, 15.0, 20.0, 30.0, 50.0, and 100.0 for generating diverse captions.

ScanNet [20] consists of 1,613 scenes (1,201 scenes for training, 312 scenes for validation, and 100 for testing) densely annotated in 20 classes. We discard the “otherfurniture” class and split the rest 19 classes into three partitions for semantic segmentation, as shown in Tab. 11.

S3DIS [2] contains 271 scans across 6 building areas and 13 categories. Following previous work [23, 58], we treat the 5-th area as the validation split and other areas as the training split. We discard the “clutter” class and partition the rest 12 classes into two partitions for both semantic segmentation and instance segmentation, as demonstrated in Tab. 11.

nuScenes [7] comprises 1000 driving scenes. 850 scenes of them form the training and validation set, and the other 150 scenes are for testing. It contains 16 semantic classes for the LiDAR semantic segmentation task. The scene is scanned by 32-line LiDAR, which is different from the previous two datasets.

Baselines. Following PLA [23], we use LSeg-3D [45] (UNet as the backbone, vision-language adapter implemented by MLP, and the CLIP [59] ViT-B/16 text encoder). For 3DGenZ [54] and 3DTZSL [15], we use the same setting with PLA [23].

Implementation details. We use 4 learnable tokens in the input space and each layer of the text encoder for enabling the adaptive segmentation module. For each temperature, we generate 30 captions enabling the sampling.

C.2. Ablation Studies

In this part, we extend our ablation studies to the choice of temperatures (Tab. 12 and Fig. 4) and the importance of aligning with image space (Tab. 16).

Which temperature generates the captions that are

closest to the corresponding 3D/image data? To better understand the impact of varying temperatures in generating paired 3D-(image)-text data, we evaluate the similarity between generated captions and their corresponding images. Specifically, we identify the temperature setting that produces the most aligned captions for each image, which we refer to as the “winner” temperature, and then count the time each temperature “wins”. In our experiments, we use fourteen different temperatures, *i.e.*, {0.5, 1.0, 1.5, 2.0, 2.5, 3.0, 4.0, 5.0, 10.0, 15.0, 20.0, 30.0, 50.0, 100.0}, and generate 30 captions for each temperature. The results are summarized in Fig. 4, which indicates that temperatures of 4.0 and 5.0 frequently emerge as winners. This observation aligns with the fact that higher temperatures tend to introduce more variability, resulting in noisier captions, whereas lower temperatures lead to more deterministic and potentially less diverse captions. Therefore, a moderate temperature setting is optimal for generating the best-aligned captions and aligned 3D-(image)-text data.

Impact of numbers of temperatures used for generation. To understand how the diversity of temperatures used for generating captions affects the performance, we employ four different temperature sets and evaluate the performance of AiDE on ScanNet (B15/N4) as shown in Tab. 12. A diverse set of temperatures significantly enhances AiDE’s performance over the baseline, which only uses one temperature, *i.e.*, 1.0, with the optimal range being {0.5, 1.0, 1.5, 2.0, 3.0, 4.0, 5.0}, which slightly outperforms other settings with hIoU, mIoU^B, and mIoU^N scores of 73.0, 71.7, and 74.3, respectively. This finding aligns perfectly with the observations in Fig. 4 as temperatures 4.0 and 5.0 yield the most wins. Furthermore, increasing the temperature diversity further, even up to a wide range of {0.5, 1.0, 1.5, 2.0, 3.0, 4.0, 5.0, 10.0, 15.0, 20.0, 30.0, 50.0, 100.0}, results in diminishing returns, indicating a threshold beyond which additional temperature variance does not contribute to, and may even detract from performance. That might be because most captions will not be utilized even once throughout the training process. As the number of training epochs is 128, employing 14 temperatures results in a total of 420 captions, with most captions having a low probability of being sampled.

C.3. Qualitative Results

In this part, we extend our qualitative results on the interpretation of learnable prompts in the input space (Tab. 15), the quality of generated captions (Fig. 5), and generalization on instance segmentation (Tab. 17).

Generated captions. To understand how the proposed CLIP-rewarded alignment module improves the quality of generated 3D-text data, we randomly choose some samples of captions presented in Fig. 5. It is obvious that captions

| Partition | Base Categories | Novel Categories |
|-----------|--|---|
| ScanNet | | |
| B15/N4 | wall, floor, cabinet, bed, chair, table, door, window, picture, counter, curtain, refrigerator, showercurtain, sink, bathtub | sofa, bookshelf, desk, toilet |
| B12/N7 | wall, floor, cabinet, sofa, door, window, counter, desk, curtain, refrigerator, showercurtain, toilet | bed, chair, table, bookshelf, picture, sink, bathtub |
| B10/N9 | wall, floor, cabinet, bed, chair, sofa, table, door, window, curtain | bookshelf, picture, counter, desk, refrigerator, showercurtain, toilet, sink, bathtub |
| S3DIS | | |
| B8/N4 | ceiling, floor, wall, beam, column, door, chair, board | window, table, sofa, bookcase |
| B6/N6 | ceiling, wall, beam, column, chair, bookcase | floor, window, door, table, sofa, board |

Table 11. Category partitions for open-vocabulary semantic segmentation on ScanNet and S3DIS.

| Temperatures used for Caption Generation | ScanNet (B15/N4) | | |
|--|------------------|-------------------|-------------------|
| | hIoU | mIoU ^B | mIoU ^N |
| {1.0} (Baseline) | 65.3 | 68.3 | 62.4 |
| {0.5, 1.0, 1.5, 2.0} | 72.8 | 71.9 | 73.8 |
| {0.5, 1.0, 1.5, 2.0, 3.0, 4.0, 5.0} | 73.0 | 71.7 | 74.3 |
| {0.5, 1.0, 1.5, 2.0, 3.0, 4.0, 5.0, 10.0, 15.0, 20.0, 30.0} | 72.6 | 71.7 | 73.6 |
| {0.5, 1.0, 1.5, 2.0, 3.0, 4.0, 5.0, 10.0, 15.0, 20.0, 30.0, 50.0, 100.0} | 71.5 | 71.4 | 71.6 |
| Fully-Supervised | 73.3 | 68.4 | 79.1 |

Table 12. Ablation studies of different temperatures used for caption generation of AIDE.

| ScanNet | | | | | | | | | | |
|-------------|------------|-------------|------------|---------|--------------|---------------|----------|----------|---------|-----------|
| Class Names | wall | floor | cabinet | bed | chair | sofa | table | door | window | bookshelf |
| Synonyms | palisade | deck | locker | bunk | bench | couch | board | exit | window | bookrack |
| Hypernyms | embankment | compartment | - | - | seat | - | - | entrance | - | shelf |
| Class Names | picture | counter | desk | curtain | refrigerator | showercurtain | toilet | sink | bathtub | |
| Synonyms | figure | - | escritoire | drapery | icebox | - | bathroom | - | - | |
| Hypernyms | image | - | - | - | - | - | room | - | - | |
| S3DIS | | | | | | | | | | |
| Class Names | ceiling | floor | wall | beam | column | window | door | table | chair | sofa |
| Synonyms | roof | deck | palisade | - | - | - | exit | board | bench | couch |
| Synonyms | cap | compartment | embankment | - | - | - | entrance | - | seat | - |
| Class Names | bookcase | board | | | | | | | | |
| Synonyms | - | plank | | | | | | | | |
| Hypernyms | - | - | | | | | | | | |

Table 13. Synonyms and hypernyms of class names on ScanNet and S3DIS used for generating class embeddings. For classes without suitable synonyms or hypernyms, we use the original class name as their synonyms and hypernyms and mark “-” in the table.

generated by lower temperatures (closer to the baseline of 1.0) might be more generic and closely tied to the most apparent elements in the images. In contrast, higher temperatures could lead to more diverse and potentially creative interpretations of the same visual content. Moreover, high temperature always leads to informative generated captions, e.g., “a wood desk topped with a computer and a yellow and

white striped arm chair in front of the computer and the desk is full of clutter next to a very messy desk” for the second example. And high temperatures can capture key information too as 2.0 captures “blurry” in the second image while 1.0 can barely capture that. Meanwhile, as captions generated by higher temperatures are more informative, they exhibit higher similarity to the images than the baseline setting

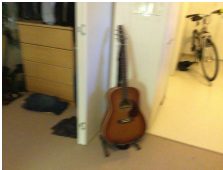
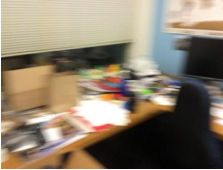
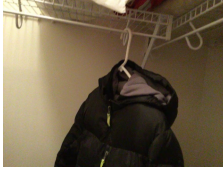

| Images | Similarity | Temperatures | Captions |
|---|------------|----------------|--|
|  | 0.3748 | 4.0 | a guitar is sitting on the floor next to a closet |
| | 0.3682 | 1.5 | a guitar is sitting on the floor next to a door |
| | 0.3380 | 2.0 | a small closet with a guitar and a skateboard |
| | ... | ... | ... |
| | 0.2219 | Baseline (1.0) | a black and white cat standing next to a door |
|  | 0.4023 | 2.0 | a blurry picture of a cluttered office space |
| | 0.3784 | 2.0 | a blurry picture of a cluttered computer desk with many items on the desk |
| | 0.3237 | 4.0 | a wood desk topped with a computer and a yellow and white striped arm chair in front of the computer and the desk is full of clutter next to a very messy desk |
| | ... | ... | ... |
| | 0.3157 | Baseline (1.0) | a cluttered desk with a computer on it |
|  | 0.3833 | 1.5 | a black jacket and black pants hanging from a rack in a closet |
| | 0.3520 | 2.0 | a black jacket is hanging on the handlebars of a black jacket hanging on a rope in a closet |
| | 0.3433 | 5.0 | a black jacket resting against a wall while a dog stands by a backboard to hang inside of a wall with cables next to it is hanging on two handles in a garage window with two open shelves on the ground near a metal fence and some hanging towels hanging beside it and hanging towels hanging in a ceiling rack on some |
| | ... | ... | ... |
| | 0.2947 | Baseline (1.0) | a black and white photo of a black and white closet |
|  | 0.3810 | 4.0 | a white towel hanging from a rack in a closet |
| | 0.3710 | 0.5 | a white towel hanging in a closet with towels on the shelves |
| | 0.3596 | 1.5 | a white towel hanging in a bathroom next to a white shelf full of clothes and other items |
| | ... | ... | ... |
| | 0.2960 | Baseline (1.0) | a bathroom with a bunch of towels hanging on the wall |

Figure 5. Samples of captions generated at different temperatures and their cosine similarity to the corresponding images. We notice that a high temperature leads to informative and also noisy caption generation.

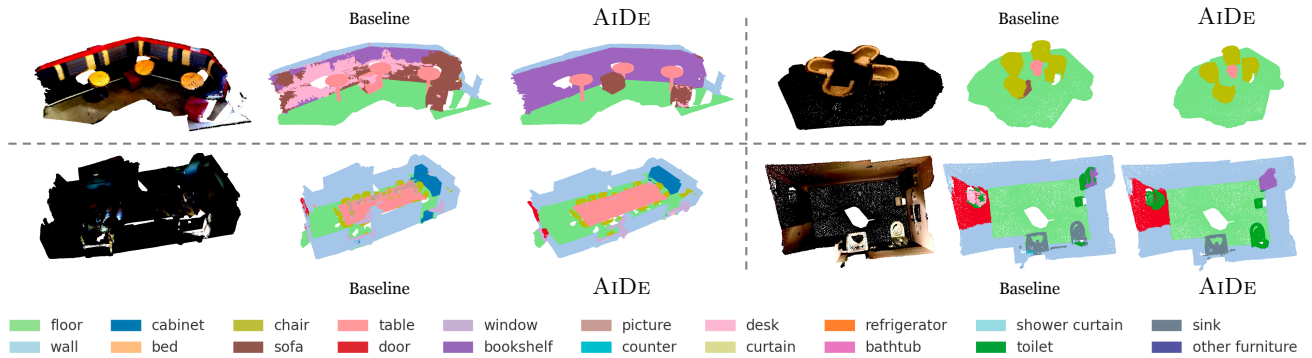


Figure 6. Qualitative results of segmentation compared between baseline and AiDE.

(1.0). This variability underscores the importance of temperature in controlling the trade-off between diversity and relevance of generated captions, a critical aspect in tasks requiring a nuanced understanding of visual content, such as semantic segmentation, object recognition, and image captioning in computer vision.

Synonyms and hypernyms. Here, we present the synonyms and hypernyms of different classes in Tab. 13 used for calculating the results in Tabs. 8 and 14. We use Word-

Net [25] and Cambridge Dictionary² to select appropriate synonyms and hypernyms to avoid confusion. For classes without suitable synonyms or hypernyms, we use the original class name as their synonyms and hypernyms and mark “-” in the table.

The meaning of learned prompts. To understand the meaning of learned prompts, we calculate the cosine similarity between the learned prompts in the input space and the human vocabulary (subwords in CLIP’s vocabulary)

²<https://dictionary.cambridge.org/>.

| Methods | hIoU | mIoU ^B | mIoU ^N | IoU on Base Categories | | | | | | | |
|-----------------------------|------------------------|-------------------|-------------------|------------------------|---------------|--------------|--------------|-------------------------|--------------|--------------|--------------|
| | | | | wall | floor | cabinet | bed | chair | table | door | window |
| <i>Original Class Names</i> | | | | | | | | | | | |
| Baseline | 0.645 | 0.679 | 0.615 | 0.839 | 0.950 | 0.638 | 0.808 | 0.880 | 0.705 | 0.595 | 0.617 |
| AiDE | 0.728 | 0.719 | 0.738 | 0.874 | 0.979 | 0.684 | 0.818 | 0.916 | 0.744 | 0.637 | 0.658 |
| <i>Synonyms</i> | | | | | | | | | | | |
| Baseline | 0.289 | 0.207 | 0.479 | 0.262 | 0.138 | 0.029 | 0.566 | 0.292 | 0.004 | 0.155 | 0.346 |
| AiDE | 0.340 | 0.243 | 0.568 | 0.166 | 0.352 | 0.012 | 0.739 | 0.550 | 0.007 | 0.269 | 0.326 |
| <i>Hypernyms</i> | | | | | | | | | | | |
| Baseline | 0.311 | 0.230 | 0.478 | 0.000 | 0.000 | 0.418 | 0.401 | 0.528 | 0.140 | 0.058 | 0.344 |
| AiDE | 0.364 | 0.264 | 0.588 | 0.001 | 0.000 | 0.477 | 0.637 | 0.326 | 0.183 | 0.190 | 0.378 |
| Methods | IoU on Base Categories | | | | | | | IoU on Novel Categories | | | |
| | picture | counter | curtain | refrigerator | showercurtain | sink | bathtub | sofa | bookshelf | desk | toilet |
| <i>Original Class Names</i> | | | | | | | | | | | |
| Baseline | 0.312 | 0.591 | 0.697 | 0.488 | 0.652 | 0.571 | 0.837 | 0.734 | 0.681 | 0.449 | 0.594 |
| AiDE | 0.349 | 0.646 | 0.733 | 0.561 | 0.676 | 0.656 | 0.854 | 0.821 | 0.756 | 0.513 | 0.862 |
| <i>Synonyms</i> | | | | | | | | | | | |
| Baseline | 0.005 | 0.261 | 0.000 | 0.079 | 0.592 | 0.000 | 0.381 | 0.686 | 0.524 | 0.242 | 0.464 |
| AiDE | 0.002 | 0.196 | 0.000 | 0.249 | 0.493 | 0.000 | 0.280 | 0.643 | 0.577 | 0.304 | 0.746 |
| <i>Hypernyms</i> | | | | | | | | | | | |
| Baseline | 0.040 | 0.461 | 0.506 | 0.055 | 0.339 | 0.000 | 0.165 | 0.724 | 0.621 | 0.425 | 0.143 |
| AiDE | 0.035 | 0.418 | 0.537 | 0.031 | 0.489 | 0.000 | 0.255 | 0.750 | 0.666 | 0.474 | 0.464 |

Table 14. Open-vocabulary semantic segmentation results on ScanNet (B15/N4) using the original class names, their synonyms, and hypernyms. The synonyms and hypernyms of class names are presented in the Appendix.

presented in Tab. 15. It shows that both models tend to align with basic English words such as articles (“a”, “an”, “the”) and pronouns (“his”, “my”). Notably, the learned prompts on both ScanNet and S3DIS exhibit high similarity with “scene”. On the other side, learned prompts of different benchmarks tend to have different composed meanings as the top 2 - top 5 ranked words are different. It reflects the AiDE’s capacity to approximate human-like linguistic structures and adaptation to domain-specific linguistic contexts.

Visualization. We further visualize more examples in Fig. 6. It is obvious that, compared with our baseline (PLA), AiDE has a better segmentation on both the seen and unseen classes. The desk on the bottom left, the toilet on the bottom right, the chair on the top right, and the bookshelf on the top left are more clear than the baseline model.

C.4. Aligning 3D with Images

In this paper, we focus on aligning the captions with 3D point cloud features, whereas another line of research [32, 57] aims at aligning 3D point cloud features with the corresponding images.

To have a fair comparison with these methods and to augment AiDE with image alignment capabilities, we implement view-level image alignment minimizing the PDC

loss,

$$\ell_{\text{align}}^{\text{view, image}} = \sum_{i \in [I^{\text{view}}]} \ell_{\text{pdc}} \left(f_{3\text{D}} \left(\hat{P}_i^{\text{view}} \right), f_{\text{text}} \left(I_i^{\text{view}} \right) \right), \quad (7)$$

where I^{view} represents the images corresponding to a scene. For each image I_i^{view} , we select the points visible in that image, mapped with poses and other parameters, as the paired point set \hat{P}_i^{view} . This method is noted as “AiDE + View-Level 3D-Image Alignment” as shown in Tab. 16.

The table illustrates a compelling comparison between AiDE and other image-based methods [32, 57]. Results indicate that our proposed AiDE, even without additional view-level 3D-image alignment, achieves better results, specifically in mIoU^N, demonstrating its robustness and effectiveness. With the introduction of view-level 3D-image alignment, AiDE outperforms UniM-OV3D (3D-Text) in all metrics, illustrating the effectiveness of precise image alignment strategies in enhancing segmentation accuracy. Furthermore, integrating view-level 3D-image alignment into our framework demonstrates not only the model’s adaptability but also its ability to effectively utilize visual cues from aligned images for improved semantic alignment.

C.5. Generalization to Instance Segmentation

To understand the possibility of extending AiDE to instance segmentation, following previous methods [23, 67],

| ScanNet | | | | | | | | | | |
|------------|--------|--------|--------|---------|--------|---------|--------|---------|----------|---------|
| Token1 | | | | | | | | | | |
| Similarity | 0.8044 | 0.5539 | 0.4234 | 0.3598 | 0.3563 | 0.3497 | 0.3366 | 0.3357 | 0.3108 | 0.3102 |
| Word | a | an | the | his | my | some | this | yours | ans | s |
| Token2 | | | | | | | | | | |
| Similarity | 0.7393 | 0.4705 | 0.2389 | 0.2282 | 0.2209 | 0.213 | 0.208 | 0.1974 | 0.1966 | 0.1962 |
| Word | scene | scenes | scen | scenery | moment | stage | sight | set | cameo | display |
| Token3 | | | | | | | | | | |
| Similarity | 0.7616 | 0.4286 | 0.4227 | 0.3786 | 0.3687 | 0.3614 | 0.3614 | 0.3415 | 0.3327 | 0.3264 |
| Word | of | to | for | by | from | in | with | on | at | , |
| Token4 | | | | | | | | | | |
| Similarity | 0.6825 | 0.4426 | 0.2749 | 0.2704 | 0.2644 | 0.2624 | 0.2574 | 0.2538 | 0.2417 | 0.2387 |
| Word | a | an | the | your | my | and | as | in | any | every |
| S3DIS | | | | | | | | | | |
| Token1 | | | | | | | | | | |
| Similarity | 0.4468 | 0.3206 | 0.2098 | 0.2067 | 0.206 | 0.2036 | 0.2012 | 0.1966 | 0.1899 | 0.1892 |
| Word | a | an | ah | his | need | sus | put | some | had | took |
| Token2 | | | | | | | | | | |
| Similarity | 0.3924 | 0.2307 | 0.2305 | 0.1852 | 0.1804 | 0.1771 | 0.1755 | 0.1755 | 0.1625 | 0.1621 |
| Word | scene | scenes | sleet | steady | dumped | heavens | haz | smashes | pavement | sites |
| Token3 | | | | | | | | | | |
| Similarity | 0.3181 | 0.2087 | 0.1972 | 0.1919 | 0.1911 | 0.1865 | 0.1826 | 0.1778 | 0.1764 | 0.1748 |
| Word | of | s | in | from | and | being | into | are | you | your |
| Token4 | | | | | | | | | | |
| Similarity | 0.3856 | 0.3002 | 0.2556 | 0.2413 | 0.2379 | 0.2343 | 0.23 | 0.2257 | 0.2217 | 0.2182 |
| Word | a | an | his | my | your | you | he | us | her | me |

Table 15. Similarity between learned prompts (tokens) in the input space and “human vocabulary” under use the cosine similarity. “Word” refers to subwords in the CLIP’s vocabulary.

| Method | ScanNet (B15/N4) | | |
|--------------------------------------|------------------|-------------------|-------------------|
| | hIoU | mIoU ^B | mIoU ^N |
| OpenScene [†] [57] | 67.1 | 68.8 | 62.8 |
| UniM-OV3D (3D-Text Only) [32] | 62.1 | 66.2 | 56.4 |
| AiDE | 72.8 | 71.9 | 73.8 |
| AiDE + View-Level 3D-Image Alignment | 73.0 | 72.7 | 73.3 |
| Fully-Supervised | 73.3 | 68.4 | 79.1 |

Table 16. Ablation studies on using the image alignment and comparisons with image-based methods. † refers to numbers copied from He *et al.* [32].

we introduce an instance localization head $f_{loc}(\cdot)$ for generating instance proposal and instance class prediction (details see S1.2 [23]). Specifically, instance proposals pps are first generated by grouping the segmentation prediction $f_{seg}(f_{3D}(P))$, offset head, and the point-wise features $f_{3D}(P)$. Next, with a TinyUNet and the class-agnostic seg and score heads, perclass confidence for instance proposals pps are generated with the pooled segmentation prediction

$$f_{seg}(f_{3D}(P)).$$

The results of LSeg-3D, our baseline (PLA), and AiDE on instance segmentation on S3DIS (B8/N4) are shown in Tab. 17. We utilize AP_{50}^B and AP_{50}^N and their harmonic mean as evaluation metrics. Notably, AiDE demonstrates superior performance across all metrics, achieving hAP_{50} , mAP_{50}^B , and mAP_{50}^N of 34.5, 61.9, and 23.9, respectively. This demonstrates AiDE’s effectiveness in handling

| Methods | S3DIS (B8/N4) | | |
|------------------|---------------|--------------|--------------|
| | hAP_{50} | mAP_{50}^B | mAP_{50}^N |
| LSeg-3D | 0.5 | 58.3 | 0.3 |
| Baseline | 26.7 | 60.3 | 17.2 |
| AiDE | 34.5 | 61.9 | 23.9 |
| Fully-Supervised | 57.6 | 60.8 | 54.6 |

Table 17. Generalization of AiDE to instance segmentation on S3DIS.

instance segmentation tasks, significantly improving upon the baseline (26.7 in hAP_{50} , 60.3 in mAP_{50}^B , and 17.2 in mAP_{50}^N) and significantly closing the gap with the fully supervised model, which leads with 57.6 in hAP_{50} and 54.6 in mAP_{50}^N . This underscores AiDE’s potential to enhance open-vocabulary instance segmentation capabilities, while also indicating the need for further advancements to narrow the significant gap with fully supervised methods.

D. Prompt Templates

In this section, we present the prompt templates used for producing the results in Tab. 1b.

Identity. This prompt template is shown as follows,

{CLASS}

Simple. This prompt template is shown as follows,

a photo of a {CLASS}

LSeg. This prompt template is shown as follows,

a {CLASS} in a scene

Full-ImageNet. This prompt template uses the following 81 templates to generate text embedding for each class and then do an average of 81 embeddings.

a bad photo of a {CLASS}
 a photo of many {CLASS}
 a sculpture of a {CLASS}
 a photo of the hard to see {CLASS}
 a low resolution photo of the {CLASS}
 a rendering of a {CLASS}
 graffiti of a {CLASS}
 a bad photo of the {CLASS}
 a cropped photo of the {CLASS}
 a tattoo of a {CLASS}
 the embroidered {CLASS}
 a photo of a hard to see {CLASS}
 a bright photo of a {CLASS}
 a photo of a clean {CLASS}
 a photo of a dirty {CLASS}
 a dark photo of the {CLASS}
 a drawing of a {CLASS}
 a photo of my {CLASS}
 the plastic {CLASS}
 a photo of the cool {CLASS}
 a close-up photo of a {CLASS}
 a black and white photo of the {CLASS}
 a painting of the {CLASS}
 a painting of a {CLASS}
 a pixelated photo of the {CLASS}
 a sculpture of the {CLASS}

a bright photo of the {CLASS}
 a cropped photo of a {CLASS}
 a plastic {CLASS}
 a photo of the dirty {CLASS}
 a jpeg corrupted photo of a {CLASS}
 a blurry photo of the {CLASS}
 a photo of the {CLASS}
 a good photo of the {CLASS}
 a rendering of the {CLASS}
 a {CLASS} in a video game
 a photo of one {CLASS}
 a doodle of a {CLASS}
 a close-up photo of the {CLASS}
 a photo of a {CLASS}
 the origami {CLASS}
 the {CLASS} in a video game
 a sketch of a {CLASS}
 a doodle of the {CLASS}
 a origami {CLASS}
 a low resolution photo of a {CLASS}
 the toy {CLASS}
 a rendition of the {CLASS}
 a photo of the clean {CLASS}
 a photo of a large {CLASS}
 a rendition of a {CLASS}
 a photo of a nice {CLASS}
 a photo of a weird {CLASS}
 a blurry photo of a {CLASS}
 a cartoon {CLASS}
 art of a {CLASS}
 a sketch of the {CLASS}
 a embroidered {CLASS}
 a pixelated photo of a {CLASS}
 itap of the {CLASS}
 a jpeg corrupted photo of the {CLASS}
 a good photo of a {CLASS}
 a plushie {CLASS}
 a photo of the nice {CLASS}
 a photo of the small {CLASS}
 a photo of the weird {CLASS}
 the cartoon {CLASS}
 art of the {CLASS}
 a drawing of the {CLASS}
 a photo of the large {CLASS}
 a black and white photo of a {CLASS}
 the plushie {CLASS}
 a dark photo of a {CLASS}
 itap of a {CLASS}
 graffiti of the {CLASS}
 a toy {CLASS}
 itap of my {CLASS}
 a photo of a cool {CLASS}
 a photo of a small {CLASS}
 a tattoo of the {CLASS}