Computers & Graphics 116 (2023) 427–436

Contents lists available at ScienceDirect

Computers & Graphics

journal homepage: www.elsevier.com/locate/cag

Special Section on CAD/Graphics 2023

PointMatch: A consistency training framework for weakly supervised semantic segmentation of 3D point clouds

Yushuang Wu [a,b,c,*,1], Zizheng Yan [a,b,c,1], Shengcai Cai [a,c], Guanbin Li [d], Xiaoguang Han [b,a], Shuguang Cui [b,a]

[a] *FNii, CUHKSZ, China*
[b] *SSE, CUHKSZ, China*
[c] *SRIBD, China*
[d] *Sun Yat-sen University, China*

ARTICLE INFO

Article history:
Received 12 May 2023
Received in revised form 14 September 2023
Accepted 14 September 2023
Available online 16 September 2023

Keywords:
Point cloud analysis
Semantic segmentation
Consistency training

ABSTRACT

Semantic segmentation of point cloud usually relies on dense annotation that is exhausting and costly, so it attracts wide attention to investigate solutions for the weakly supervised scheme with only sparse points annotated. Existing works start from the given labels and propagate them to highly-related but unlabeled points, with the guidance of data, e.g. intra-point relation. However, it suffers from (i) the inefficient exploitation of data information, and (ii) the strong reliance on labels thus is easily suppressed when given much fewer annotations. Therefore, we propose a novel framework, PointMatch, that stands on both data and label, by applying consistency regularization to sufficiently probe information from data itself and leveraging weak labels as assistance at the same time. By doing so, meaningful information can be learned from both data and label for better representation learning, which also enables the model more robust to the extent of label sparsity. Simple yet effective, the proposed PointMatch achieves the state-of-the-art performance under various weakly-supervised schemes on both ScanNet-v2 and S3DIS datasets, especially on the settings with extremely sparse labels, e.g. surpassing SQN by 21.2% and 17.2% on the 0.01% and 0.1% setting of ScanNet-v2, respectively.

1. Introduction

Semantic segmentation of 3D point clouds is crucial for the application of intelligent robots' understanding scenes in the real world. Great efforts have been contributed to the fully supervised scheme, while it requires exhausting and costly per-point annotations (*e.g* around 22.3 min to annotate an indoor scene on average [1]). Thus, weakly supervised 3D semantic segmentation now receives increasing attention, where only limited point-level annotations are provided in each point cloud.

Recently, several approaches are proposed for weakly supervised point cloud semantic segmentation with different kinds of weak labels, including projected 2D image [5], subcloud-level [6], segment-level [7], and point-level [2–4,8] supervision. In this paper, we focus on addressing the setting of sparse point-level labels, which is one of the most convenient annotation schemes in the application. The key challenge of this task is the difficulty of learning a robust model given very sparse supervision in the point cloud (*e.g* 0.1%, 0.01% of points annotated in [2] and around 0.02% in [4]). Existing solutions are mainly committed to alleviating the label sparsity by reusing limited supervision, *i.e*, first probing the highly-related points [2] or super-voxels [4] and allowing them to share the same training labels. However, this line of works is explicitly constructed on label propagation and employs point cloud data as the propagation guidance, which suffers from (i) the insufficient exploitation of data information limits the learning efficiency, and (ii) the propagated labels strongly rely on the original annotation scale thus the performance is easily suppressed when given much fewer labels. Therefore, we propose to probe information from both label and data itself for more efficient and robust representation learning.

Recently, consistency training is acknowledged as a powerful algorithmic paradigm for robust learning from label-scarce data, *e.g* in unsupervised/semi-supervised learning [9–12] and unsupervised/semi-supervised domain adaptation [13–15]. It works

* Correspondence to: The Chinese University of Hong Kong (Shenzhen), Longgang District, Shenzhen, 518172, China.
E-mail addresses: yushuangwu@link.cuhk.edu.cn (Y. Wu), zizhengyan@link.cuhk.edu.cn (Z. Yan), shengcaicai@link.cuhk.edu.cn (S. Cai), liguanbin@mail.sysu.edu.cn (G. Li), hanxiaoguang@cuhk.edu.cn (X. Han), shuguangcui@cuhk.edu.cn (S. Cui).

[1] Equal Contribution.

https://doi.org/10.1016/j.cag.2023.09.006
0097-8493/© 2023 Elsevier Ltd. All rights reserved.

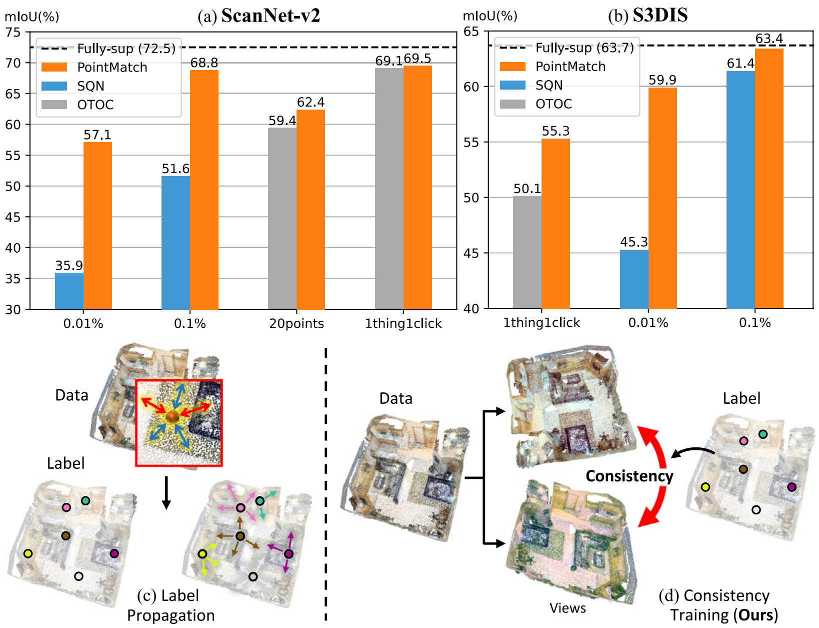

Fig. 1. (a), (b) the performance of PointMatch on the ScanNet-v2 and S3DIS datasets over various weakly supervised semantic segmentation settings: annotating 0.01%, 0.1% of points [2], 20 points per-scene [3], and "1thing1click" [4]. (c), (d) a comparison between previous works and the proposed approach.

by forcing the model to make consistent prediction under different perturbations/augmentations to the input sample (named as different *views*) and the prediction in one view usually serve as the pseudo-label of the other view. Inspired by this, we propose a novel consistency training framework, PointMatch, for the weakly supervised 3D semantic segmentation. Given a whole scene of point cloud with sparse labels, PointMatch employs the per-point prediction in one view as the other's pseudo-label to encourage the predictive consistency between two views of a scene. Such consistency facilitates (i) robustness to easily-perturbed low-level input features and (ii) stronger capability in learning useful high-level representations to keep predictive consistency. Besides, the provided labels act as extra supervision to assist high-level semantic feature discrimination, which also benefits the representation learning from data. By doing so, the reliance on the given label is relieved and more information is probed from the point cloud data itself.

Originating from the per-point prediction in one view, the pseudo-label should be of high quality to provide positive guidance for the other. Whereas there exist considerable mispredictions especially at the early learning stage. Thus, we exploit the inherent structure of the point cloud to improve the pseudo-label quality, via integrating the super-point grouping information where similar points are clustered by low-level features (*e.g* position and color) into the same group and are assumed to have the same semantic meaning. Specifically, the grouping information is used to correct the minor predictions that diverge from the "mainstream" in the super-point. Despite its good property, the super-point-aware pseudo-label actually introduces noise from the pretext super-point generation. Therefore, to fully utilize these two types of pseudo-labels, we design an adaptive pseudo-labeling mechanism, where the model is encouraged to believe the super-point-aware pseudo-label more at the beginning, and gradually resorts to its raw prediction when the model itself is reliable enough. Extensive experiments and analysis on the ScanNet-v2 [1] and S3DIS [16] dataset validate the effectiveness of the proposed approach. As shown in Fig. 1, the proposed PointMatch significantly surpasses the state of the art on various weakly supervised schemes and impressively, shows great robustness given extremely sparse labels.

The contributions of this paper are listed as follows:

- We propose a novel consistency training framework, PointMatch, for the weakly supervised 3D semantic segmentation, which can facilitate the network to learn robust representation from sparse labels and point cloud data.
- We introduce super-point information to promote the pseudo-label quality in our framework, and it is employed in an adaptive manner to well utilize the advantages of both two types of pseudo-label.
- Extensive experiments validate the effectiveness and superiority of PointMatch, and the proposed approach achieves significant improvements beyond the state of the art over various weakly-supervised settings.

2. Related work

Fully supervised 3D semantic segmentation. Semantic segmentation approaches for 3D point cloud can be mainly classified into two groups: point-based and voxel-based methods. Point-based Methods [17–23] apply convolutional kernels to a local region of points for feature extraction and the neighbors of a point are computed from k-NN or spherical search. In the case of voxel-based methods [24–27], the points in the 3D space are first transformed into voxel representations so that standard CNN can be adapted to process the structured voxels. In either point-based or voxel-based methods, feature aggregation is performed in the Euclidean space, while there are some recent works [28–31] that consider geodesic information for better feature representation. More recently, the Transformer structure [32,33] is also proposed for point clouds, as an alternative to the classic convolutional structure. However, most of the above methods are designed for the fully-supervised scheme [34–36], while annotation on point clouds is exhausting and costly, especially in the application of semantic segmentation, where the scene (indoor or outdoor) point cloud is usually of a large scale. In this work, we focus on weakly-supervised point cloud segmentation, where only very sparse points are annotated in each scene.

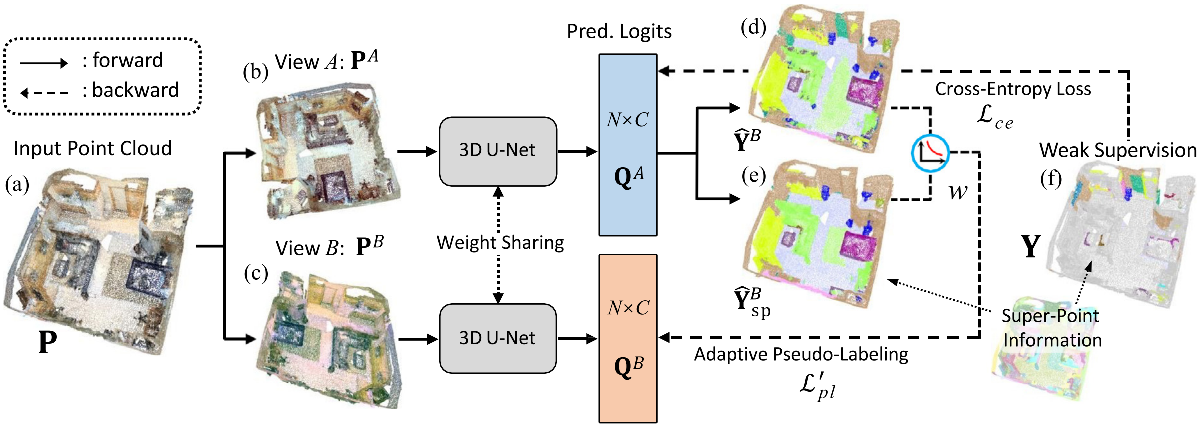

Fig. 2. The overview of PointMatch. (a) the input point cloud; (b) the view A augmented from the input point cloud; (c) view B generated via another augmentation; (d) the point-wise pseudo-label; (e) the super-point-wise pseudo-label; (f) the weak supervision ("1thing1click" setting), points in gray are unlabeled ones and other colors indicate different semantic meanings.

Weakly supervised 3D semantic segmentation. Existing works explore the 3D semantic segmentation with various types of weak supervision, including 2D image [5], subcloud-level [6], segment-level [7], and point-level supervision [2,4,8,37]. The first three types can be grouped into indirect annotations [2]. The work of [5] utilizes the annotations on the projected 2D image of a point cloud, with only a single view per sample. In [6], a classifier is trained first with sub-cloud labels, from which point-level pseudo labels can be generated via class activation mapping techniques [38]. In another way, the work of [7] pre-generates segments/super-points to extend sparse click annotation into segment-level supervision, and groups unlabeled segments into the relevant nearby labeled ones for label sharing. For point-level weak supervision, the work of [2] proposes to use only 10% of labels by learning gradient approximation and utilizing low-level smoothness constraints. A harder setting with a much lower label ratio, 1‰, is further investigated in [2], where a Semantic Query Network (SQN) is proposed based on leveraging the semantic similarity between neighboring points. Another work OTOC [4] proposes a novel weakly supervised setting, One Thing One Click ("1thing1click"), *i.e*, with only one point annotated for each instance in the scene. They employ an extra branch of network to probe the relation between super-points and propagate labels among highly-related ones. Besides, authors of [37] propose an active learning approach for annotating selected super-point with a limited budget to maximize model performance. Another line of work is contributed to self-supervised pre-training of 3D point clouds [3,39–42]. The pre-training usually needs weak or even no labels and provides a better network initialization for the downstream tasks. Besides, two related works are [43,44] which adopts consistency learning and super-point guidance, respectively.

Existing point-level weakly supervised 3D semantic segmentation methods act on label propagation by leveraging the relation between points/super-points. However, the proposed PointMatch takes a novel way based on consistency regularization to better probe information in the point cloud data itself and alleviates the reliance on the given labels.

Consistency training. Consistency training is a powerful algorithmic paradigm proposed for robust learning from label-scarce data. It is constructed on enforcing the prediction stability under different input transformations [45], *e.g* adversarial perturbations [46] or data augmentations [11,12], in the manner of pseudo-labeling, *i.e*, using the prediction of one transformation as the fitting target of the other. Thus it combines the advantages of both consistency regularization and pseudo-labeling (or self-training). This approach has been applied in many domains, such as semi-supervised learning (SSL) [11,12,47,48], unsupervised learning (USL) [9,10], unsupervised domain adaptation (UDA) [13,14], and semi-supervised domain adaptation (SSDA) [15,49], all of which prove the effectiveness of consistency training in learning high-quality representations from label-scarce data. More recently, there are some works extending consistency training into other tasks, such as unsupervised domain adaptation for image segmentation [50] and semi-supervised 3D object detection [51].

To our knowledge, it is the first time that consistency training is applied in the weakly supervised semantic segmentation of 3D point clouds. Different from the previous works, consistency training is novelly used in a weakly-supervised scenario where limited point-wise supervision is provided in each training sample. In addition, our work properly leverages the super-point grouping information in point clouds to further improve the whole framework.

3. Methodology

3.1. Problem definition

We first formulate the weakly supervised 3D semantic segmentation problem, taking the indoor scene scenario as an example. Given the point cloud $\mathbf{P} \in \mathbb{R}^{N\times D}$ of a scene of N points with D-dimension features, there are only partial points annotated for training. The points with labels are denoted as $\{(\mathbf{x}^l_i, y_i), i \in L\}$, and other unlabeled points are denoted as $\{\mathbf{x}^u_i, i \in U\}$, where L and U are two sets, satisfying $L \cap U = \varnothing$ and $L \cup U = \langle N\rangle$ ($\langle N\rangle$ is a short form of $\{1, 2, \dots, N\}$, the same hereinafter). The target of f is to predict the semantic category $y_i \in \langle C\rangle$ of each point $\mathbf{x}_i$, where C is the number of possible categories. Taking the point cloud $\mathbf{P}$ as input, f outputs the prediction probability $\mathbf{Q} \in [0, 1]^{N\times C}$ over all C classes, for all N points of $\mathbf{P}$. Note that the summation of values in each row of $\mathbf{Q}$ is equal to 1. Denote the weak semantic label of the whole scene as $\mathbf{y} \in \langle C\rangle^N$ and its one-hot extension as $\mathbf{Y} \in \{0, 1\}^{N\times C}$. To optimize f, a straightforward way is to compute the cross-entropy loss $\mathcal{L}_{ce}$ between $\mathbf{Q}$ and $\mathbf{Y}$, formulated as:

$$\mathcal{L}_{ce} = \frac{1}{|L|}\sum_{i\in L}\text{cross-entropy}(\mathbf{Q}_i,\ \mathbf{Y}_i), \tag{1}$$

where $|L|$ represents the set size of L and the subscript i indicates the row index, so $\mathbf{Q}_i$ and $\mathbf{Y}_i$ are two C-class distributions corresponding to the ith point. At the inference stage, the semantic segmentation result of a scene can be generated from f's prediction, by simply choosing the class with the highest score in each row of $\mathbf{Q}$.

To probe more information the limited labels and point cloud data itself, we design a novel framework, PointMatch, with the pipeline illustrated in Fig. 2. It conducts a consistency training framework designed for weakly labeled point clouds, and an adaptive pseudo-labeling mechanism by incorporating the super-point information, described in the following Section 3.2 and Section 3.3, respectively.

3.2. Consistency training

The proposed consistency training framework focuses on better exploitation of data itself, by encouraging the model's point-wise predictive consistency between two views of an input scene, by employing the prediction in one view as the pseudo-label of the other. Such a consistency training approach has three advantages: (i) various augmentations enable the network robust to different kinds of perturbation on low-level input features; (ii) the consistency target facilitates the model's ability in extracting high-level semantic features from the point cloud data itself; (iii) the self-training process implicitly propagates sparse training signals to unlabeled points and provide dense pseudo-labels, which increases the learning stability.

Formally, given a point cloud $\mathbf{P} \in \mathbb{R}^{N\times D}$, our PointMatch applies two different groups of data augmentations to create its two views $\mathbf{P}^A \in \mathbb{R}^{N\times D}$ and $\mathbf{P}^B \in \mathbb{R}^{N\times D}$, respectively. To avoid breaking the local structure of the point cloud too much, we perform scene-level augmentations like offsetting, scaling, rotation, flipping, jittering, *etc*. The obtained two views $\mathbf{P}^A$ and $\mathbf{P}^B$ are then fed into the 3D U-Net f_θ for point-wise semantic prediction, where θ is the network parameters. The network f_θ outputs the per-point probability distribution of $\mathbf{P}^A$, denoted as $\mathbf{Q}^A \in [0, 1]^{N\times C}$, and similarly, $\mathbf{Q}^B \in [0, 1]^{N\times C}$ can be generated from $\mathbf{P}^B$, formulated as:

$$\begin{aligned}\mathbf{Q}^A &= f_\theta(\mathbf{P}^A),\\ \mathbf{Q}^B &= f_\theta(\mathbf{P}^B).\end{aligned} \tag{2}$$

In the next step, we generate the pseudo-label of $\mathbf{Q}^B$ from $\mathbf{Q}^A$ to create the self-consistency loop. Specifically, the most-likely predictive category of each point (as well as its confidence score) is chosen to form the pseudo-label, *i.e*, the indices of the highest value in each row of $\mathbf{Q}^A$. However, $\mathbf{Q}^A$ is usually noisy and even contains many uncertain predictions, so a direct use may provide negative guidance to $\mathbf{Q}^B$ and harm the whole learning scheme. Hence, we conduct a filtering operation to improve the pseudo-label quality, by ignoring those predictions with confidence lower than a threshold τ. Denote the filtering mask as $\mathbf{m} \in [0, 1]^N$, which is generated as follows:

$$\mathbf{m}_i = \begin{cases} 1, & \max(\mathbf{Q}^A_i) \geq \tau, \\ 0, & \text{otherwise}, \end{cases} \quad \forall i \in \langle N\rangle, \tag{3}$$

where i is the row index of $\mathbf{Q}^A$ and τ is set as 0.95 in our implementation. Given $\mathbf{m}$ and the one-hot extension of $\mathbf{Q}^B$'s pseudo-label, represented as $\widehat{\mathbf{Y}}^B \in \{0, 1\}^{N\times C}$, the pseudo-labeling of $\mathbf{Q}^B$ can be conducted via a cross-entropy loss:

$$\mathcal{L}_{pl} = \frac{1}{N}\sum_{i\in\langle N\rangle} \mathbf{m}_i \cdot \text{cross-entropy}(\mathbf{Q}^B_i,\ \widehat{\mathbf{Y}}^B_i). \tag{4}$$

Until this point, we are working on probing information only from point cloud data itself for better data exploitation. Then the weak labels are integrated to provide discriminative semantic information, by using $\mathbf{Y}$ as the supervision of $\mathbf{Q}^A$ via computing a cross-entropy loss as in Eq. (1). The parameters θ can then be optimized by minimizing the objective loss function $\mathcal{L}_{total}$ as follows:

$$\min_\theta \mathcal{L}_{total} = \min_\theta \mathcal{L}_{ce} + \lambda\mathcal{L}_{pl}, \tag{5}$$

where λ is a scalar weight for balancing the two loss functions. As the learning process goes, the model exploits the knowledge learned from the limited annotations to train itself via forcing the predictive consistency, and meanwhile, implicitly propagates the sparse training signals to the whole scene via pseudo-labeling.

Algorithm 1: PointMatch in a PyTorch-like style.

```
# N, C: scalar, the number of points and classes
# f: 3D U-Net, input N x 6, output N x C
# x: tensor, the input point cloud, N x 6
# y: tensor, the one-hot weak label, N x C
# mask: the mask for weak label, N x 1
# groups: list, each element is a list of point indices
      belong to one super-point
# T_pt, T_sp: threshold 0.95, scalar
# w: the balance weight with an inverse decay

def augment(x):
    x_view1 = augment1(x)
    x_view2 = augment2(x)
    # x_view1, x_view2: N x 6
    return x_view1, x_view2

def correct(x_logit1, groups):
    ps_label_sp = torch.zeros(x_logit1.shape)
    for gp in group:
        ps_label_sp[gp] = x_logit1[gp].mean(dim=0)
    ps_label_sp = torch.softmax(ps_label_sp, dim=-1)
    # ps_label_sp: N x C
    return ps_label_sp

def CE(x, y, mask):
    loss = F.cross_entropy(x, y)
    loss = (loss * mask).mean()
    return loss

for x in dataset:
    # generate two views of the input point cloud
    x_view1, x_view2 = augment(x)

    # compute the prediction logits in two views
    x_logit1 = f(x_view1)
    x_logit2 = f(x_view2)

    # get point-wise pseudo-label: ps_label_pt
    scores = torch.softmax(x_logit1.detach(), dim=-1)
    max_probs, ps_label_pt = torch.max(scores, dim=-1)
    mask_pt = max_probs.ge(T_pt)

    # get super-point-wise pseudo-label: ps_label_sp
    scores = correct(x_logit1, groups)
    max_probs, ps_label_sp = torch.max(scores, dim=-1)
    mask_pt = max_probs.ge(T_sp)

    # compute loss
    L_ce = CE(x_logit1, y, mask)
    L_pt = CE(x_logit2, ps_label_pt, mask_pt)
    L_sp = CE(x_logit2, ps_label_sp, mask_sp)
    L_tot = L_ce + (1-w) * L_pt + w * L_sp
    L_tot.backward()
```

3.3. Adaptive pseudo-labeling

Although the framework above facilitates the model's robust learning subtly, we observe that there are still considerable mispredictions in the obtained pseudo-labels, especially at the early learning stage. One reason is that the previous training scheme is mainly constructed on the predictive consistency between each pair of single points, and the inter-point relation information is learned insufficiently. Therefore, we further exploit the super-point prior to introduce local structure information of point clouds for generating pseudo-labels of higher quality.

The super-points of a scene can be generated via an unsupervised low-level clustering by the position and color information of each point. We refer to [52] for the manner of super-point generation, and it is recommended for more details. Formally, given a point cloud $\mathbf{P} \in \mathbb{R}^{N\times D}$, we obtain a set of super-points $\{\mathbf{S}^{(i)}\}, i \in \langle M\rangle$, where M is the number of super-point and each $\mathbf{S}^{(i)} \in \mathbb{R}^{S^{(i)}\times D}$ includes $S^{(i)}$ D-dimension points. Each point in $\mathbf{P}$ belongs to one super-point only, so $\mathbf{S}^{(i)} \cap \mathbf{S}^{(j)} = \varnothing, \forall i \neq j$ and the summation of all $S^{(i)}$ is equal to N. The obtained super-point information is then used to improve the quality of point-wise pseudo-label $\widehat{\mathbf{Y}}^B$. Given point-wise predictions in each super-point group, a voting operation is carried out to get a "mainstream" category. The elected category is then propagated to all points in this group to obtain a super-point-wise pseudo-label $\widehat{\mathbf{Y}}^B_{\text{sp}}$. An illustrative example of $\widehat{\mathbf{Y}}^B$ and $\widehat{\mathbf{Y}}^B_{\text{sp}}$ is shown in Fig. 2(d) and (e), respectively. It can be observed that $\widehat{\mathbf{Y}}^B_{\text{sp}}$ tends to have higher purity. Similar to Section 3.2, we preserve confident predictions to form high-quality super-point-wise pseudo-labels. Specifically, given $\mathbf{Q}^B$, the average probability distribution in each super-point is computed first, of which the category with the highest score is selected and propagated in the whole super-point. Then the filtering mask $\mathbf{m}^{\text{sp}}$ is generated by checking whether the confidence of each point is beyond a pre-defined threshold τ^{sp}, similar to the computation in Eq. (3).

Although the voting operation enables $\widehat{\mathbf{Y}}^B_{\text{sp}}$ more stable and accurate, it suffers from the inherent noise arising from the super-point generation process. Thus, the point-wise pseudo-labels may have higher accuracy when the model is strong enough. Accordingly, we further design an adaptive combination mechanism to exploit the advantages of both. At the early stage, the learning of f_θ relies on $\widehat{\mathbf{Y}}^B_{\text{sp}}$ via a cross-entropy loss $\mathcal{L}_{pl}^{\text{sp}}$:

$$\mathcal{L}_{pl}^{\text{sp}} = \frac{1}{N}\sum_{i\in\langle N\rangle} \mathbf{m}_i^{\text{sp}} \cdot \text{cross-entropy}(\mathbf{Q}_i^B,\ \widehat{\mathbf{Y}}^B_{\text{sp}i}). \tag{6}$$

As the learning goes, an adaptive weight w is adopted to gradually incorporate $\mathcal{L}_{pl}$ (Eq. (4)) and abandon $\mathcal{L}_{pl}^{\text{sp}}$:

$$\mathcal{L}'_{pl} = w \cdot \mathcal{L}_{pl}^{\text{sp}} + (1 - w) \cdot \mathcal{L}_{pl}, \tag{7}$$

where w is a scalar in the range of [0, 1] and drops from 1 to 0 gradually with an inverse decay. Formally, the adaptive weight w at the kth training epoch can be computed as:

$$w = \alpha \cdot k^{-1}, k \in \mathbb{N}, \tag{8}$$

where $\alpha > 0$ indicates the decay ratio. In this way, at the late stage of training, f_θ can be completely supervised by the point-wise pseudo-label, so that the model can keep from the noise in super-point grouping. The new pseudo-labeling loss $\mathcal{L}'_{pl}$ is used to substitute the original $\mathcal{L}_{pl}$ in Eq. (5) for the final loss function.

4. Experiments

4.1. Experiment setup

Datasets and metric. We choose two popular point cloud datasets for the evaluation of our method, ScanNet-v2 [1] and S3DIS [16]. The ScanNet-v2 dataset [1] contains the 3D scans of 1613 indoor scenes of 20 semantic categories (1201 for training, 312 for validation, and 100 for online testing). The whole dataset includes around 243 million points in total. The S3DIS dataset [16] contains 271 room point clouds with 13 categories, scanned from 6 areas. Following the official train/validation split, Area 1,2,3,4,6 are used for training and Area 5 is used for evaluation. Besides, the S3DIS dataset has 273 million points, *i.e*, around 1 million points per scene on average, which is denser than scenes in the ScanNet dataset. Considering there are much more points in a scene of the S3DIS dataset [16], we randomly sample 800k points for those too-large scenes, while this number is 250k for the ScanNet-v2 dataset [1]. The evaluation metric for 3D semantic segmentation we use is intersection-over-union, and we report the mean result (MIoU) over all categories for comparison with other methods.

Table 1
MIoU (%) on the ScanNet-v2 dataset (online test set). * means the performance of our baseline on the fully-supervised setting. The underline indicates the previous SOTA performance on each setting. The supervision types "subcloud" and "segment" mean using subcloud-level and segment-level annotation, respectively. "20 points" and "1thing1click" mean annotating 20 points per scene and annotating one point in each instance, respectively.

Method	Supervision	MIoU
[18] PointNet++	100%	33.9
[53] SPLATNet	100%	39.3
[54] TangentConv	100%	43.8
[19] PointCNN	100%	45.8
[23] FPConv	100%	63.9
[20] PointConv	100%	66.6
[22] KPConv	100%	68.4
[25] MinkowskiNet	100%	73.6
[31] VMNet	100%	74.6
[55] Occuseg	100%	76.4
[56] Mix3D	100%	78.1
[24] SparseConv	100%	72.5*
[6] MPRM	subcloud	41.1
[7] SegGroup	segment	61.1
[2] SQN	0.01%	35.9
[2] SQN	0.1%	51.6
[4] OTOC	20 points	59.4
[4] OTOC	1thing1click	69.1
PointMatch	0.01%	**57.1**
PointMatch	0.1%	**68.8**
PointMatch	20 points	**62.4**
PointMatch	1thing1click	**69.5**

Table 2
MIoU (%) on the ScanNet-v2 dataset validation set. * means the performance of our baseline on the fully-supervised setting. Note that SQN [2] reports only its performance of 0.1% label setting on the ScanNet-v2 validation set.

Method	Supervision	MIoU
[24] SparseConv	100%	72.2*
[2] SQN	0.1%	53.5
[4] OTOC	20 points	61.4
[4] OTOC	1thing1click	70.5
PointMatch	0.01%	**58.7**
PointMatch	0.1%	**69.3**
PointMatch	20 points	**64.8**
PointMatch	1thing1click	**70.7**

Augmentations. We use two different groups of augmentations to create two views for each point cloud. The first group of augmentations includes random flipping and rotation, and the second group includes augmentations with slightly higher complexity, such as position jittering, affine transformation, and color jittering. The scale for jittering and affine transformation are 0.1 and 0.5, respectively. The color jittering may transform the original RGB values out of the standard range [0, 255], so we employ a clipping operation to restrict the jittered colors.

Implementation details. We adopt SparseConv [24] as the 3D U-Net backbone in PointMatch, which is also used in OTOC [4] and has a same-level performance compared with the backbone used in [57]. The output dimension of the SparseConv is set to 32, which is the same as in [4]. Following [4,57], we randomly sample 250k points for too-large scenes in the ScanNet-v2 dataset. Hyper-parameters in our experiment τ, τ^{sp}, ϵ, λ, and α are set to

Table 3
IoU(%) on the ScanNet-v2 online test set over 20 categories. "Super." means the supervision type.

Method	Super.	MIoU	*bath.*	*bed*	*book.*	*cab.*	*chair*	*count.*	*curt.*	*desk*	*door*	*floor*	*other.*	*pic.*	*refrig.*	*show.*	*sink*	*sofa*	*table*	*toil.*	*wall*	*wind.*
[18] PointNet++	100%	33.9	58.4	47.8	45.8	25.6	36.0	25.0	24.7	27.8	26.1	67.7	18.3	11.7	21.2	14.5	36.4	34.6	23.2	54.8	52.3	25.2
[19] PointCNN	100%	45.8	57.7	61.1	35.6	32.1	71.5	29.9	37.6	32.8	31.9	94.4	28.5	16.4	21.6	22.9	48.4	54.5	45.6	75.5	70.9	47.5
[20] PointConv	100%	55.6	63.6	64.0	57.4	47.2	73.9	43.0	43.3	41.8	44.5	94.4	37.2	18.5	46.4	57.5	54.0	63.9	50.5	82.7	76.2	51.5
[22] KPConv	100%	68.4	84.7	75.8	78.4	64.7	81.4	47.3	77.2	60.5	59.4	93.5	45.0	18.1	58.7	80.5	69.0	78.5	61.4	88.2	81.9	63.2
[24] SparseConv	100%	72.5	64.7	82.1	84.6	72.1	86.9	53.3	75.4	60.3	61.4	95.5	57.2	32.5	71.0	87.0	72.4	82.3	62.8	93.4	86.5	68.3
[25] Mink.Net	100%	73.6	85.9	81.8	83.2	70.9	84.0	52.1	85.3	66.0	64.3	95.1	54.4	28.6	73.1	89.3	67.5	77.2	68.3	87.4	85.2	72.7
[31] VMNet	100%	74.6	87.0	83.8	85.8	72.9	85.0	50.1	87.4	58.7	65.8	95.6	56.4	29.9	76.5	90.2	71.6	81.2	63.1	93.9	85.8	70.9
[55] OccuSeg	100%	76.4	75.8	79.6	83.9	74.6	90.7	56.2	85.0	68.0	67.2	97.8	61.0	33.5	77.7	81.9	84.7	83.0	69.1	97.2	88.5	72.7
[56] Mix3D	100%	78.1	96.4	85.5	84.3	78.1	85.8	57.5	83.1	68.5	71.4	97.9	59.4	31.0	80.1	89.2	84.1	81.9	72.3	94.0	88.7	72.5
[2] SQN	0.1%	51.6	44.2	68.3	58.7	47.2	75.5	30.7	47.9	48.9	33.3	93.0	29.6	32.7	27.0	42.3	38.7	68.3	54.0	76.2	71.1	44.7
[2] SQN	0.01%	35.9	35.5	59.0	53.6	21.4	62.8	25.8	40.4	34.0	19.9	91.8	24.2	14.5	01.5	16.6	09.4	53.4	36.7	33.3	58.1	25.8
PointMatch	0.1%	**68.8**	87.0	78.7	71.8	67.2	83.8	40.8	78.2	60.7	60.2	94.2	50.3	23.8	69.2	77.9	66.1	73.4	58.3	90.8	83.1	81.0
PointMatch	0.01%	**57.1**	81.5	77.8	60.1	51.7	78.6	28.4	60.3	52.9	48.6	91.4	39.5	08.3	49.2	50.3	37.1	68.4	46.5	74.5	79.9	57.8

0.95, 0.95, 0.5, 1.0, and 1.0, respectively. The network is trained for 512 epochs using Adam optimizer [58] with a learning rate of 0.01 and a mini-batch size of 8 on the ScanNet-v2 dataset and 4 for the S3DIS dataset. Considering the total number of training epochs, we replace the epoch number k in Eq. (8) with $\lfloor k/64 \rfloor$ on the "1thing1click" setting and $\lfloor k/32 \rfloor$ on others, which is the round-off of k divided by 32 or 64, in order to slow the decay rate. For the super-point generation, we follow [4] to use the mesh segment results [1] on the ScanNet-v2 dataset and the super-point graph partition manner proposed by [52] on the S3DIS dataset. Note that the super-points are used in training, and the inference stage does not rely on super-points. In terms of training strategy, a 0.01 learning rate is used with a step-decay at the epoch 384 to 0.001. All experiments are conducted on an Intel Xeon Gold 6226R CPU and an NVIDIA RTX3090 GPU with 24 GB memory.

Pseudo-code. To increase the reproducibility, we provide a pseudo-code of the core part of our PointMatch in Alg. 1. The complete version will be released upon acceptance.

4.2. Experiment results

Evaluation on ScanNet-v2. On the ScanNet-v2 [1] dataset, the evaluation of PointMatch is conducted on four weakly-supervised settings, *i.e*, 0.01% of points annotated in each scene [2], 0.1% of points annotated in each scene [2], 20 points annotated per scene [3] (20 points), and 1 point annotated for each instance in the scene [4] (1thing1click). The annotated points in the first two settings (0.01% and 0.1%) are randomly chosen following [2]. The "20 points" setting is implemented following the official ScanNet-v2 "3D Semantic label with Limited Annotations" benchmark [3]. Annotated points in the "1thing1click" setting are randomly chosen from each instance following [4]. The average point label in this setting is around 0.02% [4]. The evaluation results on the ScanNet-v2 online test set are presented in Table 1. Existing weakly supervised 3D semantic segmentation methods are also included for comparison, and some fully supervised methods are also listed in the table. As shown in the table, the proposed PointMatch consistently surpasses all existing methods over all weakly-supervised settings. It outperforms the state-of-the-art (SOTA) result by 21.2% on the 0.01% setting, by 17.2% on the 0.1% setting, and by 3.0% on the "20 points" setting. The performance on the "1thing1click" setting is further close to the fully-supervised baseline. Note that the work OTOC [4] takes 5 turns of iterative training to reach the above results, which is around 1536 epochs (3 times of ours). In addition, we also provide the performance of PointMatch on the ScanNet-v2 validation set in Table 2, on four weakly-supervised settings mentioned above, which also proves the superiority of PointMatch. Detailed results over 20 categories on the 0.1% and 0.01% settings are shown in Table 3.

Evaluation on s3DIS. We also evaluate the proposed method on the S3DIS [16] dataset to further validate the effectiveness of the proposed method. Three weakly-supervised settings are included for evaluation, *i.e*, 0.01%, 0.1%, and "1thing1click" (no official "20 points" setting provided for S3DIS). Note that the point cloud in the S3DIS dataset usually contains much more points than in the ScanNet-v2 dataset. By estimate, around 0.0036% of points are annotated in the "1thing1click" setting. The results on these three settings are listed in Table 5. The SOTA methods on both the fully-supervised and weakly-supervised settings are presented in the table for comparison. It is observed that the proposed PointMatch achieves the best performance over all three settings. It surpasses the SOTA result on the 0.01% setting by a large margin of 14.6%, by 5.2% on the "1thing1click" setting, and by 2.0% on the 0.1% setting. Impressively, our result on the 0.1% setting is very close to the fully-supervised baseline (63.4% v.s. 63.7%). The above results strongly prove the effectiveness and superiority of PointMatch, especially in the scenario of very sparse annotations (0.01%). Detailed results on all 13 categories on the 0.1% and 0.01% settings are listed in Table 4.

Qualitative results. Except for the quantitative results, we also exhibit some qualitative segmentation results of PointMatch. As shown in Fig. 3, we visualize each sample in two rows and six columns, namely the input point cloud (upper) and its super-point grouping (lower) in column (a), its globally-augmented (upper) and locally-augmented (lower) views in column (b), its point-wise (upper) and super-point-wise (lower) pseudo-label at the early and late stage of training in column (c) and (d), respectively, the prediction of PointMatch under the weak (upper) and full (lower) supervision in column (e), and the corresponding weak label (upper) and ground truth (lower) in column (f). Note that all results we visualize are generated under the "1thing1click" weak supervision. It is observed that the predictions of PointMatch under weak supervision are close to the ground truths and the fully-supervised predictions. More impressively, the super-point-wise pseudo-labels are superior to the point-wise ones at the early stage, while get inferior at the late stage of training (see red boxes in Fig. 3), which confirms our claim.

4.3. Ablation study

The proposed PointMatch mainly includes two components, the consistency training paradigm and the adaptive pseudo-labeling mechanism. Corresponding ablative experiments are conducted for the analysis of them.

Table 4
IoU(%) on the S3DIS dataset Area-5 over 13 categories. "Super." means the supervision type.

Method	Super.	MIoU	*ceil.*	*floor*	*wall*	*beam*	*col.*	*win.*	*door.*	*table*	*chair*	*sofa*	*book.*	*board.*	*clutter*
[18] PointNet++	100%	52.4	88.8	90.9	75.8	00.2	10.5	43.6	13.9	71.9	82.8	35.7	67.3	51.6	47.8
[19] PointCNN	100%	57.3	92.3	98.2	79.4	00.0	17.6	22.8	62.1	74.4	80.6	31.7	66.7	62.1	56.7
[52] SPGraph	100%	58.0	89.4	96.9	78.1	00.0	42.8	48.9	61.6	84.7	75.4	69.8	52.6	02.1	52.2
[23] FPConv	100%	62.8	94.6	98.5	80.9	00.0	19.1	60.1	48.9	80.6	88.0	53.2	68.4	68.2	54.9
[22] KPConv	100%	67.1	92.8	97.3	82.4	00.0	23.9	58.0	69.0	91.0	81.5	75.3	75.4	66.7	58.9
[32] PointTransformer	100%	70.4	94.0	98.5	86.3	00.0	38.0	63.4	74.3	89.1	82.4	74.3	80.2	76.0	59.3
[59] Π Model	10%	46.3	91.8	97.1	73.8	00.0	5.1	42.0	19.6	67.2	66.7	47.9	19.1	30.6	41.3
[60] MT	10%	47.9	92.2	96.8	74.1	00.0	10.4	46.2	17.7	70.7	67.0	50.2	24.4	30.7	42.2
[8] DGCNN+CRF	10%	48.0	90.9	97.3	74.8	00.0	08.4	49.3	27.3	71.7	69.0	53.2	16.5	23.3	42.8
[59] Π Model	0.2%	44.3	89.1	97.0	71.5	00.0	03.6	43.2	27.4	63.1	62.1	43.7	14.7	24.0	36.7
[60] MT	0.2%	44.4	88.9	96.8	70.1	00.1	03.0	44.3	28.8	63.7	63.6	43.7	15.5	23.0	35.8
[8] DGCNN+CRF	0.2%	44.5	90.1	97.1	71.9	00.0	01.9	47.2	29.3	64.0	62.9	42.2	15.9	18.9	37.5
[2] SQN	0.1%	61.4	91.7	95.6	78.7	00.0	24.2	55.9	63.1	70.5	83.1	60.7	67.8	56.1	50.6
[2] SQN	0.01%	45.3	89.2	93.5	71.3	00.0	04.1	34.7	41.0	54.9	66.9	25.7	55.4	12.8	39.6
PointMatch	0.1%	**63.4**	92.8	97.4	81.7	00.0	29.3	46.9	73.8	76.7	87.2	70.7	50.8	63.0	53.7
PointMatch	0.01%	**59.9**	90.7	97.1	80.4	00.0	15.2	51.2	62.1	72.7	83.7	68.1	43.9	67.1	46.7

Table 5
MIoU (%) on the S3DIS dataset (Area-5 for validation). * means the performance of our fully-supervised baseline. The underline indicates the previous SOTA performance on each setting.

Method	Supervision	MIoU
[17] PointNet	100%	41.1
[61] SegCloud	100%	48.9
[54] TangentConv	100%	52.8
[19] PointCNN	100%	57.3
[52] SPGraph	100%	58.0
[25] MinkowskiNet	100%	65.4
[22] KPConv	100%	67.1
[32] PointTransformer	100%	70.4
[24] SparseConv	100%	63.7*
[59] Π Model	0.2%	44.3
[60] MT	0.2%	44.4
[8] DGCNN+CRF	0.2%	44.5
[59] Π Model	10%	46.3
[60] MT	10%	47.9
[8] DGCNN+CRF	10%	48.0
[4] OTOC	1thing1click	50.1
[2] SQN	0.01%	45.3
[2] SQN	0.1%	61.4
PointMatch	1thing1click	**55.3**
PointMatch	0.01%	**59.9**
PointMatch	0.1%	**63.4**

Table 6
Ablative results of consistency training in PointMatch. MIoU (%) on the ScanNet-v2 dataset validation set.

Method	Supervision	MIoU
Fully-Sup. Version	100%	72.2
PointMatch	0.01%	**58.7**
w/o Consist. Training	0.01%	51.3
PointMatch	0.1%	**69.3**
w/o Consist. Training	0.1%	67.3
PointMatch	20 points	**64.8**
w/o Consist. Training	20 points	55.0
PointMatch	1thing1click	**70.7**
w/o Consist. Training	1thing1click	62.2

Table 7
Ablative results of adaptive pseudo-labeling in PointMatch. MIoU (%) on the S3DIS dataset Area-5.

Method	Supervision	MIoU
Fully-Sup. Version	100%	63.7
PointMatch	0.01%	**59.9**
$w = 0$	0.01%	58.4
$w = 1$	0.01%	56.1
$w = 0.5$	0.01%	54.6
$k \leftarrow \lfloor k/16 \rfloor$	0.01%	58.7
PointMatch	1thing1click	**55.3**
$w = 0$	1thing1click	52.6
$w = 1$	1thing1click	50.2
$w = 0.5$	1thing1click	48.4
$k \leftarrow \lfloor k/32 \rfloor$	1thing1click	53.3

Consistency training. To validate the effectiveness of the consistency training, we remove one branch in our framework as well as the pseudo-labeling mechanism, so the resultant version is a SparseConv simply trained on the weak supervision (extended by super-point information as the original) with a cross-entropy loss. We implement ablative experiments on four weakly-supervised settings on the ScanNet-v2 validation set. As shown in Table 6, removing the consistency training results in noticeable performance drops consistently over all weakly-supervised settings, especially on the schemes with extremely little supervision, which strongly proves its great effectiveness.

Adaptive pseudo-labeling. The adaptive pseudo-labeling mechanism plays the role of pseudo-label correction at the early stage of training, and it is implemented with an inverse decay. To confirm the effectiveness of our design, we implement four versions on two weakly-supervised settings ("1thing1click" and "0.01%") for comparison: (i) using point-wise pseudo-label only ($w = 0$); (ii) using super-point-wise pseudo-label only ($w = 1$); (iii) using both two pseudo-labels but in a constant manner, by setting w to 0.5 ($w = 0.5$); (iv) using the adaptive mechanism with a larger decay ratio, by using $\lfloor k/32 \rfloor$ ("1thing1click" settings) and $\lfloor k/32 \rfloor$ (0.01% settings) in Eq. (8) ($k \leftarrow \lfloor k/16(32) \rfloor$). Results are listed in Table 7. Using either type of pseudo-label only is inferior to the adaptive combination, because both point-wise and super-point-wise pseudo-label have their own strengths. Using a constant weight also leads to a performance drop, which proves that giving temporally different reliance on the two pseudo-labels can better exploit their advantages. Besides, a faster decay of the weight w also results in a slightly worse result, which is usually close to the results of using point-wise pseudo labels only ($w = 0$). One reason is that the network is unable to learn adequate information from super-points when w drops too fast.

5. Conclusion and discussion

We propose a novel approach, PointMatch, which introduces a consistency training framework into weakly supervised semantic segmentation of point clouds. It works by enforcing the predictive

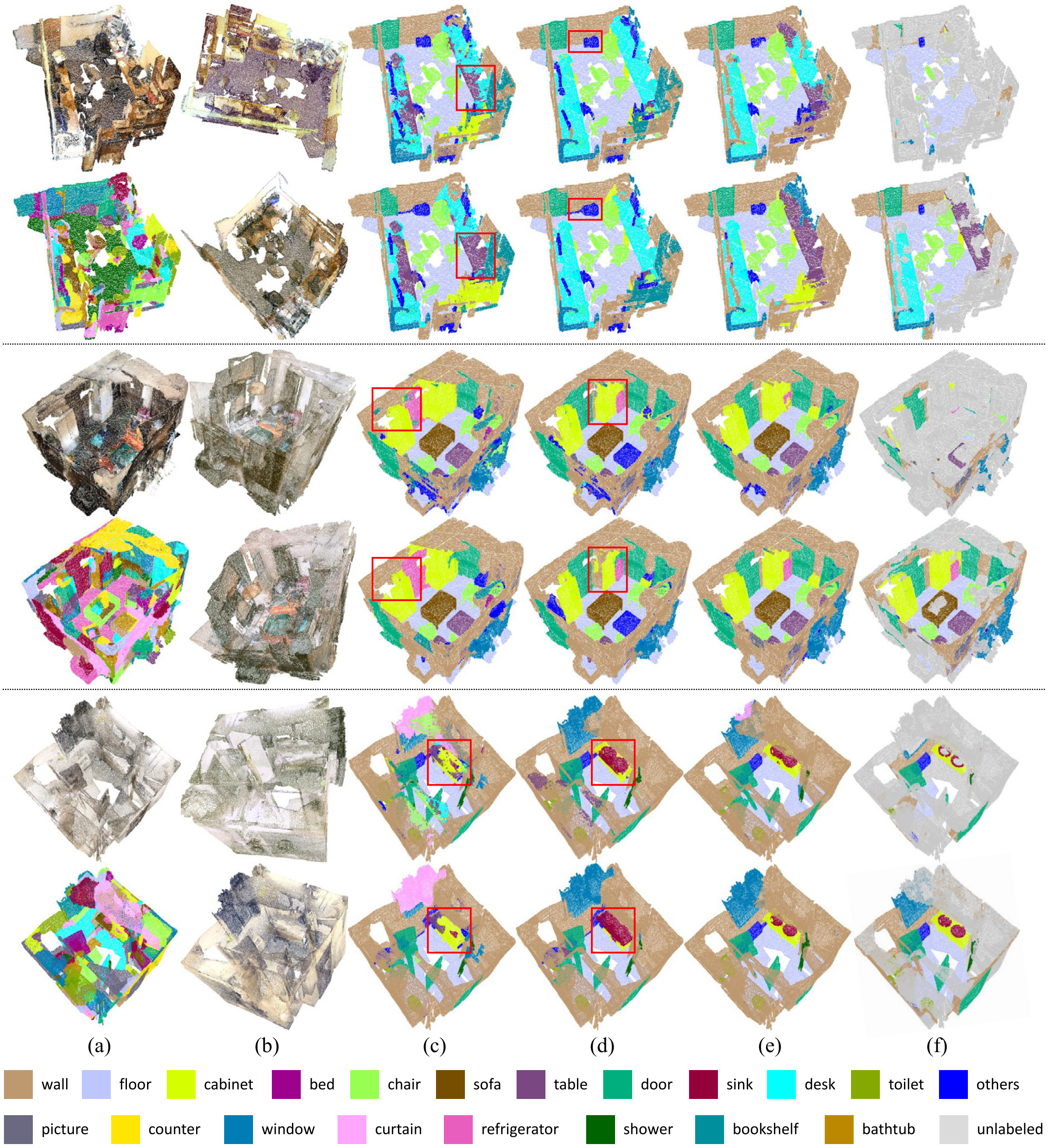

Fig. 3. Visualization of the qualitative results. We sample three scenes from the training set and their related results include, (a) upper: input point clouds, lower: the super-point grouping, in which colors do not indicate category information; (b): two views of the input point cloud; (c) upper: the point-wise pseudo-label at the early stage, lower: the super-point-level pseudo-label at the early stage; (d) upper: the point-wise pseudo-label at the late stage, lower: the super-point-level pseudo-label at the late stage; (e) upper: the weakly-supervised prediction, lower: the fully-supervised prediction; (f) upper: the weak supervision, lower: the full supervision (ground truth).

consistency between two views of a point cloud via pseudo-labeling, and enables the network to perform robust representation learning from weak label and data itself. The pseudo-label quality is further promoted by integrating super-point information in an adaptive manner. We use super-point clustering to extend the initial sparse supervision for the early stage of training, while in the later stage, we encourage more belief on the point-wise prediction by the model itself to introduce more precise pseudo supervision. Impressively, PointMatch achieves SOTA performance over various weakly-supervised semantic segmentation settings on both ScanNet-v2 and S3DIS datasets, and shows strong robustness given even extremely few labels, *e.g* 20 points per scene and 0.01% of points annotated.

As for the limitation of this approach, it relies on a considerably good super-point division to guarantee the label quality in the initial learning stage, so the performance may be greatly influenced in extremely noisy cases.

CRediT authorship contribution statement

Yushuang Wu: Contributed to the algorithm design and experiments, Wrote most of parts of the paper. **Zizheng Yan:** Contributed to the algorithm design and experiments, Wrote related works and part of methodology. **Shengcai Cai:** Conducted some parts of experiments. **Guanbin Li:** Suggested the key idea of this work. **Xiaoguang Han:** Proposed the research, Advised the project. **Shuguang Cui:** Proposed the research, Advised the project.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Data will be made available on request.

Acknowledgements

The work was supported in part by NSFC-62172348, the Basic Research Project No. HZQB-KCZYZ-2021067 of Hetao Shenzhen-HK S&T Cooperation Zone. It was also partially supported by NSFC-62172348, Outstanding Yound Fund of Guangdong Province with No. 2023B1515020055, the National Key R&D Program of China with grant No. 2018YFB1800800, by Shenzhen Outstanding Talents Training Fund 202002, by Guangdong Research Projects No. 2017ZT07X152 and No. 2019CX01X104, by the Guangdong Provincial Key Laboratory of Future Networks of Intelligence (Grant No. 2022B1212010001), and by Shenzhen Key Laboratory of Big Data and Artificial Intelligence (Grant No. ZDSYS201707251 409055).

References

[1] Dai A, Chang AX, Savva M, Halber M, Funkhouser T, Nießner M. Scannet: Richly-annotated 3d reconstructions of indoor scenes. In: 2017 proceedings of the IEEE conference on computer vision and pattern recognition. 2017, p. 5828–39.

[2] Hu Q, Yang B, Fang G, Guo Y, Leonardis A, Trigoni N, et al. Sqn: Weakly-supervised semantic segmentation of large-scale 3d point clouds. In: 2022 European conference on computer vision. Springer; 2022, p. 600–19.

[3] Hou J, Graham B, Nießner M, Xie S. Exploring data-efficient 3d scene understanding with contrastive scene contexts. In: 2021 proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2021, p. 15587–97.

[4] Liu Z, Qi X, Fu C-W. One thing one click: A self-training approach for weakly supervised 3D semantic segmentation. In: 2021 proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2021, p. 1726–36.

[5] Wang H, Rong X, Yang L, Wang S, Tian Y. Towards weakly supervised semantic segmentation in 3D graph-structured point clouds of wild scenes. In: 2019 British machine vision conference. 2019, p. 284.

[6] 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: 2020 proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2020, p. 4384–93.

[7] Tao A, Duan Y, Wei Y, Lu J, Zhou J. Seggroup: Seg-level supervision for 3d instance and semantic segmentation. IEEE Trans Image Process 2022;31:4952–65.

[8] Xu X, Lee GH. Weakly supervised semantic point cloud segmentation: Towards 10x fewer labels. In: 2020 proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2020, p. 13706–15.

[9] Hu W, Miyato T, Tokui S, Matsumoto E, Sugiyama M. Learning discrete representations via information maximizing self-augmented training. In: 2017 proceedings of the international conference on machine learning. PMLR; 2017, p. 1558–67.

[10] Grill JB, Strub F, Altché F, Tallec C, Richemond P, Buchatskaya E, et al. Bootstrap your own latent-a new approach to self-supervised learning. Adv Neural Inf Process Syst 2020;33(1786):21271–84.

[11] Xie Q, Luong M-T, Hovy E, Le QV. Self-training with noisy student improves imagenet classification. In: 2020 proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2020, p. 10687–98.

[12] Sohn K, Berthelot D, Carlini N, Zhang Z, Zhang H, Raffel CA, et al. Fixmatch: Simplifying semi-supervised learning with consistency and confidence. Adv Neural Inf Process Syst 2020;33(51):596–608.

[13] French G, Mackiewicz M, Fisher M. Self-ensembling for visual domain adaptation. In: 2018 international conference on learning representations. 2018, p. 1–18.

[14] Shu R, Bui H, Narui H, Ermon S. A DIRT-T approach to unsupervised domain adaptation. In: 2018 international conference on learning representations. 2018, p. 1–19.

[15] Li J, Li G, Shi Y, Yu Y. Cross-domain adaptive clustering for semi-supervised domain adaptation. In: 2021 proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2021, p. 2505–14.

[16] Armeni I, Sener O, Zamir AR, Jiang H, Brilakis I, Fischer M, et al. 3D semantic parsing of large-scale indoor spaces. In: 2016 proceedings of the IEEE conference on computer vision and pattern recognition. 2016, p. 1534–43.

[17] Qi CR, Su H, Mo K, Guibas LJ. Pointnet: Deep learning on point sets for 3d classification and segmentation. In: 2017 proceedings of the IEEE conference on computer vision and pattern recognition. 2017, p. 652–60.

[18] Qi CR, Yi L, Su H, Guibas LJ. PointNet++: Deep hierarchical feature learning on point sets in a metric space. Adv Neural Inf Process Syst 2017;30:5105–14.

[19] Li Y, Bu R, Sun M, Wu W, Di X, Chen B. Pointcnn: Convolution on x-transformed points. Adv Neural Inf Process Syst 2018;31:828–38.

[20] Wu W, Qi Z, Fuxin L. Pointconv: Deep convolutional networks on 3d point clouds. In: 2019 proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2019, p. 9621–30.

[21] Wang Y, Sun Y, Liu Z, Sarma SE, Bronstein MM, Solomon JM. Dynamic graph CNN for learning on point clouds. ACM Trans Graph 2019;38(5):1–12.

[22] Thomas H, Qi CR, Deschaud J-E, Marcotegui B, Goulette F, Guibas LJ. Kpconv: Flexible and deformable convolution for point clouds. In: 2019 IEEE/CVF international conference on computer vision. 2019, p. 6411–20.

[23] Lin Y, Yan Z, Huang H, Du D, Liu L, Cui S, et al. Fpconv: Learning local flattening for point convolution. In: 2020 proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2020, p. 4293–302.

[24] Graham B, Engelcke M, Van Der Maaten L. 3D semantic segmentation with submanifold sparse convolutional networks. In: 2018 proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2018, p. 9224–32.

[25] Choy C, Gwak J, Savarese S. 4D spatio-temporal convnets: Minkowski convolutional neural networks. In: 2019 proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2019, p. 3075–84.

[26] Wang Z, Lu F. VoxSegNet: Volumetric CNNs for semantic part segmentation of 3D shapes. IEEE Trans Vis Comput Graphics 2020;26(9):2919–30.

[27] Huang SS, Ma ZY, Mu TJ, Fu H, Hu SM. Supervoxel convolution for online 3d semantic segmentation. ACM Trans Graph 2021;40(3):1–15.

[28] Jiang L, Zhao H, Liu S, Shen X, Fu C-W, Jia J. Hierarchical point-edge interaction network for point cloud semantic segmentation. In: 2019 IEEE/CVF international conference on computer vision. 2019, p. 10432–40.

[29] Lei H, Akhtar N, Mian A. Spherical kernel for efficient graph convolution on 3d point clouds. IEEE Trans Pattern Anal Mach Intell 2021;43(10):3664–80.

[30] Schult J, Engelmann F, Kontogianni T, Leibe B. Dualconvmesh-net: Joint geodesic and euclidean convolutions on 3d meshes. In: 2020 proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2020, p. 8612–22.

[31] Hu Z, Bai X, Shang J, Zhang R, Dong J, Wang X, et al. Vmnet: Voxel-mesh network for geodesic-aware 3D semantic segmentation. In: 2021 IEEE/CVF international conference on computer vision. 2021, p. 15488–98.

[32] Zhao H, Jiang L, Jia J, Torr PH, Koltun V. Point transformer. In: 2021 IEEE/CVF international conference on computer vision. 2021, p. 16259–68.

[33] Guo MH, Cai JX, Liu ZN, Mu TJ, Martin RR, Hu SM. PCT: Point cloud transformer. Comput Vis Media 2021;7(2):187–99.

[34] Vanian V, Zamanakos G, Pratikakis I. Improving performance of deep learning models for 3D point cloud semantic segmentation via attention mechanisms. Comput Graph 2022;106:277–87.

[35] Gong J, Ye Z, Ma L. Neighborhood co-occurrence modeling in 3D point cloud segmentation. Comput Vis Media 2022;8(2):303–15.

[36] Zhao Y, Ma X, Hu B, Zhang Q, Ye M, Zhou G. A large-scale point cloud semantic segmentation network via local dual features and global correlations. Comput Graph 2023;111:133–44.

[37] Shi X, Xu X, Chen K, Cai L, Foo CS, Jia K. Label-efficient point cloud semantic segmentation: An active learning approach, CoRR. 2021, p. 1–16, URL: https://arxiv.org/abs/2101.06931.

[38] Zhou B, Khosla A, Lapedriza A, Oliva A, Torralba A. Learning deep features for discriminative localization. In: 2016 proceedings of the IEEE conference on computer vision and pattern recognition. 2016, p. 2921–9.

[39] Sharma C, Kaul M. Self-supervised few-shot learning on point clouds. Adv Neural Inf Process Syst 2020;33(605):7212–21.

[40] Liu Y, Yi L, Zhang S, Fan Q, Funkhouser TA, Dong H. P4contrast: Contrastive learning with pairs of point-pixel pairs for RGB-D scene understanding, CoRR. 2020, p. 1–12, URL: https://arxiv.org/abs/2012.13089.

[41] Xie S, Gu J, Guo D, Qi CR, Guibas L, Litany O. Pointcontrast: Unsupervised pre-training for 3d point cloud understanding. In: 2020 European conference on computer vision. Springer; 2020, p. 574–91.

[42] Zhang Z, Girdhar R, Joulin A, Misra I. Self-supervised pretraining of 3d features on any point-cloud. In: 2021 IEEE/CVF international conference on computer vision. 2021, p. 10252–63.

[43] Li M, Xie Y, Shen Y, Ke B, Qiao R, Ren B, et al. Hybridcr: Weakly-supervised 3d point cloud semantic segmentation via hybrid contrastive regularization. In: 2022 proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2022, p. 14930–9.

[44] Deng S, Dong Q, Liu B, Hu Z. Superpoint-guided semi-supervised semantic segmentation of 3D point clouds. In: 2022 international conference on robotics and automation. IEEE; 2022, p. 9214–20.

[45] Wei C, Shen K, Chen Y, Ma T. Theoretical analysis of self-training with deep networks on unlabeled data. In: 2020 international conference on learning representations. 2020, p. 1–30.

[46] Miyato T, Maeda S-i, Koyama M, Ishii S. Virtual adversarial training: a regularization method for supervised and semi-supervised learning. IEEE Trans Pattern Anal Mach Intell 2019;41(8):1979–93.

[47] Berthelot D, Carlini N, Goodfellow I, Papernot N, Oliver A, Raffel CA. MixMatch: A holistic approach to semi-supervised learning. Adv Neural Inf Process Syst 2019;32(454):5049–59.

[48] Berthelot D, Carlini N, Cubuk ED, Kurakin A, Sohn K, Zhang H, et al. ReMixMatch: Semi-supervised learning with distribution matching and augmentation anchoring. In: 2019 international conference on learning representations. 2019, p. 1–13.

[49] Li K, Liu C, Zhao H, Zhang Y, Fu Y. Ecacl: A holistic framework for semi-supervised domain adaptation. In: 2021 IEEE/CVF international conference on computer vision. 2021, p. 8558–67.

[50] Melas-Kyriazi L, Manrai AK. PixMatch: Unsupervised domain adaptation via pixelwise consistency training. In: 2021 proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2021, p. 12435–45.

[51] Wang H, Cong Y, Litany O, Gao Y, Guibas LJ. 3DIoUMatch: Leveraging iou prediction for semi-supervised 3d object detection. In: 2021 proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2021, p. 14615–24.

[52] Landrieu L, Simonovsky M. Large-scale point cloud semantic segmentation with superpoint graphs. In: 2018 proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2018, p. 4558–67.

[53] Su H, Jampani V, Sun D, Maji S, Kalogerakis E, Yang MH, et al. Splatnet: Sparse lattice networks for point cloud processing. In: 2018 proceedings of the IEEE conference on computer vision and pattern recognition. 2018, p. 2530–9.

[54] Tatarchenko M, Park J, Koltun V, Zhou QY. Tangent convolutions for dense prediction in 3d. In: 2018 proceedings of the IEEE conference on computer vision and pattern recognition. 2018, p. 3887–96.

[55] Han L, Zheng T, Xu L, Fang L. Occuseg: Occupancy-aware 3d instance segmentation. In: 2020 proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2020, p. 2940–9.

[56] Nekrasov A, Schult J, Litany O, Leibe B, Engelmann F. Mix3d: Out-of-context data augmentation for 3d scenes. In: 2021 international conference on 3D vision. IEEE; 2021, p. 116–25.

[57] Jiang L, Zhao H, Shi S, Liu S, Fu C-W, Jia J. Pointgroup: Dual-set point grouping for 3d instance segmentation. In: 2020 proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2020, p. 4867–76.

[58] Kingma DP, Ba J. Adam: A method for stochastic optimization. In: 2015 international conference on learning representations. 2015, p. 1–15.

[59] Laine S, Aila T. Temporal ensembling for semi-supervised learning. In: 2017 international conference on learning representations. 2017, p. 1–13.

[60] Tarvainen A, Valpola H. Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results. Adv Neural Inf Process Syst 2017;30:1195–204.

[61] Tchapmi L, Choy C, Armeni I, Gwak J, Savarese S. Segcloud: Semantic segmentation of 3d point clouds. In: 2017 international conference on 3D vision. IEEE; 2017, p. 537–47.