3D Weakly Supervised Semantic Segmentation with 2D Vision-Language Guidance

Xiaoxu Xu[1], Yitian Yuan[2], Jinlong Li[3], Qiudan Zhang[1], Zequn Jie[2], Lin Ma[2], Hao Tang[4,5], Nicu Sebe[3], and Xu Wang[1⋆]

[1] College of Computer Science and Software Engineering, Shenzhen University, Shenzhen, 518060, China.
[2] Meituan Inc., China.
[3] University of Trento, Italy.
[4] Peking University, China.
[5] Carnegie Mellon University, USA.

Abstract. In this paper, we propose **3DSS-VLG**, a weakly supervised approach for **3D S**emantic **S**egmentation with 2D **V**ision-**L**anguage **G**uidance, an alternative approach that a 3D model predicts dense-embedding for each point which is co-embedded with both the aligned image and text spaces from the 2D vision-language model. Specifically, our method exploits the superior generalization ability of the 2D vision-language models and proposes the Embeddings Soft-Guidance Stage to utilize it to implicitly align 3D embeddings and text embeddings. Moreover, we introduce the Embeddings Specialization Stage to purify the feature representation with the help of a given scene-level label, specifying a better feature supervised by the corresponding text embedding. Thus, the 3D model is able to gain informative supervisions both from the image embedding and text embedding, leading to competitive segmentation performances. To the best of our knowledge, this is the first work to investigate 3D weakly supervised semantic segmentation by using the textual semantic information of text category labels. Moreover, with extensive quantitative and qualitative experiments, we present that our 3DSS-VLG is able not only to achieve the state-of-the-art performance on both S3DIS and ScanNet datasets, but also to maintain strong generalization capability. The code will be available at https://github.com/xuxiaoxxxx/3DSS-VLG/.

Keywords: 3D Weakly Supervised Semantic Segmentation · Vision-Language Model

1 Introduction

3D point cloud semantic segmentation [13, 16, 27–29, 43] can provide valuable geometric and semantic data about the 3D environment and has gained considerable attention over the past few years. Learning-based semantic segmentation methods have achieved remarkable performance recently, but they need per-point annotations, which is time consuming and labor intensive.

⋆ Corresponding author: wangxu@szu.edu.cn

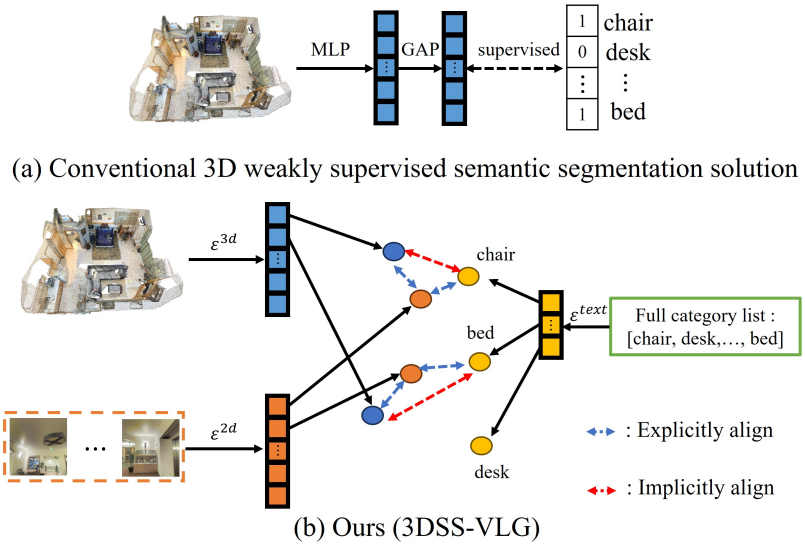

Fig. 1: Comparison of different approaches. (a) The conventional 3D WSSS approach adopts the coarse-grained CAM method in a global manner and is supervised by scene-level annotations or subcloud-level annotations. (b) Our proposed 3DSS-VLG approach leverages natural 3D-2D correspondence from geometric camera calibration and 2D-text correspondence from vision-language models, to implicitly align texts and 3D point clouds.

To address this issue, existing weakly supervised methods derive the segmentation model with different weak supervisory signals, such as subcloud-level annotations [36], scene-level annotations [18, 31, 45] and so on. As shown in Fig. 1 (a), the 3D Weakly Supervised Semantic Segmentation (3D WSSS) approaches typically adopt a Class Activation Map (CAM) [50] solution. Point clouds are first processed by several Multi-Layer Perception (MLP) layers and thus get a point cloud feature map, and then this point cloud feature map is processed by a Global Average Pooling (GAP) to get a global classification prediction, which is trained with subcloud-level or scene-level annotations. Given the simple GAP connectivity structure, these methods can easily identify the importance of each point by projecting back the output classification weight onto the point cloud feature maps, a technique we call CAM. In this way, the semantic segmentation for each category is back-derived from the global prediction. Recently, with the remarkable success of 2D vision, some methods [18, 44] also use the 2D module to enhance the 3D WSSS.

Although leveraging 2D-3D fusion in 3D WSSS seems to be promising, there also exist some problems. Kweon *et al.* [18] need extra detailed annotations of 2D images. As for MIT [44], although it avoids additional per-point/pixel annotations or per-image class labels, its performance is not expected. Therefore, how to design a network that achieves good performance despite the lack of 2D anno-

tations still remains a big challenge. Meanwhile, we notice that the conventional methods for 3D WSSS only use the scene-labels or subcloud-labels to supervise the model, but ignore that the textual category labels such as "chair, table" also have semantic meanings and could be embedded to help the model learning. At the same time, we also find that some methods [12, 25, 46] like Openseg [12], which leverage the pretrained vision-language models such as CLIP [30] to establish precise semantic matching relationships between natural languages and 2D images, have achieved good results in 2D open vocabulary semantic segmentation (2D OVSS). The above two points inspire us to consider whether we can use the well-pretrained 2D OVSS model to help the 3D WSSS. As shown in Fig. 1 (b), the point cloud and 2D images could be mutually mapped with geometric projections, and the 2D images and textual categories could be compared with pretrained vision-language models. Therefore, why do not we take the 2D images as a bridge, leveraging the correspondences between point clouds and images, images and natural languages, to implicitly build matching relationships between point clouds and natural languages?

To this end, we propose a simple yet effective method, namely **3DSS-VLG**, *i.e.,* a weakly supervised approach for **3D S**emantic **S**egmentation with 2D **V**ision-**L**anguage **G**uidance. Our 3DSS-VLG only needs to use 2D images, but no need for their 2D image-level annotations during training. Specifically, for the input 3D point cloud, the dataset also provides a set of multi-view images corresponding to it. We first process these multi-view images using the image encoder of the pretrained off-the-shelf 2D OVSS model such as Openseg [12] to get the 2D embeddings. Then, for each point in the 3D point cloud, we project it to the multi-view images with geometric projections, and integrate these corresponding 2D embeddings to get the 2D-projected embeddings for the point. Next, we utilize the text module of the 2D OVSS model to obtain the textual embeddings of each semantic category label. Since in the embedding space of 2D OVSS, the textual category labels and 2D images could be directly compared, we only need to learn a 3D backbone which could generate 3D embeddings aligned with 2D embeddings; thus, the category labels and the 3D point cloud could be implicitly compared.

Actually, if the 3D embedding is learned well enough, it can be directly compared with the text embedding by the similarity measurement to classify. However, we find that only relying on pulling the 2D-projected embeddings and 3D embeddings closely is not reliable since the pretrained 2D OVSS model are designed to learn the general knowledge and do not have specialized knowledge to the indoor point cloud scene. Therefore, we propose to alleviate this problem by three stages. (1) First, as shown in Fig. 2, we perform matrix multiplication on projected 2D embeddings and text embeddings of category labels and get the classification logits. Then, we use the scene-level labels as mask to filter out some confusing and unreliable predictions in the classification logits and thus get a more reliable pseudo label vector. (2) Second, as shown in Fig. 3 (a), we propose the Embeddings Specialization Stage, which transfers the 2D-projected embeddings with an adapter module to obtain adapted 3D embeddings, and the

training of this adapter module will be supervised with the pseudo label vector. This stage is designed to induce a more reliable target 3D embeddings suited for the indoor point cloud scene from the 2D-projected embeddings. (3) Finally, as shown in Fig. 3 (b), we design Embeddings Soft-Guidance Stage, which freezes the adapter module introduced in the second stage and leverages cosine similarity to align the adapted 3D embeddings and the MinkowskiNet [9] 3D embeddings. Combining the above three stages, we can learn a more reliable 3D embedding space for semantic segmentation in indoor point cloud scene. In the inference procedure, we only need to compare the MinkowskiNet 3D embeddings of the point cloud and the text embeddings of the semantic category labels, thus accomplishing the 3D semantic segmentation. Note that we do not need 2D images to participate in the inference process of our model.

In summary, the main contributions of this paper are as follows:

- We propose a weakly supervised method 3DSS-VLG for 3D WSSS, which takes 2D images as a bridge, and leverages natural 3D-2D correspondence from geometric camera calibration and 2D-text correspondence from vision-language models to implicitly establish the semantic relationships between texts and 3D point clouds.
- We design a three-stage training procedure to learn a reliable 3D embedding space in 3DSS-VLG for 3D semantic segmentation. Embeddings Specialization Stage is designed to utilize the pretrained 2D vision-language model to provide a embedding space for 3D point cloud representation with MinkowskiNet 3D backbone. Moreover, we propose Embeddings Specialization Stage to make the embedding space to be more robust based on the pseudo label filtering with indoor point cloud scene knowledge.
- Extensive experiments on the ScanNet and S3DIS dataset show that the proposed 3DSS-VLG significantly outperforms the previous state-of-the-art methods, even Kweon *et al.* which use extra 2D image-level annotations. Moreover, our futher experiments show our 3DSS-VLG has strong generalization capability and can be extended to handle unobserved general datasets.

2 Related Work

2.1 2D Open-Vocabulary Semantic Segmentation

The recent advances of large vision-language models have enabled a remarkable level of robustness in open-vocabulary semantic segmentation [6,25,39–41]. Open vocabulary semantic segmentation aims to segment the target categories that cannot be access during the training procedure. Pioneering work ZS3Net [4] uses generative models to synthesize pixel-level features by word embeddings of unseen classes. SPNet [37] encodes visual features into the semantic embeddings space to align with text embeddings. More recently, researchers propose to leverage the pretrained CLIP [30] for open-vocabulary semantic segmentation. ZSSeg [41] leverages the visual module to generate class-agnostic masks and uses the pretrained text encoder to retrieve the unseen class masks. OpenSeg [12]

proposes to align the segment-level visual embeddings with text embeddings via region-word grounding. In this work, we solely rely on pretrained 2D open-vocabulary models and perform 3D weakly supervised semantic segmentation understarnding tasks. We pull the 3D embeddings and 2D embeddings which features exacted from pretrained model back-project onto point cloud closed to implicitly align 3D embeddings and text embeddings.

2.2 3D Weakly Supervised Semantic Segmentation

This task aims to learn point cloud semantic segmentation using weakly annotated data, such as sparsely labeled points [15,48], box-level labels [8], subcloud-level labels [36] and scene-level labels [9,18,31,44]. Though the state-of-the-art methods based on sparsely labeled points show performance comparable to that of supervised ones, they require at least partial point-wise annotation in a scene, which is still expensive compared to subcloud-level labels and scene-level labels. The pipeline of the conventional CAM solution has been used in the majority of previous 3D WSSS works and only treats the scene-level labels as one-hot digit. MPRM [36] proposes the subcloud-level annotations method that samples subclouds from the full scene and annotates them, which can alleviate the class imbalance issue commonly appearing in almost scene. However, the subcloud-level annotations need to divide the point cloud into small that we need to annotations more than one for a scene, which is too much trouble and time-consuming. Therefore, some methods that use scene-level annotations are proposals for the 3D WSSS. Kweon *et al.* [18] utilizes 2D and 3D data for semantic segmentation and gets good performance, however, requiring extra 2D image-level labels. MIT [44] proposes the interlaced transformer structure to fuse 2D-3D information with only scene-level labels. However, its performance is not as good as expected.

Therefore, in this work, we explore a 3D WSSS method with only scene-level labels. Unlike those previous works, we use the semantic meanings of textual category labels to assist in model learning. Moreover, the performance of our 3DSS-VLG is over the Kweon *et al.*, which uses extra 2D image-level labels.

2.3 2D Semantic in 3D task

Studies on 3D object detection and semantic segmentation [1,2,5,11,14,20–24] have explored the use of 2D image semantics to assist 3D tasks. There are almost two approaches: concatenating the image embeddings with each point in the 3D scene as extra information [7,17,42,47] or projecting image semantic results into a 3D space to assist 3D semantic segmentation [19,26,38]. Previous studies usually used 2D image semantics as extra inputs in both training and inference. Although performance has improved, the extra 2D inputs have the potential to constrain the range of application scenarios. This is due to the fact that 2D information may be absent during inference or necessitate laborious pre-processing. In this paper, we aim to investigate the potential of using 2D semantics exclusively during training to assist in the 3D WSSS task.

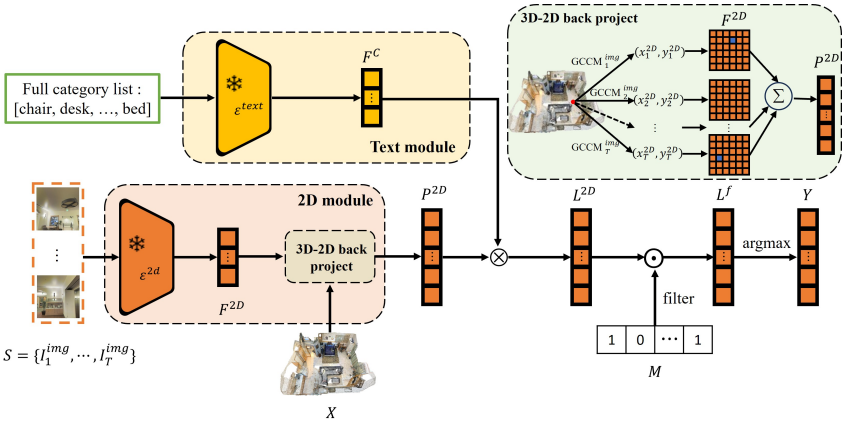

Fig. 2: The proposed pseudo label generation procedure. We first leverage the text encoder ε^{text} of Openseg to get embeddings of the full category labels $\mathbf{F}^C$, and leverage the 2D image encoder ε^{2D} of Openseg to get embeddings of the 2D image $\mathbf{F}^{2D}$. It is important to note that we freeze the whole Openseg model during the procedure of pseudo label generation. Then we back-project the 2D embeddings $\mathbf{F}^{2D}$ to integrate the 2D-projected embeddings $\mathbf{P}^{2D}$. Specifically, for each point in the point cloud (x^{3D}, y^{3D}, z^{3D}), we use geometric camera calibration matrixes $GCCM^{img}$ to calculate the corresponding positions (x^{2D}, y^{2D}) on the multi-view images S. Then we integrate these corresponding 2D embeddings in $\mathbf{F}^{2D}$ and average them to get the 2D-projected embeddings $\mathbf{P}^{2D}$. We perform matrix multiplication on $\mathbf{F}^C$ and $\mathbf{P}^{2D}$, and get the 3D point cloud semantic segmentation prediction logits $\mathbf{L}^{2D}$. Finally we utilize the scene-level labels as mask M to filter out some confusing and unreliable predictions in the classification and get the more accurate predicted logits $\mathbf{L}^f$ and pseudo labels $\mathbf{Y}$.

3 The Proposed Method

In this section, we will first introduce the procedure of pseudo label generation in Sec. 3.1. Then, we will demonstrate the training procedure of our 3DSS-VLG in Sec. 3.2 and Sec. 3.3. Finally, we will describe the 3DSS-VLG inference procedure in Sec. 3.4.

3.1 Pseudo Label Generation Stage

This stage aims to utilize the pretrained vision-language model and scene-level labels to generate more precise pseudo label. Given an input point cloud with multi-view images as shown in Fig. 2, we first implement dense 2D embeddings extraction for each RGB image via the frozen visual encoder of Openseg [12], and back-project them onto the 3D surface points of a scene to integrate the 2D-projected embeddings. Afterward, more accurate pseudo labels are generated based on 2D-projected embeddings, text embeddings and scene-level labels.

2D Embeddings Extraction. The inputs of 3DSS-VLG comprise a scene with 3D point cloud, scene-level labels and the associated multi-view RGB images

set. Given the RGB images set S consists of T images with a resolution of $H \times W$. The point cloud $\mathbf{X} \in \mathbb{R}^{N\times 6}$ contains N points in the scene, and each point is represented with six dimensions of RGBXYZ. We leverage the pre-trained image encoder of OpenSeg [12] to get per-pixel embedding, denoted as $\mathbf{F}^{2D} \in \mathbb{R}^{T\times H\times W\times d}$, where d is the 2D embedding dimension. For each point in the 3D point cloud, we project it onto multi-view images through geometric camera calibration matrixes and get the corresponding 2D positions. Then we can exact the corresponding projected 2D embeddings from $\mathbf{F}^{2D}$ according to the calculated 2D image positions. Since each point may have multiple correspondences in different images, the final 2D-projected embeddings $\mathbf{P}^{2D} \in \mathbb{R}^{N\times d}$ is obtained via average all the corresponding projected 2D embeddings of each point.

Text Embeddings Extraction. We take the text encoder of Openseg to exact text embeddings $\mathbf{F}^{C} \in \mathbb{R}^{K\times d}$ of full category labels, where K denoted the number of categories. Similarly, we also freeze the text encoder and directly load the pretrained Openseg parameters.

Filtering Strategy. After getting the 2D-projected embeddings $\mathbf{P}^{2D}$ and the text embeddings $\mathbf{F}^{C}$, we perform matrix multiplication on them and obtain the classification logits $\mathbf{L}^{2D} \in \mathbb{R}^{N\times K}$. To make classification logits more reliable, the filtering strategy is employed to filter out confusing and unreliable predictions. For instance, as shown in Fig. 2, we create a boolean scene-level label mask $\mathbf{M} \in \mathbb{R}^{1\times K}$, where the element value in the mask indicated whether the corresponding category existed. Finally, we perform matrix inner product on classification logits $\mathbf{L}^{2D}$ and scene-level label mask $\mathbf{M}$ and obtain filtered classification logits $\mathbf{L}^{f} \in \mathbb{R}^{N\times K}$. After ranking the filtered classification logits $\mathbf{L}^{f}$, we can get the more precise pseudo label $\mathbf{Y} \in \mathbb{R}^{N}$ of the input point cloud.

3.2 Embeddings Specialization Stage

As we know, the 2D OVSS model is designed to learn general knowledge and do not have any specialized knowledge of the indoor point cloud scene. Therefore, only relying on the 2D embeddings to build the 3D-text correlation will make the 3D WSSS process not reliable. To mitigate this issue, the Embeddings Specialization Stage is proposed to further improve the perception of indoor knowledge of 3D embeddings.

Specifically, the 2D-projected embeddings $\mathbf{P}^{2D}$ of input are transferred into another space through the adapter module, which simply contains two fully-connected layers. Besides, to keep both the source and adapted semantics, we employ the residual connections to get the adapted 3D embeddings $\mathbf{A}^{3D} \in \mathbb{R}^{N\times d}$:

$$\mathbf{A}^{3D} = \alpha \cdot MLP(\mathbf{P}^{2D}) + (1-\alpha)\cdot \mathbf{P}^{2D}, \tag{1}$$

where α is the ratio of residual connections. Next, we perform matrix multiplication on text embeddings $\mathbf{F}^{C}$ and adapted 3D embeddings $\mathbf{A}^{3D}$ and obtain the classification logits $\mathbf{L}^{a} \in \mathbb{R}^{N\times K}$. The softmax layer is applied on $\mathbf{L}^{a}$ and

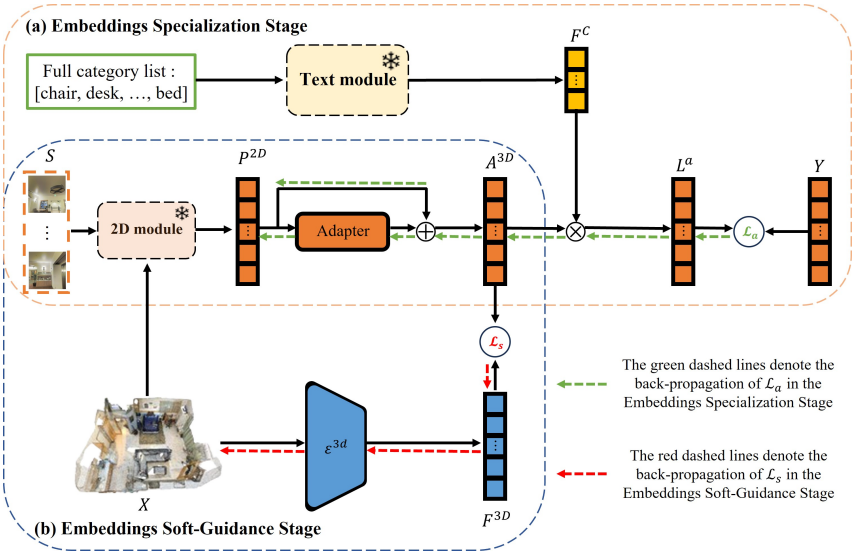

Fig. 3: The proposed training procedure of our proposed 3DSS-VLG. Here, it is mainly divided into two stages: (a) Embeddings Specialization Stage and (b) Embeddings Soft-Guidance Stage. For (a), we first utilize the text encoder ε^{text} of Openseg to obtain embeddings of the category labels $\mathbf{F}^C$, which are frozen during the training procedure of (a). Meanwhile, we get the initial 2D-projected embeddings $\mathbf{P}^{2D}$ from the 2D module and leverage the adapter module to transfer the $\mathbf{P}^{2D}$ to a new embedding spaces to obtain the adapted 3D embeddings $\mathbf{A}^{3D}$. We perform matrix multiplication on $\mathbf{A}^{3D}$ and $\mathbf{F}^C$ and get the predicted probability $\mathbf{L}^a$. Finally, we use the pseudo labels $\mathbf{Y}$ to supervise the model, and the green dashed lines denote back-propagation of the loss $\mathcal{L}_a$. For (b), we first utilize the adapter module and obtain the adapted 3D embeddings $\mathbf{A}^{3D}$. It is important to note that we freeze the adapter module during the training procedure of (b). Meanwhile, we use the 3D module ε^{3D} to obtain the 3D embeddings $\mathbf{F}^{3D}$. The cosine similarity loss $\mathcal{L}_s$ will be integrated to train the model. The red dashed lines denote back-propagation of the loss $\mathcal{L}_s$.

a classification cross-entropy loss $\mathcal{L}_a$ is introduced to supervise the procedure. Here we leverage the pseudo labels $\mathbf{Y}$ of the point cloud to supervise the model.

Just by introducing the above simple adapter module, we can make the learned adapted embeddings have better semantic awareness of the point clouds of indoor scenes, thus assisting the 3D WSSS task.

3.3 Embeddings Soft-Guidance Stage

Since Openseg has established a high level of semantic alignment between 2D embeddings and text embeddings, we propose the Embeddings Soft-Guidance Stage, which can naturally take the 2D embeddings as a bridge to implicitly align the 3D embeddings and text embeddings via cosine similarity. Specifically, as shown in Fig. 3 (b), we take the point cloud $\mathbf{X}$ as input, and use Minkowsk-

iNet18A UNet [9] as our 3D module meanwhile, we change the dimension of the outputs to d. Therefore, we can get the learned 3D embeddings $\mathbf{F}^{3D} \in \mathbb{R}^{N \times d}$. Then we take the corresponding 2D-projected embeddings $\mathbf{P}^{2D}$ as input, processed by the adapter module, and get the adapted 3D embeddings $\mathbf{A}^{3D}$. We follow the typical cosine similarity loss by pulling the paired 3D embeddings $\mathbf{F}^{3D}$ and adapted 3D embeddings $\mathbf{A}^{3D}$ closer. We need to note that we freeze the adapter module and directly load the parameters provided by Sec. 3.2 during training. Therefore, we define the 3DSS-VLG loss as:

$$\mathcal{L}_s = 1 - \cos\left(\mathbf{F}^{3D}, \mathbf{A}^{3D}\right). \tag{2}$$

3.4 Inference

During inference, we only retain the 3D and text modules and remove the 2D module. Specifically, we take the 3D embeddings $\mathbf{F}^{3D}$ from the 3D module, as well as the category embeddings $\mathbf{F}^{C}$ from text module, to perform matrix multiplication on them and get the classification logits. Finally, we rank the logits and obtain the final per-point segmentation for the input point cloud $\mathbf{X}$.

4 Experiments

In this section, we first present our experimental settings, including datasets, evaluation metrics, and implementation details. The competing methods are then presented and compared. Finally, ablation studies are provided to further demonstrate the necessity and effectiveness of each component of our framework.

4.1 Datasets and Evaluation Metrics

We evaluate our 3DSS-VLG on two publicly and widely-used large-scale point cloud datasets with multi-view images, S3DIS [3] and ScanNet [10]. S3DIS is proposed for indoor scene understanding. It consists of 6 areas including 271 rooms with 13 classes. Each room is scanned via RGBD sensors and is represented by a point cloud with 3D coordinates and RGB values. Following previous works, we take area 5 as the test scene. ScanNet [10] has 1513 training scenes and 100 test scenes with 20 classes. We adopt the default train-val split setting, where there are 1201 training scenes and 312 validation scenes. The mean intersection over Union (mIoU) is employed as the evaluation metric for datasets.

4.2 Implementations Details

3DSS-VLG is implemented by PyTorch. For the training procedure of Sec. 3.2, we use Adam optimizer with batch size of 16 and set an initial learning rate of 0.003 for the model. We reduce the learning rate by a multiplying factor of 0.7 every 20 epochs for a total of 80 epochs. For the training procedure of Sec. 3.3, the model optimization is conducted using Adam optimizer with a batch size of 8. We set an initial learning rate of 0.0001 for the model and use the poly learning rate policy to adjust the learning rate.

Table 1: Performance comparison on the S3DIS dataset. "Sup." indicates the type of supervision. "100%" represents full annotation. "scene." denotes scene-level annotation.

Method	Label Effort	Sup.	Test
PointNet [27]	>20 min	100%	41.1
TangentConv [33]		100%	52.8
MinkowskiNet [9]		100%	65.8
KPConv [34]		100%	67.1
PointTransformer [49]		100%	70.4
PointNetXt [29]		100%	70.5
DeepViewAgg [32]		100%	67.2
SemAffiNet [35]		100%	71.6
MPRM [36]	<1 min	scene.	10.3
MIL-Trans [45]		scene.	12.9
WYPR [31]		scene.	22.3
MIT [44]		scene.	27.7
Ours		scene.	**45.3**

4.3 3D Semantic Segmentation Results

We evaluate our proposed approach against state-of-the-art techniques for 3D weakly supervised semantic segmentation with scene-level labels. Firstly, we demonstrate some full supervised point cloud semantic segmentation methods to compare the gap between the performances of ours and full supervised methods. Subsequently, we introduce semantic segmentation methods supervised by scene-level labels or subcloud-level labels and compare them with our method. Meanwhile, we indicate the average annotation time per scene.

Evaluation on S3DIS. Tab. 1 Show the performance of each type of 3D point cloud semantic segmentation methods evaluated on the S3DIS dataset. We can find that in the scene-level annotations setting, our method greatly surpasses the existing state-of-the-art method MIT [44] by 17.6%. This shows that using textual semantic information ignored by previous 3D weakly supervised semantic segmentation can significantly improve segmentation performance. The textual semantic information of each category is unique; then the 2D embeddings and 3D embeddings are aligned so that the 3D embeddings can be implicitly aligned to the corresponding unique category semantic information, which allows the model to achieve greater performance improvements.

Meanwhile, we compare our method with some full supervised methods. It can be observed that our 3DSS-VLG can outperform some fully supervised methods, *i.e.*, PointNet [27]. Moreover, we notice that the annotations cost time of different types of supervision and find that the scene-level annotation is the most efficient compared to other types annotations. Such results demonstrate the effectiveness and potential of our weakly supervised method.

Evaluation on ScanNet. We also evaluate our 3DSS-VLG on the ScanNet online test set and the validation set and presented the performance results of 3DSS-VLG in Tab. 2. For the test set, it can be observed that our 3DSS-VLG

Table 2: Performance comparison on the ScanNet test set and validation set. "Sup." indicates the type of supervision. "100%" represents full annotation. "subcloud." and "scene." imply subcloud-level annotation and scene-level annotation respectively. "image." denotes image-level annotation.

Method	Label Effort	Sup.	Test	Val
PointNet++ [28]	>20 min	100%	33.9	-
TangentConv [33]		100%	43.8	-
MinkowskiNet [9]		100%	73.6	72.2
KPConv [34]		100%	68.6	69.2
PointTransformer [49]		100%	-	70.6
PointNetXt [29]		100%	71.2	71.5
DeepViewAgg [32]		100%	-	71.0
SemAffiNet [35]		100%	74.9	-
MPRM [36]	3 min	subcloud.	41.1	43.2
Kweon *et al.* [18]	5 min	scene. + image.	47.4	49.6
MIL-Trans [45]	<1 min	scene.	-	26.2
WYPR [31]		scene.	24.0	29.6
MIT [44]		scene.	31.7	35.8
Ours		scene.	**48.9**	**49.7**

achieves the best performance under only scene-level label supervision and even surpasses the performance of MPRM [36] which is supervised by subcloud-level annotations. Moreover, we are surprised to find that our method also outperforms Kweon *et al.* [18] by 1.5%, which uses not only scene-level labels, but also extra image-level labels. Our method can achieve stronger performance with less annotations, further illustrating the superiority of our method. Meanwhile, our 3DSS-VLG can outperform some fully supervised methods. In addition for the validation set, our method also achieves the state-of-the-art during those 3D WSSS approaches. Those results demonstrate the superiority of 3DSS-VLG.

4.4 Ablation Studies

Effectiveness of Each Components. To demonstrate the advantage of each component in our 3DSS-VLG, we conduct comprehensive ablation studies on the S3DIS dataset, as shown in Tab. 3. The ablation model (a) only retains the MinkowskiNet18A UNet [9] and trains directly with the pseudo labels which are generated without using scene-level labels filtering. The cross-entropy loss is introduced to supervised this procedure. We set model (a) as the baseline of our experiment. Compared to model (a), model (b) is not directly supervised by pseudo labels. It adopts the Embeddings Soft-Guidance Stage (ESGS) and is soft-guided by the 2D-projected embeddings $\mathbf{P}^{2D}$. We can find that the performance of mIoU is improved from 37.7% to 38.2%. This observation proves that the soft-guidance strategy can guide 3D embeddings to align with the text embeddings and achieve better performance compared to directly using the pseudo

Table 3: Ablation studies of the 3DSS-VLG components on S3DIS dataset.

	ESGS	Filtering	ESS	mIoU
(a)				37.7
(b)	✓			38.2
(c)		✓		42.6
(d)	✓	✓	✓	45.3

Table 4: Performance comparisons of the generalization capability.

Domain	mIoU	mAcc
S3DIS ->ScanNet	13.4	23.0
ScanNet ->S3DIS	33.3	50.9

labels to supervised 3D model. Meanwhile, when we introduce the filtering strategy to model (a), as shown in model (c), we can find that the model performance increases greatly from 37.7% to 42.6%. Finally, by adding the filtering strategy to model (b) and utilizing the Embeddings Specialization Stage (ESS), model (d) is supervised by adapted 3D embeddings $\mathbf{A}^{3D}$ at this time. It can be observed the performance improves from 38.2% to 45.3%. Such results prove that our 3DSS-VLG can help the model to get a better, indoor point cloud specific embedding space to align 3D point clouds and text.

Generalization Capability. Due to the domain gap among different datasets, a model trained on one dataset is not applicable to another dataset. Also, this situation occurs in the 3D WSSS task. Nevertheless, we notice that, compared to previous works, our 3DSS-VLG uses textual semantic information as a guide rather than CAM, which means our model has a good relationship between 3D point cloud and the text of category labels and indicates that the model may have generalization ability. Therefore, we further explore our framework to the novel data of the unobserved scene domains.

As shown in Tab. 4, we experimentally verify the generalizability of the proposed method on the S3DIS and ScanNet dataset, respectively. The first row is the performance of model that we first train our model on the S3DIS dataset and then test the trained model on validation set of the ScanNet dataset. The second row is the performance of model that we first train our model on the ScanNet dataset and then test the trained model on the test set of the S3DIS dataset. Compared to those weakly supervised methods with scene-level labels, it can be observed that our 3DSS-VLG has a certain gap with those methods in the first row. However, for the second row, we are supervised to find that our method can outperform all the weakly supervised methods and achieve state-of-the-art performance. The ScanNet dataset provides six times more training scenes than the S3DIS dataset. Therefore, when a model is pretrained on the ScanNet dataset, the model will be more robust than a model pretrained on the S3DIS dataset. Our experimental results also prove this phenomenon.

Table 5: Performance comparisons with different 3D backbones and ESS module backbones on the S3DIS dataset.

Module	Backbone	mIoU
3D	MinkowskiNet14A	44.5
	MinkowskiNet18A	45.3
	MinkowskiNet34A	44.7
ESS	Transformers	45.0
	MLP	45.3

The results also strongly support the complementary advantages of using text semantic information, even without any further fine-tuning or domain-specific adaptation. Our 3DSS-VLG can be extended to handle unobserved general data and has strong generalization capability, which is promising for the field of 3D WSSS.

Experiments with Different Backbones. Tab. 5 shows the performances of our method on S3DIS with different 3D backbones and ESS module backbones. Finally, we use the MinkowskiNet18A as our 3D backbone and the FC-layer as the backbone of our ESS.

4.5 Qualitative Results

Fig. 4 visualizes the qualitative comparison of the proposed framework and baseline. Here the baseline is model (a) which is mentioned in Sec. 4.4. Compared with the result of baseline, our 3DSS-VLG shows significantly better results in the terms of accuracy of semantics and preciseness of segmentation. With the ESGS, ESS and filtering strategies, our 3DSS-VLG can learn a more better indoor point cloud specific embedding space to align 3D point clouds and text and achieve substantial semantic segmentation results compared to the baseline.

4.6 Limitations

Our work relies on vision-language alignment and does not address how to align visual embeddings with some abstract category text embeddings (*e.g.* "other" class in the S3DIS dataset). It is difficult for the model to understand what the difference is between the "other" class and other categories, thus making the wrong segmentation. This limitation is a direct avenue for future work.

5 Conclusion

In this paper, we propose 3DSS-VLG to address the shortage of point-level annotations. Specifically, our 3DSS-VLG exploits the superior ability of current vision-language models on aligning the semantics between texts and 2D images, as well as the naturally existing correspondences between 2D images and 3D

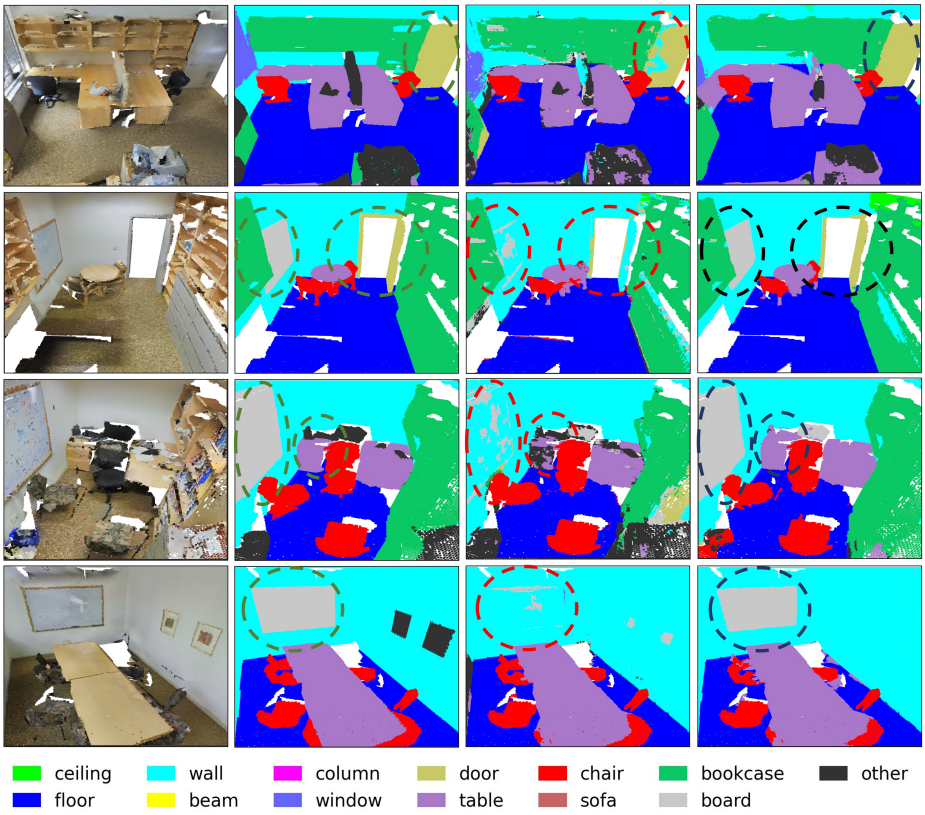

Fig. 4: Qualitative results on the S3DIS dataset of baseline and our 3DSS-VLG. From left to right: input point clouds, ground truth, baseline results, and our 3DSS-VLG results.

point clouds to implicitly co-embed texts embeddings with 3D point clouds embeddings using only scene-level labels. With extensive experiments, we verify that the textual semantic information of category labels is beneficial for 3DSS-VLG which achieves the state-of-the-art on both S3DIS and ScanNet datasets. Further, with an experiment to explore our framework to unobserved scene domains, we demonstrate the generalization capability of our method, which supports its practicality.

Acknowledgements

This work was supported in part by the National Natural Science Foundation of China under Grants 62371310 and 62032015, in part by the Guangdong Basic and Applied Basic Research Foundation under Grant 2023A1515011236, in part by the Stable Support Project of Shenzhen (Project No.20231122122722001), in

part by the third phase of high-level university construction of interdisciplinary innovation team project of Shenzhen University(24JCXK03). We also acknowledge the CINECA award under the ISCRA initiative, for the availability of partial HPC resources support, and partially supported by the Fundamental Research Funds for the Central Universities, Peking University.

References

1. Alonso, I., Riazuelo, L., Montesano, L., Murillo, A.C.: 3d-mininet: Learning a 2d representation from point clouds for fast and efficient 3d lidar semantic segmentation. IEEE Robotics and Automation Letters **5**(4), 5432–5439 (2020) 5
2. Ando, A., Gidaris, S., Bursuc, A., Puy, G., Boulch, A., Marlet, R.: Rangevit: Towards vision transformers for 3d semantic segmentation in autonomous driving. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 5240–5250 (2023) 5
3. Armeni, I., Sener, O., Zamir, A.R., Jiang, H., Brilakis, I., Fischer, M., Savarese, S.: 3d semantic parsing of large-scale indoor spaces. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 1534–1543 (2016) 9
4. Bucher, M., Vu, T.H., Cord, M., Pérez, P.: Zero-shot semantic segmentation. Advances in Neural Information Processing Systems **32** (2019) 4
5. Cardace, A., Ramirez, P.Z., Salti, S., Di Stefano, L.: Exploiting the complementarity of 2d and 3d networks to address domain-shift in 3d semantic segmentation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 98–109 (2023) 5
6. Chen, J., Zhu, D., Qian, G., Ghanem, B., Yan, Z., Zhu, C., Xiao, F., Culatana, S.C., Elhoseiny, M.: Exploring open-vocabulary semantic segmentation from clip vision encoder distillation only. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 699–710 (2023) 4
7. Chen, R., Liu, Y., Kong, L., Chen, N., Zhu, X., Ma, Y., Liu, T., Wang, W.: Towards label-free scene understanding by vision foundation models. Advances in Neural Information Processing Systems **36** (2024) 5
8. Chibane, J., Engelmann, F., Anh Tran, T., Pons-Moll, G.: Box2mask: Weakly supervised 3d semantic instance segmentation using bounding boxes. In: European Conference on Computer Vision. pp. 681–699. Springer (2022) 5
9. Choy, C., Gwak, J., Savarese, S.: 4d spatio-temporal convnets: Minkowski convolutional neural networks. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 3075–3084 (2019) 4, 5, 9, 10, 11
10. Dai, A., Chang, A.X., Savva, M., Halber, M., Funkhouser, T., Nießner, M.: Scannet: Richly-annotated 3d reconstructions of indoor scenes. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 5828–5839 (2017) 9
11. Genova, K., Yin, X., Kundu, A., Pantofaru, C., Cole, F., Sud, A., Brewington, B., Shucker, B., Funkhouser, T.: Learning 3d semantic segmentation with only 2d image supervision. In: 2021 International Conference on 3D Vision (3DV). pp. 361–372 (2021) 5
12. Ghiasi, G., Gu, X., Cui, Y., Lin, T.Y.: Scaling open-vocabulary image segmentation with image-level labels. In: European Conference on Computer Vision. pp. 540–557. Springer (2022) 3, 4, 6, 7
13. Hegde, D., Valanarasu, J.M.J., Patel, V.: Clip goes 3d: Leveraging prompt tuning for language grounded 3d recognition. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 2028–2038 (2023) 1

14. Hou, J., Xie, S., Graham, B., Dai, A., Nießner, M.: Pri3d: Can 3d priors help 2d representation learning? In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 5693–5702 (2021) 5
15. Hu, Q., Yang, B., Fang, G., Guo, Y., Leonardis, A., Trigoni, N., Markham, A.: Sqn: Weakly-supervised semantic segmentation of large-scale 3d point clouds. In: European Conference on Computer Vision. pp. 600–619. Springer (2022) 5
16. Hu, Q., Yang, B., Xie, L., Rosa, S., Guo, Y., Wang, Z., Trigoni, N., Markham, A.: Randla-net: Efficient semantic segmentation of large-scale point clouds. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 11108–11117 (2020) 1
17. Hu, W., Zhao, H., Jiang, L., Jia, J., Wong, T.T.: Bidirectional projection network for cross dimension scene understanding. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 14373–14382 (2021) 5
18. Kweon, H., Yoon, K.J.: Joint learning of 2d-3d weakly supervised semantic segmentation. Advances in Neural Information Processing Systems **35**, 30499–30511 (2022) 2, 5, 11
19. Lahoud, J., Ghanem, B.: 2d-driven 3d object detection in rgb-d images. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 4622–4630 (2017) 5
20. Li, J., Dai, H., Han, H., Ding, Y.: Mseg3d: Multi-modal 3d semantic segmentation for autonomous driving. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 21694–21704 (2023) 5
21. Li, J., Jie, Z., Ricci, E., Ma, L., Sebe, N.: Enhancing robustness of vision-language models through orthogonality learning and cross-regularization (2024), https://arxiv.org/abs/2407.08374 5
22. Li, J., Jie, Z., Wang, X., Wei, X., Ma, L.: Expansion and shrinkage of localization for weakly-supervised semantic segmentation. In: Advances in Neural Information Processing Systems. vol. 35, pp. 16037–16051 (2022) 5
23. Li, J., Jie, Z., Wang, X., Zhou, Y., Ma, L., Jiang, J.: Weakly supervised semantic segmentation via self-supervised destruction learning. Neurocomputing **561**, 126821 (2023) 5
24. Li, J., Jie, Z., Wang, X., Zhou, Y., Wei, X., Ma, L.: Weakly supervised semantic segmentation via progressive patch learning. IEEE Transactions on multimedia **25**, 1686–1699 (2022) 5
25. Liang, F., Wu, B., Dai, X., Li, K., Zhao, Y., Zhang, H., Zhang, P., Vajda, P., Marculescu, D.: Open-vocabulary semantic segmentation with mask-adapted clip. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 7061–7070 (2023) 3, 4
26. Qi, C.R., Liu, W., Wu, C., Su, H., Guibas, L.J.: Frustum pointnets for 3d object detection from rgb-d data. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 918–927 (2018) 5
27. Qi, C.R., Su, H., Mo, K., Guibas, L.J.: Pointnet: Deep learning on point sets for 3d classification and segmentation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 652–660 (2017) 1, 10
28. Qi, C.R., Yi, L., Su, H., Guibas, L.J.: Pointnet++: Deep hierarchical feature learning on point sets in a metric space. Advances in neural information processing systems **30** (2017) 1, 11
29. Qian, G., Li, Y., Peng, H., Mai, J., Hammoud, H., Elhoseiny, M., Ghanem, B.: Pointnext: Revisiting pointnet++ with improved training and scaling strategies.

Advances in Neural Information Processing Systems **35**, 23192–23204 (2022) 1, 10, 11
30. Radford, A., Kim, J.W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., et al.: Learning transferable visual models from natural language supervision. In: International Conference on Machine Learning. pp. 8748–8763 (2021) 3, 4
31. Ren, Z., Misra, I., Schwing, A.G., Girdhar, R.: 3d spatial recognition without spatially labeled 3d. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 13204–13213 (2021) 2, 5, 10, 11
32. Robert, D., Vallet, B., Landrieu, L.: Learning multi-view aggregation in the wild for large-scale 3d semantic segmentation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 5575–5584 (2022) 10, 11
33. Tatarchenko, M., Park, J., Koltun, V., Zhou, Q.Y.: Tangent convolutions for dense prediction in 3d. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 3887–3896 (2018) 10, 11
34. Thomas, H., Qi, C.R., Deschaud, J.E., Marcotegui, B., Goulette, F., Guibas, L.J.: Kpconv: Flexible and deformable convolution for point clouds. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 6411–6420 (2019) 10, 11
35. Wang, Z., Rao, Y., Yu, X., Zhou, J., Lu, J.: Semaffinet: Semantic-affine transformation for point cloud segmentation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 11819–11829 (2022) 10, 11
36. Wei, J., Lin, G., Yap, K.H., Hung, T.Y., Xie, L.: Multi-path region mining for weakly supervised 3d semantic segmentation on point clouds. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 4384–4393 (2020) 2, 5, 10, 11
37. Xian, Y., Choudhury, S., He, Y., Schiele, B., Akata, Z.: Semantic projection network for zero-and few-label semantic segmentation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 8256–8265 (2019) 4
38. Xu, D., Anguelov, D., Jain, A.: Pointfusion: Deep sensor fusion for 3d bounding box estimation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 244–253 (2018) 5
39. Xu, J., Hou, J., Zhang, Y., Feng, R., Wang, Y., Qiao, Y., Xie, W.: Learning open-vocabulary semantic segmentation models from natural language supervision. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 2935–2944 (2023) 4
40. Xu, M., Zhang, Z., Wei, F., Hu, H., Bai, X.: Side adapter network for open-vocabulary semantic segmentation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 2945–2954 (2023) 4
41. Xu, M., Zhang, Z., Wei, F., Lin, Y., Cao, Y., Hu, H., Bai, X.: A simple baseline for open-vocabulary semantic segmentation with pre-trained vision-language model. In: European Conference on Computer Vision. pp. 736–753. Springer (2022) 4
42. Xu, X., Yuan, Y., Zhang, Q., Wu, W., Jie, Z., Ma, L., Wang, X.: Weakly-supervised 3d visual grounding based on visual linguistic alignment. arXiv preprint arXiv:2312.09625 (2023) 5
43. Yan, X., Gao, J., Zheng, C., Zheng, C., Zhang, R., Cui, S., Li, Z.: 2dpass: 2d priors assisted semantic segmentation on lidar point clouds. In: European Conference on Computer Vision. pp. 677–695. Springer (2022) 1

44. Yang, C.K., Chen, M.H., Chuang, Y.Y., Lin, Y.Y.: 2d-3d interlaced transformer for point cloud segmentation with scene-level supervision. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 977–987 (2023) 2, 5, 10, 11
45. Yang, C.K., Wu, J.J., Chen, K.S., Chuang, Y.Y., Lin, Y.Y.: An mil-derived transformer for weakly supervised point cloud segmentation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 11830–11839 (2022) 2, 10, 11
46. Yun, S., Park, S.H., Seo, P.H., Shin, J.: Ifseg: Image-free semantic segmentation via vision-language model. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 2967–2977 (2023) 3
47. Zhang, R., Wang, L., Qiao, Y., Gao, P., Li, H.: Learning 3d representations from 2d pre-trained models via image-to-point masked autoencoders. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 21769–21780 (2023) 5
48. Zhang, Y., Hu, Q., Xu, G., Ma, Y., Wan, J., Guo, Y.: Not all points are equal: Learning highly efficient point-based detectors for 3d lidar point clouds. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 18953–18962 (2022) 5
49. Zhao, H., Jiang, L., Jia, J., Torr, P.H., Koltun, V.: Point transformer. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 16259–16268 (2021) 10, 11
50. Zhou, B., Khosla, A., Lapedriza, A., Oliva, A., Torralba, A.: Learning deep features for discriminative localization. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 2921–2929 (2016) 2