Density-Guided Semi-Supervised 3D Semantic Segmentation with Dual-Space Hardness Sampling

Jianan Li[1,2], Qiulei Dong[*,1,2,3]
[1]School of Artificial Intelligence, University of Chinese Academy of Sciences,
[2]State Key Laboratory of Multimodal Artificial Intelligence Systems,
Institute of Automation, Chinese Academy of Sciences
[3]Center for Excellence in Brain Science and Intelligence Technology, Chinese Academy of Sciences
lijianan211@mails.ucas.ac.cn, qldong@nlpr.ac.cn

Abstract

Densely annotating the large-scale point clouds is laborious. To alleviate the annotation burden, contrastive learning has attracted increasing attention for tackling semi-supervised 3D semantic segmentation. However, existing point-to-point contrastive learning techniques in literature are generally sensitive to outliers, resulting in insufficient modeling of the point-wise representations. To address this problem, we propose a method named ***DDSemi*** *for semi-supervised 3D semantic segmentation, where a density-guided contrastive learning technique is explored. This technique calculates the contrastive loss in a point-to-anchor manner by estimating an anchor for each class from the memory bank based on the finding that the cluster centers tend to be located in dense regions. In this technique, an inter-contrast loss is derived from the perturbed unlabeled point cloud pairs, while an intra-contrast loss is derived from a single unlabeled point cloud. The derived losses could enhance the discriminability of the features and implicitly constrain the semantic consistency between the perturbed unlabeled point cloud pairs. In addition, we propose a dual-space hardness sampling strategy to pay more attention to the hard samples located in sparse regions of both the geometric space and feature space by reweighting the point-wise intra-contrast loss. Experimental results on both indoor-scene and outdoor-scene datasets demonstrate that the proposed method outperforms the comparative state-of-the-art semi-supervised methods.*

1. Introduction

3D semantic segmentation [13, 17, 23, 27, 30] is a fundamental task in computer vision and plays an essential role in scene understanding. Most of the existing 3D segmentation models in literature are trained in a fully-supervised manner, where the labor-intensive and time-consuming data annotation is required. To address this issue, several semi-supervised methods [11, 18, 20, 24, 36], weakly-supervised methods [25, 26, 28, 43], and annotation-free methods [5, 6, 39, 46] have been explored. Among them, semi-supervised 3D semantic segmentation has drawn growing interest, where the training data contains a small amount of densely-labeled data and a large amount of unlabeled data.

A typical manner for handling the semi-supervised semantic segmentation tasks [1, 18, 37, 48] is to apply contrastive learning to explore the information encapsulated in the unlabeled data while preserving the nutrition in the limited densely-labeled data.

Originating from the classification task, contrastive learning has become a prevailing technique in many visual tasks [7, 8, 12, 15]. However, as revealed in [32],

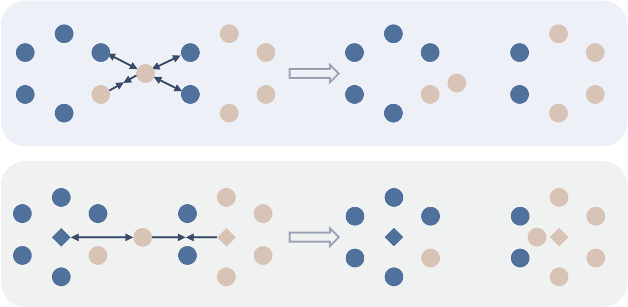


Figure 1. Illustration of the point-to-point contrast (top) and point-to-anchor contrast (bottom). The circles represent the point features and the diamonds represent the anchors. Different colors denote different categories. The direction of the arrows shows the force of pull or push in contrastive learning. The point-to-point contrast is prone to be affected by the confusing points while the point-to-anchor contrast is robust to them.

[*]Corresponding author

there exists a supervision gap between the classification task and the dense prediction tasks (*e.g*., the segmentation task). Contrastive learning in the classification task tends to focus on the most representative part for learning a discriminative representation, while the information contained in a single element (*e.g*., pixel or point) may not be representative enough for effective contrast in dense prediction tasks. Moreover, as stated in [33], directly utilizing point-to-point contrast usually results in insufficient modeling of the point-wise representations, for some confusing points may be sampled to construct the undesired pairs, as illustrated at the top part of Figure 1. The above issues motivate us to investigate the following problem: How to find efficacious supervisory signals for contrastive learning in semi-supervised 3D semantic segmentation?

To address this problem, we propose a method named **DDSemi** for semi-supervised 3D semantic segmentation. Inspired by the finding about clustering in [31] that the cluster centers tend to be located in dense regions and data located in sparser regions is less representative, we explore a density-guided contrastive learning technique for the unlabeled data. Specifically, we estimate an anchor for each class by using the high-density features stored in a memory bank. The anchors are regarded as the supervisory signals for the point-to-anchor contrastive learning. Each point is pulled closer to its corresponding anchor and pushed away from other anchors during the contrastive learning process, as illustrated at the bottom part of Figure 1.

In the explored density-guided contrastive learning technique, an inter-contrast loss and an intra-contrast loss are designed. The inter-contrast loss utilizes the anchors estimated from one perturbed point cloud and the features extracted from another perturbed point cloud for contrastive learning and vice versa, based on the assumption that semantic consistency should be maintained between the point clouds under different perturbations. The intra-contrast loss utilizes the anchors and features from the same point cloud for contrastive learning. The proposed contrastive learning technique not only enhances the discriminability of the features, but also implicitly constrains the semantic consistency between the perturbed point cloud pairs. Moreover, we propose a dual-space hardness sampling strategy for the unlabeled data to mine $hard$ points in both the geometric space and feature space. The $hard$ points are defined as the points located in sparse regions and more attention is paid to them by reweighting the point-wise intra-contrast loss.

In summary, our contributions are as follows:

- We propose a density-guided contrastive learning technique, where an inter-contrast loss and an intra-contrast loss are designed in a point-to-anchor manner. This technique could provide some insights to the application of contrastive learning in semi-supervised segmentation.
- We propose a dual-space hardness sampling strategy to pay more attention to the $hard$ points in both the geometric space and feature space. This strategy is also density-guided and explicitly shrinks the sparse regions, which is beneficial to the segmentation performances.
- By integrating the above contrastive learning technique and hardness sampling strategy, we propose a method for semi-supervised 3D semantic segmentation, named DDSemi. The effectiveness of DDSemi is demonstrated by the experimental results in Section 4.

2. Related Work

2.1. Fully-supervised 3D Semantic Segmentation

The existing methods for fully-supervised 3D semantic segmentation could be roughly divided into three categories: projection-based methods, voxel-based methods, and point-based methods.

The projection-based methods [19, 29, 34, 42, 45] generally project the point clouds into the image plane and use the 2D Convolutional Neural Networks (CNNs) or transformer blocks to extract features. Kong *et al*. [19] proposed a full-cycle framework and scalable training strategy to process the LiDAR point clouds from the range view.

The voxel-based methods [10, 14, 22, 44, 49] divide the 3D points into regular voxels and extract features from the discrete voxels. Lai *et al*. [22] designed the radial window self-attention and exponential splitting to mitigate the information disconnection and limited receptive field issues.

The point-based methods [16, 21, 38, 40, 47] take the raw point clouds as input and extract point-wise features for segmentation. Lai *et al*. [21] proposed the stratified strategy to enlarge the receptive field of the model and capture the long-range contexts at a low computational cost.

2.2. Semi-supervised 3D Semantic Segmentation

As defined in [41], semi-supervised 3D semantic segmentation [11, 18, 20, 24] aims at utilizing a small number of densely-labeled point cloud frames and a large number of unlabeled point cloud frames for model training, which could alleviate the annotation burden to some extent.

Deng *et al*. [11] focused on indoor-scene segmentation and proposed to optimize the pseudo labels for the unlabeled point using the superpoints generated by geometry-based and color-based region growing algorithms. Kong *et al*. [20] focused on segmenting the outdoor-scene LiDAR point clouds and proposed to leverage the spatial prior of LiDAR point clouds to exploit the unlabeled data. Also focusing on the outdoor-scene LiDAR point clouds, Li *et al*. [24] utilized the reflectivity-prior descriptors to generate high-quality pseudo labels and made use of the unreliable pseudo-labels for learning more discriminative representations. Jiang *et al*. [18] proposed the label-guided point-to-point contrastive learning for the unlabeled points

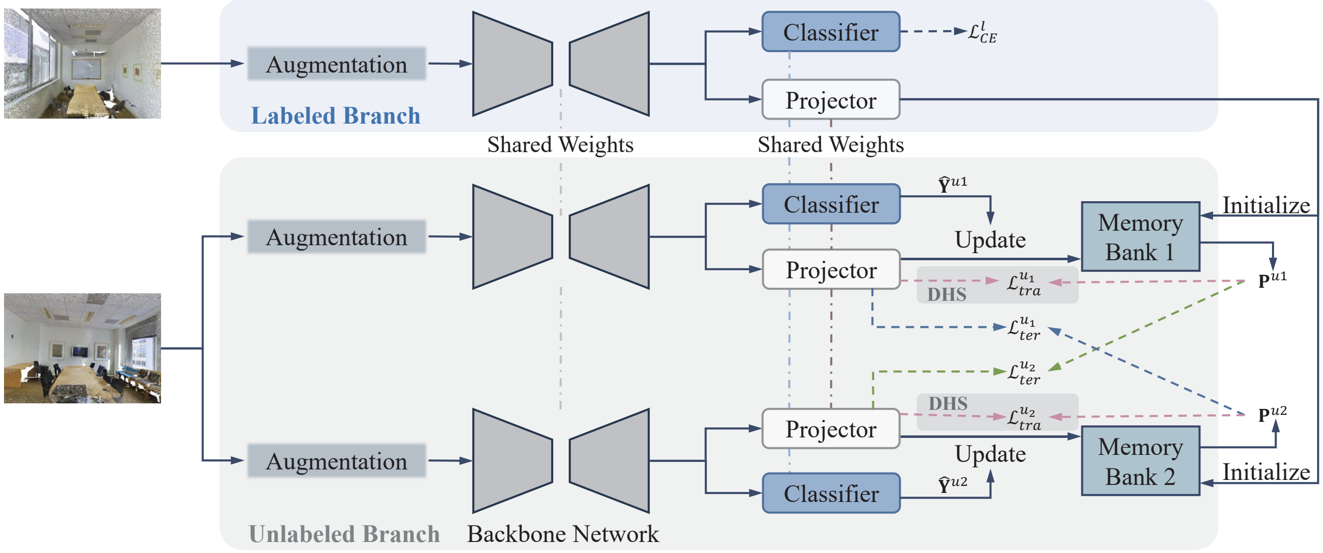


Figure 2. Architecture of the proposed DDSemi. It contains a labeled branch and an unlabeled branch. The weights of the backbone networks, classifiers, and projectors in two branches are shared. $\hat{\mathbf{Y}}^{u1}$ and $\hat{\mathbf{Y}}^{u2}$ are the pseudo labels predicted by the classifiers. $\mathbf{P}^{u1}$ and $\mathbf{P}^{u2}$ are the anchors estimated from the memory banks. The DHS in the unlabeled branch denotes the proposed dual-space hardness sampling strategy, which is elaborately introduced in Section 3.3.

to enhance the feature representation ability of the model.

Unlike existing method [18] that uses the point-to-point contrast, we design the density-guided contrastive learning technique to avoid the adverse effect caused by the undesired pairs by conducting the point-to-anchor contrast learning. In addition, unlike existing methods [11, 18, 20, 24] that treat all points equally, we propose the dual-space hardness sampling strategy to put more emphasis on the *hard* points located in sparse regions of both the geometric space and feature space.

3. Methodology

3.1. Architecture

Figure 2 depicts the architecture of the proposed DDSemi. As seen from this figure, it contains a labeled branch and an unlabeled branch.

The labeled branch takes the labeled point clouds as input. It consists of a backbone network for extracting features from the input data, a classifier for supervised segmentation, and a projector for mapping the extracted features to a novel feature space, as shown at the top part of Figure 2. The labeled branch is trained under the supervision of the ground-truth labels via the cross-entropy loss $\mathcal{L}^{l}_{CE}$.

The unlabeled branch takes the unlabeled point clouds as input. It contains two streams, which share the same architecture but use two different augmentation techniques. Each stream contains a backbone network, a classifier, a projector, and a memory bank, as shown at the bottom part of Figure 2. The memory bank is utilized to estimate a representative anchor for each class. It is initialized with the features output by the projector in the labeled branch and updated with the features output by its corresponding projector in the unlabeled branch. The density-guided contrastive learning technique contains the inter-contrast losses $\mathcal{L}^{u1}_{ter}, \mathcal{L}^{u2}_{ter}$ and the intra-contrast losses $\mathcal{L}^{u1}_{tra}, \mathcal{L}^{u2}_{tra}$, which are calculated between the anchors and features output by the projectors. The dual-space hardness sampling strategy is used for reweighting each point in the intra-contrast losses to pay more attention to the *hard* points.

At the training stage, the labeled branch is first pretrained. Then, the labeled branch and the unlabeled branch are trained jointly. At the inference stage, only the backbone network and classifier are used for segmentation.

3.2. Density-guided Contrastive Learning

To effectively exploit the unlabeled points, we propose the density-guided contrastive learning technique that calculates the contrastive loss in a point-to-anchor manner.

In this technique, a category-wise memory bank is built in each stream of the unlabeled branch to store high-quality features. Then, an anchor is estimated for each class using the features stored in the memory bank. The anchors are utilized to construct the point-anchor pairs for contrastive learning. This technique includes an inter-contrast loss calculated between the perturbed point cloud pairs and an intra-contrast calculated within the same point cloud. In this subsection, we will introduce the above parts in detail.

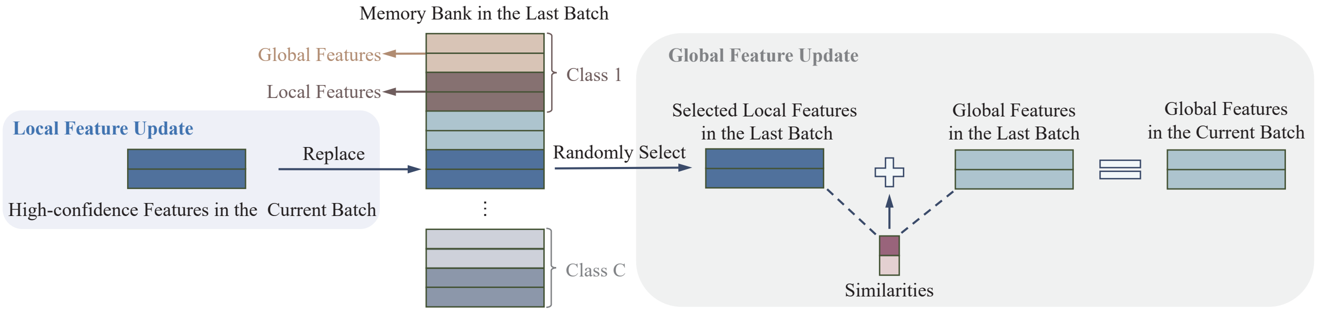

Memory Bank in the Last Batch
Global Features
Local Features
Class 1
Local Feature Update
Replace
High-confidence Features in the Current Batch
Randomly Select
Global Feature Update
Selected Local Features in the Last Batch
Global Features in the Last Batch
Global Features in the Current Batch
Similarities
Class C

Figure 3. Illustration of the update process of the category-wise memory bank. Different colors represent different categories. The light colors represent the global part and the corresponding dark colors represent the local part. We first progressively update the global features with the local features in the last batch. Then we update the local features with the high-confidence features in the current batch.

Memory Bank Construction. Ideally, a memory bank should contain the holistic information of the whole dataset for estimating representative anchors and the distribution of the features in the memory bank should be consistent with that of the latest features output by the projector. To this end, we construct a category-wise memory bank $\mathbf{M} = \{M_c\}_{c=1}^{C}$ in each stream of the unlabeled branch, where C denotes the number of classes and M_c stores a set of global features from previous batches and a set of local features from the current batch.

As mentioned in Section 3.1, the memory bank is initialized with the features output by the projector in the labeled branch. Then, the global and local features in the memory bank are updated respectively at each training iteration. Figure 3 illustrates the update process of the memory bank. As seen from the right part of this figure, the global features are updated progressively. For class c, we first randomly select a local feature $f^l_{c,j}$ for the global feature $f^g_{c,i}$. Then, we calculate the cosine similarity β_{ij} between $f^g_{c,i}$ and $f^l_{c,j}$. Finally, the updated global feature $\hat{f}^g_{c,i}$ is obtained by the weighted sum of $f^g_{c,i}$ and $f^l_{c,j}$. The above update process is formulated as:

$$\hat{f}^g_{c,i} = (1-\beta_{i,j}) \cdot f^g_{c,i} + \beta_{i,j} \cdot f^l_{c,j}. \tag{1}$$

As seen from the left part of Figure 3, the local features are updated with high-confidence features from the current batch in a Fist-In-First-Out (FIFO) manner.

Density-guided Anchor Estimation. Here, we develop a density-guided anchor estimation strategy to estimate an anchor for each class, based on the finding that the cluster centers tend to be located in dense regions. As seen from Figure 4, given a feature $f_{c,i}$ in M_c, we first search its k-nearest neighbors $\mathcal{N}_f(f_{c,i})$ in M_c. Then, we leverage the cosine similarities between $f_{c,i}$ and $\mathcal{N}_f(f_{c,i})$ to estimate the density $d(f_{c,i})$, which is formulated as:

$$d(f_{c,i}) = \frac{1}{|\mathcal{N}_f(f_{c,i})|} \sum_{f_{c,j} \in \mathcal{N}_f(f_{c,i})} \frac{f^{\mathrm{T}}_{c,i} \cdot f_{c,j}}{\|f_{c,i}\| \cdot \|f_{c,j}\|}, \tag{2}$$

where $\|\cdot\|$ denotes the L_2-Norm.

The anchor p_c of class c is estimated by the features with the top-K densities in its corresponding memory bank M_c:

$$p_c = \frac{\sum_{k \in \Omega_c} d(f_{c,k}) \cdot f_{c,k}}{\sum_{k \in \Omega_c} d(f_{c,k})}, \tag{3}$$

where Ω_c denotes the index set of the features with the top-K densities in M_c.

Density-guided Inter-contrast Loss. As seen from Figure 2, two different augmentation techniques are used for the same unlabeled point cloud. Thus, the semantic information should be consistent between the perturbed unlabeled point cloud pairs (*e.g.*, the anchors and semantic predictions), which means that the anchors from one perturbed point cloud could serve as the supervisory signals for another perturbed point cloud, and vice versa. According to this assumption, the following point-wise inter-contrast losses are designed:

$$\mathcal{L}^{u1}_{e,i} = -\log \frac{\exp(f^{u1}_i \cdot p^{u2}_{\hat{y}^{u1}_i}/\tau)}{\exp(\sum_{c=1}^{C} f^{u1}_i \cdot p^{u2}_c/\tau)}, \tag{4}$$

$$\mathcal{L}^{u2}_{e,i} = -\log \frac{\exp(f^{u2}_i \cdot p^{u1}_{\hat{y}^{u2}_i}/\tau)}{\exp(\sum_{c=1}^{C} f^{u2}_i \cdot p^{u1}_c/\tau)}, \tag{5}$$

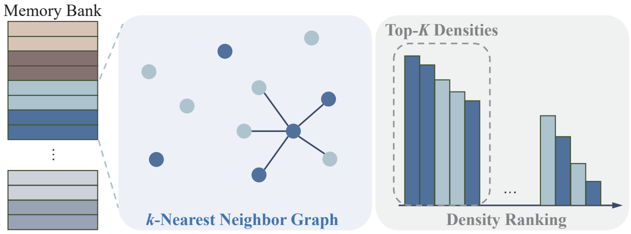


Figure 4. Illustration of the density-guided anchor estimation. The density of each feature is calculated within the memory bank based on its k-nearest neighbors. Only the features with the top-K densities are used for estimating the anchors.

where f_i^{u1} and f_i^{u2} denote the features output by their corresponding projectors in the unlabeled branch, p_c^{u1} and p_c^{u2} denote the anchors of class c in $\mathbf{P}^{u1}$ and $\mathbf{P}^{u2}$ respectively, $\hat{y}_i^{u1}$ and $\hat{y}_i^{u2}$ denote the pseudo labels of the i-th point in two streams, and τ denotes the temperature hyper-parameter.

In addition, since the point-anchor pairs are constructed based on the pseudo labels and inaccurate pseudo labels may result in undesired contrast, we eliminate the points with low-confidence predictions. Then, the inter-contrast loss for the point cloud is derived by combing the point-wise inter-contrast losses in Equation (4) and Equation (5):

$$\mathcal{L}_{ter}^{u} = \frac{\sum\limits_{i=1}^{N} \mathbb{1}(w_i^{u1}>\gamma_e)\cdot\mathcal{L}_{e,i}^{u1}}{\sum\limits_{i=1}^{N} \mathbb{1}(w_i^{u1}>\gamma_e)} + \frac{\sum\limits_{i=1}^{N} \mathbb{1}(w_i^{u2}>\gamma_e)\cdot\mathcal{L}_{e,i}^{u2}}{\sum\limits_{i=1}^{N} \mathbb{1}(w_i^{u2}>\gamma_e)}, \quad (6)$$

where N is the number of the unlabeled points, w_i^{u1} and w_i^{u2} are the confidences output by their corresponding classifiers in the unlabeled branch, $\mathbb{1}(\cdot)$ is the indicator function, and γ_e is the confidence threshold in the inter-contrast loss.

Density-guided Intra-contrast Loss. The intra-contrast loss is calculated between the features and anchors from the same point cloud, which is explored to tighten the feature distribution. Similar to Equation (4) and Equation (5), the point-wise intra-contrast losses are formulated as:

$$\mathcal{L}_{a,i}^{u1} = -\log \frac{\exp(f_i^{u1}\cdot p_{\hat{y}_i^{u1}}^{u1}/\tau)}{\exp(\sum_{c=1}^{C} f_i^{u1}\cdot p_c^{u1}/\tau)}, \quad (7)$$

$$\mathcal{L}_{a,i}^{u2} = -\log \frac{\exp(f_i^{u2}\cdot p_{\hat{y}_i^{u2}}^{u2}/\tau)}{\exp(\sum_{c=1}^{C} f_i^{u2}\cdot p_c^{u2}/\tau)}. \quad (8)$$

Likewise, the points with low-confidence predictions are also eliminated here for reliable contrast.

3.3. Dual-space Hardness Sampling

The aforementioned density-guided contrastive learning technique focuses on the high-density points in the feature space but neglects the low-density points. To make use of the low-density points effectively, we propose the dual-space hardness sampling strategy that takes the structural information of the point clouds into account by mining *hard* points in both the geometric space and feature space.

Considering the uneven densities of the point clouds, we define the *hard* points in the geometric space as the points located in sparse regions, due to the paucity of spatial adjacency information. The geometric density of each point is calculated according to its local structural information. Specifically, given the Cartesian coordinates of an unlabeled point x_i^u, its geometric density $d_g(x_i^u)$ is calculated based on the number of its neighbors within a certain radius R:

$$d_g(x_i^u) = \frac{|\mathcal{N}_g(x_i^u, R)| - \min\limits_{j\in\mathbf{X}^u} |\mathcal{N}_g(x_j^u, R)|}{\max\limits_{j\in\mathbf{X}^u} |\mathcal{N}_g(x_j^u, R)| - \min\limits_{j\in\mathbf{X}^u} |\mathcal{N}_g(x_j^u, R)|}, \quad (9)$$

where $\mathcal{N}_g(\cdot, R)$ is the neighbor set of a given point within the radius R and $\mathbf{X}^u$ denotes the set of the unlabeled points.

Based on the finding in [31], we define the *hard* points in the feature space as the points whose features are located in sparse regions. And the density of each point feature is calculated in a point-to-memory fashion. Specifically, given a feature f_i with its pseudo label $\hat{y}_i$, we search its k-nearest neighbors $\mathcal{N}_f(f_i)$ in its corresponding memory bank $M_{\hat{y}_i}$, and its density $d(f_i)$ is calculated according to Equation (2).

Considering that the *hard* points are less representative, more emphasis needs to be put on them. Thus, we devise a density-based weighting function as follows:

$$r_i = \frac{\max(1-\frac{\alpha}{2}\cdot[d_g(x_i^u)+d(f_i)], \epsilon)}{\frac{1}{N}\sum_{j=1}^{N}\max(1-\frac{\alpha}{2}\cdot[d_g(x_i^u)+d(f_i)], \epsilon)}, \quad (10)$$

where α and ϵ are two predetermined hyper-parameters.

As seen from Equation (10), points with lower densities will obtain larger weights and these weights are integrated into the intra-contrast loss. Finally, the intra-contrast loss for the point cloud is derived by combining the point-wise intra-contrast losses in Equation (7) and Equation (8):

$$\mathcal{L}_{tra}^{u} = \frac{\sum\limits_{i=1}^{N} \mathbb{1}(w_i^{u1}>\gamma_a)\cdot r_i\cdot\mathcal{L}_{a,i}^{u1}}{\sum\limits_{i=1}^{N} \mathbb{1}(w_i^{u1}>\gamma_a)} + \frac{\sum\limits_{i=1}^{N} \mathbb{1}(w_i^{u2}>\gamma_a)\cdot r_i\cdot\mathcal{L}_{a,i}^{u2}}{\sum\limits_{i=1}^{N} \mathbb{1}(w_i^{u2}>\gamma_a)}, \quad (11)$$

where γ_a is the confidence threshold in $\mathcal{L}_{tra}^{u}$.

3.4. Total Loss

As seen from Figure 2, the labeled branch is trained with the cross-entropy loss $\mathcal{L}_{CE}^{l}$ and the unlabeled branch is trained with the density-guided contrastive losses $\mathcal{L}_{ter}^{u}$ and $\mathcal{L}_{tra}^{u}$.

The total loss is the weighted sum of the above losses:

$$\mathcal{L}_{total} = \mathcal{L}_{CE}^{l} + \lambda_{ter}\mathcal{L}_{ter}^{u} + \lambda_{tra}\mathcal{L}_{tra}^{u}, \quad (12)$$

where λ_{ter} and λ_{tra} are the weights of $\mathcal{L}_{ter}^{u}$ and $\mathcal{L}_{tra}^{u}$.

4. Experiments

4.1. Datasets and Evaluation Metric

The following outdoor-scene and indoor-scene datasets are used for evaluating the proposed DDSemi:

- **SemanticKITTI** [3] is a large-scale 3D outdoor driving-scene LiDAR dataset consisting of 22 sequences, among which 10 sequences are used for training, 1 sequence is used for validation, and 11 sequences are used for testing.
- **nuScenes** [4] is a large-scale 3D outdoor driving-scene LiDAR dataset, which contains 1000 scenes. According to the official splitting, 850 scenes are used for training and validation, and 150 scenes are utilized for testing.
- **S3DIS** [2] is a 3D indoor-scene dataset, which contains 13 object classes and 6 areas. Following the common split [18], we utilize Area 5 as the validation set and adopt the other five areas as the training set.

Method	SemanticKITTI [3]				nuScenes [4]			
	1%	10%	20%	50%	1%	10%	20%	50%
MeanTeacher [35]	45.4	57.1	59.2	60.0	51.6	66.0	67.1	71.7
CBST [50]	48.8	58.3	59.4	59.7	53.0	66.5	69.6	71.6
CPS [9]	46.7	58.7	59.6	60.5	52.9	66.3	70.0	72.5
LaserMix (Range View) [20]	43.4	58.8	59.4	61.4	49.5	68.2	70.6	73.0
LaserMix (Voxel) [20]	50.6	60.0	61.9	62.3	<u>55.3</u>	<u>69.9</u>	<u>71.8</u>	<u>73.2</u>
GPC [18]	54.1	62.0	62.5	62.8	-	-	-	-
LiM3D [24]	<u>58.4</u>	<u>62.2</u>	<u>63.1</u>	<u>63.6</u>	-	-	-	-
DDSemi	**59.3**	**65.1**	**66.3**	**67.0**	**58.1**	**70.2**	**74.0**	**76.5**

Table 1. Comparative results on the SemanticKITTI [3] and nuScenes [4] datasets with varying labeled ratios. All mIoU scores are given in percentage (%). The best results are in **bold** and the second best results are marked with <u>underlines</u>.

As done in previous works [20, 24], we use the mIoU (mean Intersection over Union) as the evaluation metric.

4.2. Implementation Details

We follow the basic experimental settings of LaserMix [20] and GPC [18] for evaluating the proposed DDSemi on the outdoor-scene and indoor-scene datasets. We store 500 features per class in the memory bank, with the global part and the local part accounting for half respectively. The k in k-nearest neighbors and K in anchor estimation are set to 8 and 16. The hyper-parameters γ_e, γ_a, τ, α, ϵ, λ_{ter}, and λ_{tra} are set to 0.9, 0.75, 1, 0.9, 0.1, 0.1, and 1. More details are introduced in the supplementary material.

4.3. Comparative Evaluation

We first evaluate the proposed DDSemi on the outdoor-scene datasets (SemanticKITTI [3] and nuScenes [4]) in comparison to the state-of-the-art (SoTA) methods (GPC [18], LaserMix [20], and LiM3D [24]) that are specially designed for semi-supervised 3D semantic segmentation. We also extend some classic semi-supervised learning methods in the 2D domain to 3D segmentation for further comparison, including MeanTeacher [35], CBST [50], and CPS [9].

As done in [20], we set the labeled ratio of the outdoor-scene datasets as $\{1\%, 10\%, 20\%, 50\%\}$, and the corresponding results are reported in Table 1. As seen from this table, the proposed DDSemi outperforms the comparative methods on both two datasets.

Method	S3DIS [2]				
	5%	10%	20%	30%	40%
MeanTeacher [35]	46.3	53.3	60.1	61.5	62.6
CBST [50]	48.7	54.0	60.3	61.8	62.9
CPS [9]	48.5	54.4	60.9	62.0	63.4
SSS-Net [11]	-	51.1	55.5	-	-
GPC [18]	<u>53.0</u>	<u>57.7</u>	<u>63.5</u>	<u>64.9</u>	<u>65.0</u>
DDSemi	**63.2**	**66.8**	**70.3**	**70.5**	**70.8**

Table 2. Comparative results on the S3DIS dataset [2].

We also evaluate the proposed DDSemi on the indoor-scene dataset (S3DIS [2]), where the extended 2D methods (MeanTeacher, CBST, and CPS) and the SoTA methods (SSS-Net [11] and GPC) in semi-supervised 3D indoor-scene segmentation are compared. Note that LaserMix and LiM3D are specially designed for LiDAR point cloud segmentation, which leverage the spatial cues and the reflectivity of LiDAR point clouds respectively, whereas the point clouds in the S3DIS dataset are reconstructed from multi-view RGB-D images. Thus, LaserMix and LiM3D are not compared in the indoor-scene experiments.

As done in [18], we set the labeled ratio of the indoor-scene dataset as $\{5\%, 10\%, 20\%, 30\%, 40\%\}$. The corresponding results reported in Table 2 show that the proposed DDSemi achieves the best performances, which are consistent with the results in the outdoor-scene experiments and

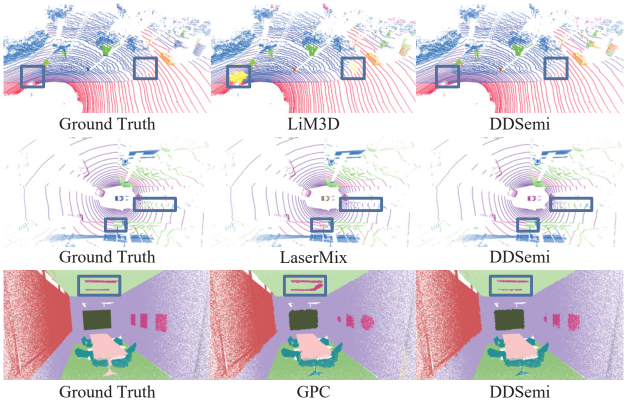


Figure 5. Visualization of the semantic segmentation results on the SemanticKITTI [3] (top), nuScenes [4] (middle), and S3DIS [2] (bottom) datasets by our proposed DDSemi and the second-best methods. All models are trained with 10% labeled data and the highlighted areas are marked with blue boxes.

further verify the effectiveness of our method.

Moreover, we visualize the segmentation results of the proposed DDSemi and the second-best methods on the three datasets in Figure 5. As seen from this figure, DDSemi outperforms its second-best counterparts, which shows the effectiveness of DDSemi from the qualitative perspective.

4.4. Ablation Studies

We conduct ablation studies on SemanticKITTI [3] to verify the effectiveness of each key element in DDSemi.

Effectiveness of the involved components. In the proposed DDSemi, the labeled data is trained with the cross-entropy loss $\mathcal{L}^l_{CE}$, and the unlabeled data is trained with the inter-contrast loss $\mathcal{L}^u_{ter}$ and intra-contrast loss $\mathcal{L}^u_{tra}$. The dual-space hardness sampling (DHS) strategy is adopted to reweight each point in $\mathcal{L}^u_{tra}$, which includes the $hard$ point mining in both the geometric space and feature space. We progressively add these components into training to verify their effectiveness. The experiments are conducted with 10% labeled data and the results are reported in Table 3.

The results in the first two rows of Table 3 indicate that the proposed inter-contrast loss is beneficial to mine extra information from the unlabeled data by constraining the semantic consistency between the perturbed point cloud pairs. The result in the third row of Table 3 shows that the intra-contrast loss could further improve the segmentation performance. The results in the last three rows of Table 3 demonstrate the effectiveness of the proposed DHS strategy from the quantitative perspective. To further verify the effectiveness of the DHS strategy, we visualize the features in Figure 6. As seen from this figure, the proposed DHS strategy is beneficial to the compactness of the intra-class feature distribution and the separation of the inter-class feature distribution, leading to better performances.

Effect of the contrastive learning strategy. To evaluate the effect of the contrastive learning strategy, we change the proposed point-to-anchor strategy to the point-to-point strategy, and the corresponding results are reported in Table 4. As seen from this table, the model trained with the point-to-anchor strategy achieves better performance, which demonstrates the effectiveness of the proposed density-guided point-to-anchor contrastive learning technique.

$\mathcal{L}^l_{CE}$	$\mathcal{L}^u_{ter}$	$\mathcal{L}^u_{tra}$	DHS		mIoU (%)
			GS	FS	
✓					56.1
✓	✓				59.7
✓	✓	✓			61.3
✓	✓	✓	✓		61.9
✓	✓	✓		✓	63.8
✓	✓	✓	✓	✓	**65.1**

Table 3. Ablation studies of the involved elements in the proposed method. DHS denotes dual-space hardness sampling. GS and FS stand for the geometric space and feature space respectively.

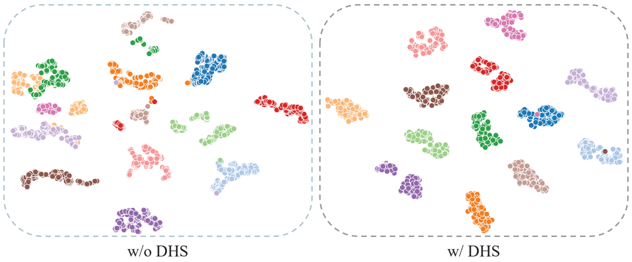


Figure 6. Visualization of the features sampled from the S3DIS dataset [2]. 200 features are sampled for each category. DHS denotes the dual-space hardness sampling strategy. Different colors represent different categories.

Effect of the update strategy of the memory bank. As mentioned in Section 3.2, the memory bank stores the global and local features, which are updated in a progressive manner and in a FIFO manner respectively. We evaluate the effect of the update strategy by replacing it with a pure progressive strategy and a pure FIFO strategy. The corresponding results reported in Table 5 show that our strategy achieves the best performance. This is mainly because the progressive strategy only focuses on holistic information and neglects the latest feature distribution. The FIFO strategy only focuses on the features in the current batch and overlooks the overall information of each class. Our strategy combines the advantages of these two complementary strategies, which leads to better performance.

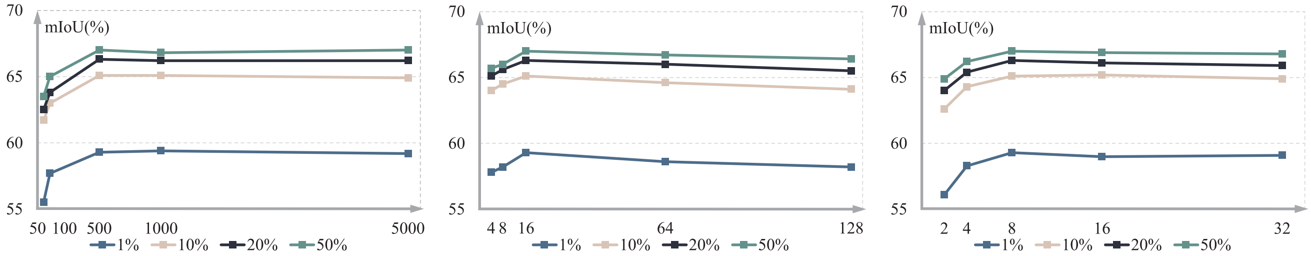
Effect of the size of the memory bank. We define the size of the memory bank as the number of stored features per class. To investigate its effect, we evaluate the proposed method with the size set as $\{50, 100, 500, 1000, 5000\}$. Figure 7a shows the corresponding results. As seen from this figure, a larger size improves the performance of the model

Strategy	1%	10%	20%	50%
Point-to-point	55.6	63.1	64.0	64.8
Point-to-anchor	**59.3**	**65.1**	**66.3**	**67.0**

Table 4. Ablation studies of the contrastive learning strategy.

Strategy	1%	10%	20%	50%
Progressive	58.7	64.3	65.5	66.3
FIFO	58.8	64.5	65.7	66.2
Ours	**59.3**	**65.1**	**66.3**	**67.0**

Table 5. Ablation studies of the update strategy.



(a) The size of the memory bank.

(b) K in anchor estimation.

(c) k in k-nearest neighbors.

Figure 7. Ablation studies of the size of the memory bank, K in anchor estimation, and k in k-nearest neighbors, which are the number of per-class features stored in the memory bank, the number of features used for anchor estimation, and the number of neighbors respectively.

at first (*e.g.*, from 50 to 500), because a stronger capacity enables the memory bank to give a more comprehensive description for each class. When the size increases to a certain level (*e.g.*, 1000, 5000), the model is generally insensitive to it. Because the memory bank is utilized to estimate anchors from the features with the top-K densities, and thus only those high-density features could make a difference in the performance. Moreover, a large size brings more storage cost and computational cost when calculating the densities. Hence, we set the size as 500 here.

Effect of the estimation strategy of the anchors. We utilize the features with the top-K densities to estimate the anchors. To verify the effectiveness of our density-based estimation strategy, we change it with a random strategy and a confidence-based strategy. The random strategy utilizes the mean of K randomly selected features in the memory bank to estimate the anchors. The confidence-based strategy utilizes the weighted mean of the features with the top-K confidences to estimate the anchors. The corresponding results are reported in Table 6. As seen from this table, our density-based strategy achieves the best performances, which proves the validity of the assumption in [31] and demonstrates the superiority of our strategy.

Effect of K in anchor estimation. K denotes the number of features utilized for the anchor estimation. Here, we evaluate the proposed method with $K = \{4, 8, 16, 64, 128\}$ and show the results in Figure 7b. As seen from this figure, the K that is too small or too large may lead to performance degradation. This is mainly because a smaller K ensures the reliability of the features used for anchor estimation, while less information is contained in the anchors. A larger K indicates that the anchors contain information from more features, but features with lower densities may not be reliable enough for a valid estimation. Therefore, we set $K = 16$ with the best performances.

Strategy	1%	10%	20%	50%
Random	58.2	64.0	64.9	65.8
Confidence-based	58.4	64.3	65.1	66.0
Ours	**59.3**	**65.1**	**66.3**	**67.0**

Table 6. Ablation studies of the estimation strategy of the anchors.

Effect of k in k-nearest neighbors. k denotes the number of neighbors and affects the calculation of density. Here, we evaluate the proposed method with $k = \{2, 4, 8, 16, 32\}$ and depict the results in Figure 7c. As seen from this figure, when k is too small (*e.g.*, 2 or 4), the performances drop evidently, due to the insufficient perception of the vicinity of each point. The performances are relatively stable when $k = \{8, 16, 32\}$. A larger k facilitates a more holistic perception of the local region of each point, but the computational cost increases accordingly. Thus, we set $k = 8$ for a trade-off between the accuracy and computational cost.

5. Conclusion

In this work, we propose a method for semi-supervised 3D semantic segmentation, named DDSemi. In DDSemi, a density-guided contrastive learning technique is explored, which calculates the contrastive loss in a point-to-anchor manner. This technique contains an inter-contrast loss derived from the perturbed point cloud pairs and an intra-contrast loss derived from a single point cloud. In addition, a dual-space hardness sampling strategy is proposed to pay more attention to the $hard$ points located in sparse regions of both the geometric space and feature space by reweighting the point-wise intra-contrast loss. Experimental results on three public datasets demonstrate that the proposed DDSemi outperforms the comparative methods.

Acknowledgements

This work was supported by the National Natural Science Foundation of China (Grant Nos. 61991423, 62376269, U1805264), the Strategic Priority Research Program of the Chinese Academy of Sciences (Grant No. XDA27040811).

References

[1] Inigo Alonso, Alberto Sabater, David Ferstl, Luis Montesano, and Ana C Murillo. Semi-supervised semantic segmentation with pixel-level contrastive learning from a class-wise memory bank. In *ICCV*, pages 8219–8228, 2021. 1

[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 *CVPR*, pages 1534–1543, 2016. 5, 6, 7

[3] Jens Behley, Martin Garbade, Andres Milioto, Jan Quenzel, Sven Behnke, Cyrill Stachniss, and Jurgen Gall. Semantickitti: A dataset for semantic scene understanding of lidar sequences. In *ICCV*, pages 9297–9307, 2019. 5, 6, 7

[4] 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 *CVPR*, pages 11621–11631, 2020. 5, 6

[5] 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 *CVPR*, pages 7020–7030, 2023. 1

[6] Runnan Chen, You-Chen Liu, Lingdong Kong, Nenglun Chen, Xinge Zhu, Yuexin Ma, Tongliang Liu, and Wenping Wang. Towards label-free scene understanding by vision foundation models. In *NeurIPS*, 2023. 1

[7] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *ICML*, pages 1597–1607, 2020. 1

[8] Xinlei Chen, Saining Xie, and Kaiming He. An empirical study of training self-supervised vision transformers. In *ICCV*, pages 9620–9629, 2021. 1

[9] Xiaokang Chen, Yuhui Yuan, Gang Zeng, and Jingdong Wang. Semi-supervised semantic segmentation with cross pseudo supervision. In *CVPR*, pages 2613–2622, 2021. 6

[10] Christopher Choy, JunYoung Gwak, and Silvio Savarese. 4d spatio-temporal convnets: Minkowski convolutional neural networks. In *CVPR*, pages 3075–3084, 2019. 2

[11] Shuang Deng, Qiulei Dong, Bo Liu, and Zhanyi Hu. Superpoint-guided semi-supervised semantic segmentation of 3d point clouds. In *ICRA*, pages 9214–9220, 2021. 1, 2, 3, 6

[12] Ye Du, Zehua Fu, Qingjie Liu, and Yunhong Wang. Weakly supervised semantic segmentation by pixel-to-prototype contrast. In *CVPR*, pages 4320–4329, 2022. 1

[13] Siqi Fan, Qiulei Dong, Fenghua Zhu, Yisheng Lv, Peijun Ye, and Feiyue Wang. Scf-net: Learning spatial contextual features for large-scale point cloud segmentation. In *CVPR*, pages 14499–14508, 2021. 1

[14] Benjamin Graham, Martin Engelcke, and Laurens van der Maaten. 3d semantic segmentation with submanifold sparse convolutional networks. *CVPR*, pages 9224–9232, 2018. 2

[15] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *CVPR*, pages 9729–9738, 2020. 1

[16] Qingyong Hu, Bo Yang, Linhai Xie, Stefano Rosa, Yulan Guo, Zhihua Wang, Niki Trigoni, and Andrew Markham. Randla-net: Efficient semantic segmentation of large-scale point clouds. In *CVPR*, pages 11108–11117, 2020. 2

[17] Di Huang, Sida Peng, Tong He, Honghui Yang, Xiaowei Zhou, and Wanli Ouyang. Ponder: Point cloud pre-training via neural rendering. In *ICCV*, pages 16089–16098, 2023. 1

[18] Li Jiang, Shaoshuai Shi, Zhuotao Tian, Xin Lai, Shu Liu, Chi-Wing Fu, and Jiaya Jia. Guided point contrastive learning for semi-supervised point cloud semantic segmentation. In *ICCV*, pages 6403–6412, 2021. 1, 2, 3, 5, 6

[19] Lingdong Kong, Youquan Liu, Runnan Chen, Yuexin Ma, Xinge Zhu, Yikang Li, Yuenan Hou, Yu Qiao, and Ziwei Liu. Rethinking range view representation for lidar segmentation. In *ICCV*, pages 228–240, 2023. 2

[20] Lingdong Kong, Jiawei Ren, Liang Pan, and Ziwei Liu. Lasermix for semi-supervised lidar semantic segmentation. In *CVPR*, pages 21705–21715, 2023. 1, 2, 3, 6

[21] Xin Lai, Jianhui Liu, Li Jiang, Liwei Wang, Hengshuang Zhao, Shu Liu, Xiaojuan Qi, and Jiaya Jia. Stratified transformer for 3d point cloud segmentation. In *CVPR*, pages 8500–8509, 2022. 2

[22] Xin Lai, Yukang Chen, Fanbin Lu, Jianhui Liu, and Jiaya Jia. Spherical transformer for lidar-based 3d recognition. In *CVPR*, pages 17545–17555, 2023. 2

[23] Jianan Li and Qiulei Dong. Open-set semantic segmentation for point clouds via adversarial prototype framework. In *CVPR*, pages 9425–9434, 2023. 1

[24] Li Li, Hubert P. H. Shum, and Toby P. Breckon. Less is more: Reducing task and model complexity for 3d point cloud semantic segmentation. In *CVPR*, pages 9361–9371, 2023. 1, 2, 3, 6

[25] Mengtian Li, Yuan Xie, Yunhang Shen, Bo Ke, Ruizhi Qiao, Bo Ren, Shaohui Lin, and Lizhuang Ma. Hybridcr: Weakly-supervised 3d point cloud semantic segmentation via hybrid contrastive regularization. In *CVPR*, pages 14930–14939, 2022. 1

[26] Lizhao Liu, Zhuangwei Zhuang, Shangxin Huang, Xunlong Xiao, Tianhang Xiang, Cen Chen, Jingdong Wang, and Mingkui Tan. Cpcm: Contextual point cloud modeling for weakly-supervised point cloud semantic segmentation. In *ICCV*, pages 18413–18422, 2023. 1

[27] Youquan Liu, Lingdong Kong, Jun Cen, Runnan Chen, Wenwei Zhang, Liang Pan, Kai Chen, and Ziwei Liu. Segment any point cloud sequences by distilling vision foundation models. In *NeurIPS*, 2023. 1

[28] Zhengzhe Liu, Xiaojuan Qi, and Chi-Wing Fu. One thing one click: A self-training approach for weakly supervised 3d semantic segmentation. In *CVPR*, pages 1726–1736, 2021. 1

[29] Ryan Razani, Ran Cheng, Ehsan Taghavi, and Liu Bingbing. Lite-hdseg: Lidar semantic segmentation using lite harmonic dense convolutions. In *ICRA*, pages 9550–9556, 2021. 2

[30] Damien Robert, Hugo Raguet, and Loic Landrieu. Efficient 3d semantic segmentation with superpoint transformer. In *ICCV*, pages 17195–17204, 2023. 1

[31] Alex Rodriguez and Alessandro Laio. Clustering by fast search and find of density peaks. *Science*, 344:1492 – 1496, 2014. 2, 5, 8

[32] Wei Shen, Zelin Peng, Xuehui Wang, Huayu Wang, Jiazhong Cen, Dongsheng Jiang, Lingxi Xie, Xiaokang Yang, and Qi Tian. A survey on label-efficient deep image segmentation: Bridging the gap between weak supervision and dense prediction. *IEEE TPAMI*, 45:9284–9305, 2022. 1

[33] Xiaoxiao Sheng, Zhiqiang Shen, Gang Xiao, Longguang Wang, Yulan Guo, and Hehe Fan. Point contrastive prediction with semantic clustering for self-supervised learning on point cloud videos. In *ICCV*, pages 16515–16524, 2023. 2

[34] Kshitij Sirohi, Rohit Mohan, Daniel Büscher, Wolfram Burgard, and Abhinav Valada. Efficientlps: Efficient lidar panoptic segmentation. *IEEE TOR*, 38(3):1894–1914, 2022. 2

[35] Antti Tarvainen and Harri Valpola. Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results. In *NIPS*, 2017. 6

[36] Ozan Unal, Dengxin Dai, Lukas Hoyer, Yigit Baran Can, and Luc Van Gool. 2d feature distillation for weakly-and semi-supervised 3d semantic segmentation. In *WACV*, pages 7336–7345, 2024. 1

[37] Xiaoyang Wang, Bingfeng Zhang, Limin Yu, and Jimin Xiao. Hunting sparsity: Density-guided contrastive learning for semi-supervised semantic segmentation. In *CVPR*, pages 3114–3123, 2023. 1

[38] Xiaoyang Wu, Yixing Lao, Li Jiang, Xihui Liu, and Hengshuang Zhao. Point transformer v2: Grouped vector attention and partition-based pooling. In *NeurIPS*, pages 33330–33342, 2022. 2

[39] Xiaoyang Wu, Xin Wen, Xihui Liu, and Hengshuang Zhao. Masked scene contrast: A scalable framework for unsupervised 3d representation learning. In *CVPR*, pages 9415–9424, 2023. 1

[40] Peng Xiang, Xin Wen, Yu-Shen Liu, Hui Zhang, Yi Fang, and Zhizhong Han. Retro-fpn: Retrospective feature pyramid network for point cloud semantic segmentation. In *ICCV*, pages 17826–17838, 2023. 2

[41] Aoran Xiao, Xiaoqin Zhang, Ling Shao, and Shijian Lu. A survey of label-efficient deep learning for 3d point clouds. *arXiv preprint arXiv:2305.19812*, 2023. 2

[42] Chenfeng Xu, Bichen Wu, Zining Wang, Wei Zhan, Peter Vajda, Kurt Keutzer, and Masayoshi Tomizuka. Squeezesegv3: Spatially-adaptive convolution for efficient point-cloud segmentation. In *ECCV*, pages 1–19, 2020. 2

[43] Cheng-Kun Yang, Ji-Jia Wu, Kai-Syun Chen, Yung-Yu Chuang, and Yen-Yu Lin. An mil-derived transformer for weakly supervised point cloud segmentation. In *CVPR*, pages 11830–11839, 2022. 1

[44] Yu-Qi Yang, Yu-Xiao Guo, Jian-Yu Xiong, Yang Liu, Hao Pan, Peng-Shuai Wang, Xin Tong, and Baining Guo. Swin3d: A pretrained transformer backbone for 3d indoor scene understanding, 2023. 2

[45] Yang Zhang, Zixiang Zhou, Philip David, Xiangyu Yue, Zerong Xi, Boqing Gong, and Hassan Foroosh. Polarnet: An improved grid representation for online lidar point clouds semantic segmentation. In *CVPR*, pages 9601–9610, 2020. 2

[46] Zihui Zhang, Bo Yang, Bing Wang, and Bo Li. Growsp: Unsupervised semantic segmentation of 3d point clouds. In *CVPR*, pages 17619–17629, 2023. 1

[47] Hengshuang Zhao, Li Jiang, Jiaya Jia, Philip HS Torr, and Vladlen Koltun. Point transformer. In *ICCV*, pages 16259–16268, 2021. 2

[48] Yuanyi Zhong, Bodi Yuan, Hong Wu, Zhiqiang Yuan, Jian Peng, and Yu-Xiong Wang. Pixel contrastive-consistent semi-supervised semantic segmentation. In *ICCV*, pages 7273–7282, 2021. 1

[49] Xinge Zhu, Hui Zhou, Tai Wang, Fangzhou Hong, Yuexin Ma, Wei Li, Hongsheng Li, and Dahua Lin. Cylindrical and asymmetrical 3d convolution networks for lidar segmentation. In *CVPR*, pages 9939–9948, 2021. 2

[50] Yang Zou, Zhiding Yu, BVK Kumar, and Jinsong Wang. Unsupervised domain adaptation for semantic segmentation via class-balanced self-training. In *ECCV*, pages 289–305, 2018. 6