WeakPCSOD: Overcoming the Bias of Box Annotations for Weakly Supervised Point Cloud Salient Object Detection

Jun Wei[1,2], **S. Kevin Zhou**[3,4], **Shuguang Cui**[2,1], **Zhen Li**[2,1]*
[1] FNii, CUHK-Shenzhen, Shenzhen, China
[2] SSE, CUHK-Shenzhen, Shenzhen, China
[3] School of Biomedical Engineering & Suzhou Institute for Advanced Research, University of Science and Technology of China, Suzhou, China
[4] Institute of Computing Technology, Chinese Academy of Sciences, Beijing, China
junwei@link.cuhk.edu.cn, lizhen@cuhk.edu.cn

Abstract

Point cloud salient object detection (PCSOD) is a newly proposed task in 3D dense segmentation. However, the acquisition of accurate 3D dense annotations comes at a high cost, severely limiting the progress of PCSOD. To address this issue, we propose the first weakly supervised PCSOD (named **WeakPCSOD**) model, which relies solely on cheap 3D bounding box annotations. In WeakPCSOD, we extract noise-free supervision from coarse 3D bounding boxes while mitigating shape biases inherent in box annotations. To achieve this, we introduce a novel mask-to-box (M2B) transformation and a color consistency (CC) loss. The M2B transformation, from a shape perspective, disentangles predictions from labels, enabling the extraction of noiseless supervision from labels while preserving object shapes independently of the box bias. From an appearance perspective, we further introduce the CC loss to provide dense supervision, which mitigates the non-unique predictions stemming from weak supervision and substantially reduces prediction variability. Furthermore, we employ a self-training (ST) strategy to enhance performance by utilizing high-confidence pseudo labels. Notably, the M2B transformation, CC loss, and ST strategy are seamlessly integrated into any model and incur no computational costs for inference. Extensive experiments demonstrate the effectiveness of our WeakPCSOD model, even comparable to fully supervised models utilizing dense annotations.

Introduction

Point Cloud Salient Object Detection (PCSOD) is a 3D segmentation task newly proposed by Fan *et al.* (Fan, Gao, and Li 2022), which aims to segment the most attractive objects in point cloud scenarios. Compared with 2D salient object detection (Wei, Wang, and Huang 2020; Wei et al. 2020b), PCSOD can make full use of depth information, thus reducing the prediction ambiguity. Although formally similar to 3D semantic segmentation (3DSS), **PCSOD has its own unique characteristics**. First, PCSOD is a class-agnostic segmentation task and thus can be applied to a wide range of scenarios without being limited to the semantic classes of training data. Second, salient objects in point clouds depend on views, and PCSOD is to segment the salient objects of any given view in a 3D scene. Specifically, given a large 3D scene and an observer, salient objects are not constant but change with the position and perspective of the observer. Therefore, **PCSOD deals with the subspace of a scene noticed by the observer**, rather than the entire scene. It stimulates the attention mechanism of human eyes and avoids the saliency ambiguity of large cluttered scenes, which has great potential in VR/AR scenarios. For PCSOD data annotation, it is necessary to involve multiple professional annotators to jointly determine the salient object. An object is regarded as a positive label only if more than 80% annotators verify it. Otherwise, this sample will be ignored to avoid saliency conflict. For more comparison between PCSOD and 3DSS, and data annotation details, please refer to (Fan, Gao, and Li 2022). Due to the class-agnostic feature, PCSOD can provide the pre-processing results for many downstream tasks, such as 3D shape classification, compression, and quality assessment. Specifically, with the popularity of mobile phones and virtual reality devices, PCSOD can capture what the user watches at any time, and can also be exploited to build a 3D avatar for live stream or display. Therefore, PCSOD has a wide range of application prospects.

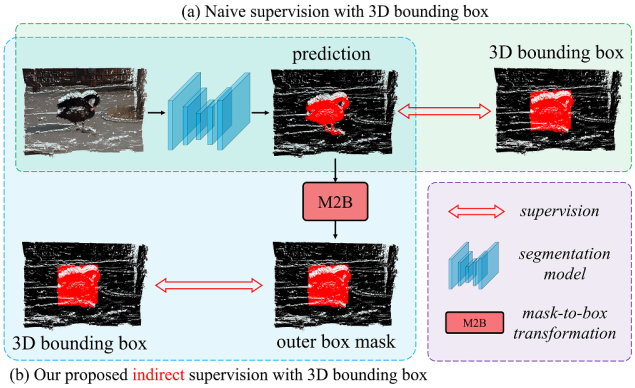

Figure 1: (a) shows the naive direct supervision with 3D bounding box annotations for point cloud salient object detection (PCSOD). (b) is our proposed WeakPCSOD model, which conducts indirect supervision to overcome the box annotation bias through a mask-to-box (M2B) transformation.

*Corresponding author: Zhen Li

Although promising, PCSOD requires point-level annotations, which is time-consuming and laborious. For example, it takes about 22.3 minutes to label a scene in ScanNet (Dai et al. 2017). The high labeling cost severely limits the size of the dataset. To alleviate the model's hunger for dense annotations, a few works (Cheng et al. 2021; Hou et al. 2021; Jiang et al. 2021; Liu, Qi, and Fu 2021; Zhang et al. 2021) have explored the weakly supervised point cloud analysis with sparse supervision, where only a few points are labeled. However, the selection of points is subjective and the sparse labels can not precisely limit the scope of the object. Therefore, we contribute the first work (named **WeakPCSOD**) to address the weakly supervised PCSOD with 3D bounding boxes, which only requires two diagonal points, *i.e.*, $(x_{min}, y_{min}, z_{min})$ and $(x_{max}, y_{max}, z_{max})$, to describe the scope of the object. Compared with full supervision, our WeakPCSOD greatly reduces the labeling cost.

Despite attractive, how to train the PCSOD model with only 3D bounding boxes is a big challenge. One naive way is to take all points inside the bounding box as salient points to train a segmentation model, as shown in Fig. 1(a). However, this direct supervision will mislead the model with a strong box bias. This bias forces the model to predict box-shaped salient objects, failing to preserve their original shapes. To overcome the box bias of annotations, for the first time, we propose the **indirect supervision**, as shown in Fig. 1(b). Instead of supervising the prediction itself, we supervise its outer box mask, which is achieved by the **mask-to-box (M2B) transformation**. This design separates the prediction from annotation, thus avoiding the misleading of box bias. As a result, the prediction can preserve the shape of salient objects well. Meanwhile, the annotation can indirectly supervise the location and scope of objects in the prediction through the outer box mask. Notably, M2B is our main contribution, which is differentiable and can be involved in gradient back-propagation during training.

However, this indirect supervision inevitably leads to non-unique predictions, since one outer box mask can correspond to multiple predictions. To alleviate the ambiguity, we propose a dense supervision based on color cues. Specifically, any two points with similar colors are assumed to have the same label. Given this, we design the **color consistency (CC) loss** to explicitly reduce the feature distance between points with the same labels. Surely, this supervision may contain errors, so only point pairs with high color similarity are involved in CC loss. Experiment results show that CC loss largely reduces the non-uniqueness of predictions. Note that, CC loss requires no annotations at all, and in CC loss, the color of the points is not used as an input feature but to generate constraints.

Taking above components together, we achieve the WeakPCSOD model. The superiority of WeakPCSOD is obvious in real applications: **(1) Low cost.** WeakPCSOD only requires cheap bounding box annotations for segmentation, rather than expensive masks. **(2) Efficiency.** The components of WeakPCSOD are only involved in the training phase, incurring no computational cost to inference. **(3) Universality.** WeakPCSOD redesigns the supervision loss and training strategy without any network modification, thus it is plug-and-play to any model. **(4) High Performance.** Although simple, WeakPCSOD surprisingly achieves high-quality salient object masks, even comparable to fully supervised models. In summary, our contributions are three-fold:

- We contribute the first work to weakly supervised PCSOD using bounding box annotations, which greatly reduces labeling cost and achieves comparable performance to full supervision.
- We overcome the bias of bounding box annotations through the proposed mask-to-box transformation and color consistency loss, which are independent of specific networks to achieve noise-free supervision.
- Extensive experiments on different backbone networks have confirmed the effectiveness of our proposed WeakPCSOD model.

Related Works

Salient Object Detection. Salient Object Detection (SOD) aims to capture the most attractive targets in the scene. For 2D images, many SOD works (Chen et al. 2020; Pang et al. 2020) have been proposed. However, most of these works (Tian et al. 2021) are limited to RGB images and ignore 3D spatial geometry of objects. Therefore, researchers extend SOD to 3D scenarios, including RGBD images (Fan et al. 2020) and point clouds (Fan, Gao, and Li 2022). For RGBD images, (Zhang et al. 2020b) adopts a two-stream architecture to process RGB features and depth features separately. Differently, (Li et al. 2020) proposes the cross-modality modulation for better feature selection (Wang et al. 2021). For point clouds, previous works (Zheng et al. 2019) mainly aim to predict the distribution heatmap of human attention, failing to segment complete salient objects. Given this, Fan *et al.* (Fan, Gao, and Li 2022) build the first SOD benchmark on a large challenging point cloud dataset. Here, we focus on PCSOD but with weak supervision. Compared with fully supervised methods, our proposed WeakPCSOD model greatly reduces labeling costs and is of practical value.

Point Cloud Processing. Different from regularly arranged RGB images, point clouds are disordered. Recently, PointNet (Charles et al. 2017) and PointNet++ (Qi et al. 2017) propose to process raw points directly and achieve surprising performance across multiple tasks. Following PointNet, PointCNN (Li et al. 2018) proposes to regularize the points with an X-transformation and then process them with typical convolutions. Differently, DGCNN (Wang et al. 2019) directly consumes raw points with an EdgeConv operation, which builds a dynamic graph to extract local geometric features, immune to the disorder. ShellNet (Zhang, Hua, and Yeung 2019) uses the statistics of concentric spherical shells to define representative features, thus allowing typical convolutions to be performed. (Goyal et al. 2021; Qian et al. 2022) show the impact of training strategies on model performance. However, all these methods are fully supervised, requiring expensive dense annotations. In contrast, our proposed WeakPCSOD model utilizes only 3D bounding boxes.

Weakly Supervised Learning. Due to the low cost of labeling, weakly supervised learning is gaining more and more

attention. For RGB images, category-based (Zhou et al. 2016), scribble-based (Zhang et al. 2020a), and bounding box-based (Hsu et al. 2019) methods are the most common paradigms. For point clouds, (Ren et al. 2021; Wei et al. 2020a) explore the category-based paradigm, which requires the minimum amount of labeling (*i.e.*, semantic categories). However, SOD is a class-agnostic task, where semantic categories are not available. Therefore, the category-based paradigm is not applicable. Recently, (Hou et al. 2021; Zhang et al. 2021) address the weakly supervised point cloud analysis with only a small number of point annotations. However, the selection of points for labeling is subjective and sparse labels can not precisely describe the scope of objects. Alternatively, the bounding box-based paradigm strikes a good balance between labeling cost and annotation accuracy, only requiring coordinates of two diagonal points but gives a tight scope of the object. Therefore, we adopt bounding box annotations for weakly supervised PCSOD. Surprisingly, by redesigning supervision loss and training strategies without modifying the network structures, our model achieves comparable performance to full supervision.

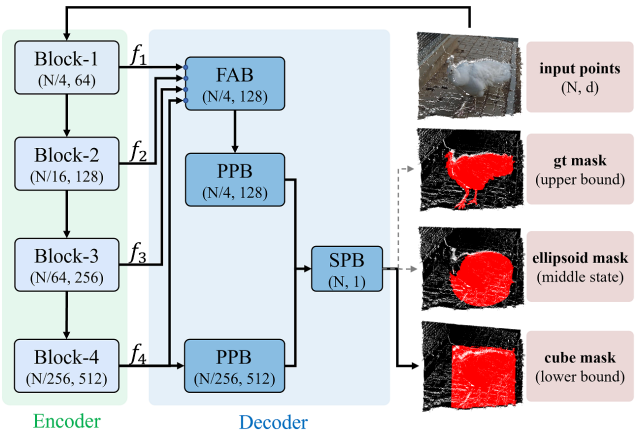

Figure 2: Visualization of the baseline models. Three types of labels are listed for comparison. Each baseline model is trained with one type of label. Feature Aggregation Block (FAB), Point Perception Block (PPB), and Saliency Perception Block (SPB) are proposed by (Fan, Gao, and Li 2022).

Proposed Methodology

Baseline Models

Before introducing the proposed methods, we discuss three baseline models. As shown in Fig. 2, each baseline model consists of three parts: encoder, decoder, and supervision. The only difference between these baseline models is the supervision (*i.e.*, three types of labels: gt mask, ellipsoid mask, and cube mask). The encoder and decoder remain the same because our design mainly focuses on the supervision loss.

Encoder. We adopt PointSOD (Fan, Gao, and Li 2022) as the baseline network, which is the state-of-the-art model in PCSOD. Its structure is shown in Fig. 2. PointSOD uses the classical PointNet++ (Qi et al. 2017) as the encoder. For input points with size (N, d), PointNet++ extracts four scales of features $\{f_i | i = 1, 2, 3, 4\}$ with size $(N/4^i, 2^{i+5})$. These features contain rich contextual information.

Decoder. To fuse the above features, PointSOD designs the Feature Aggregation Block (FAB), which first upsamples high-level features and low-level features to the same size and then concatenates them along the channel dimension. Besides, PointSOD proposes the Point Perception Block (PPB) to further abstract global semantics and strengthen the multi-scale representations. Given that the global semantics can supplement the multi-scale features and alleviate the distraction of non-salient background, a Saliency Perception Block (SPB) is introduced to integrate multi-scale features and global semantics to achieve the final predictions. Fig. 2 shows the pipeline. For the specific structures of FAB, PPB, and SPB, please refer to (Fan, Gao, and Li 2022).

Supervision. We design three baseline models with different types of labels, *i.e.*, one for full supervision and two for weak supervision. Following PointSOD (Fan, Gao, and Li 2022), the fully supervised model adopts dense labels (*i.e.*, ground-truth (gt) mask) for training, which is regarded as the performance upper bound of weakly supervised models. However, dense labels are costly and sometimes not available. Therefore, we propose to use cheap 3D bounding boxes to train the segmentation model. To achieve this, we convert bounding boxes into binary masks. One naive way is to generate an empty point cloud and set the label of points inside the bounding box to 1 and the rest to 0 (named cube mask). But the cube mask is box-biased since many background points are wrongly labeled, which misleads the model. To reduce the box bias, we further propose an ellipsoid mask, which is closer to the gt mask. Specifically, we take the shape $l \times w \times h$ and center point coordinate (x_c, y_c, z_c) of the bounding box to generate an object mask with the ellipsoid formula $\frac{(x-x_c)^2}{l^2} + \frac{(y-y_c)^2}{w^2} + \frac{(z-z_c)^2}{h^2} \leq 1$, where the label of points inside the ellipsoid is set to 1 and the rest to 0. By discarding points in the corners, ellipsoid masks contain less noise and lead to better generalization. Fig. 2 shows the masks generated by the above methods. For each baseline model, only one type of mask is involved. Binary cross entropy loss and Dice loss are used to supervise these models. We have compared these models quantitatively, where the gt mask leads to the best performance and the ellipsoid mask outperforms the cube mask.

Model Pipeline

Fig. 3 shows the pipeline of our proposed WeakPCSOD model, including the M2B transformation, CC loss, and self-training strategy. To make the pipeline clear, we divide it into two stages, using blue and green arrows to distinguish them. In stage 1, cube masks are adopted to train a weakly supervised segmentation model #1 with M2B transformation and CC loss. In stage 2, we use model #1 to predict pseudo masks for training samples and apply cube masks to filter out their background noise. Then, these refined pseudo masks are sent to train a fully supervised segmentation model #2. Note that model #1 and model #2 are based on the same backbone network. In inference, only model #2 is used to segment the salient objects, and model #1 is abandoned.

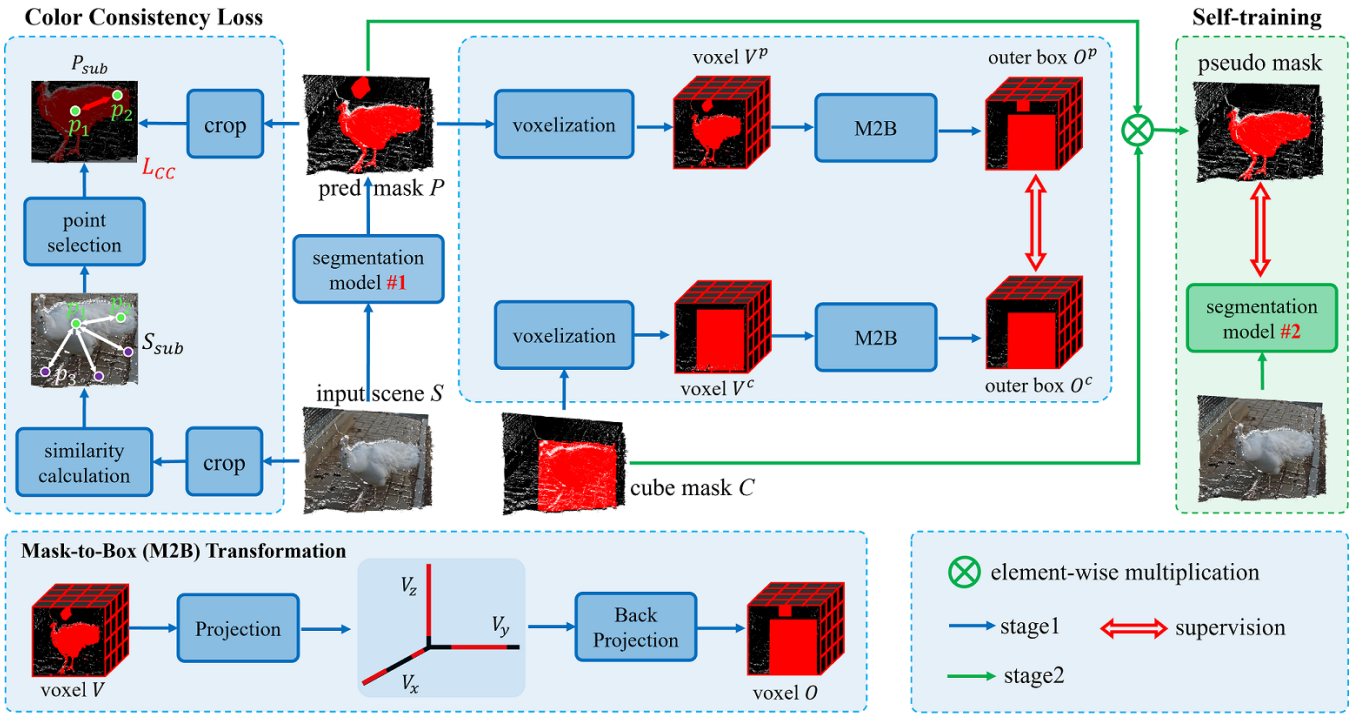

Figure 3: Pipeline of our proposed WeakPCSOD model which consists of two stages. In stage 1, as the blue arrow shows, we train a weakly supervised segmentation model #1 with mask-to-box (M2B) transformation and color consistency (CC) loss, using cube masks. Among them, M2B is used to construct the outer box mask of predictions, and CC loss is used to improve the prediction consistency between points (*i.e.*, p_1 and p_2) with high color similarity. In stage 2, as the green arrow shows, we train a separate fully supervised segmentation model #2 from scratch, using the pseudo masks generated by model #1. To clean the pseudo masks, we use cube masks to remove background false positives. In inference, only model #2 is adopted.

Mask to Box (M2B) Transformation

In Fig. 2, we convert bounding boxes into cube masks and ellipsoid masks. However, both of them can not avoid noisy labels, which is the inherent property of coarse annotations. Given this, we abandon the way of directly supervising the prediction and instead propose to supervise the outer box mask of the prediction indirectly. The outer box mask can decouple prediction from supervision, thus avoiding the bad effects of box bias. To achieve this, we design mask-to-box transformation, which is enabled by the following steps.

$$x_n = \left\lfloor \frac{xL}{x_{max}} \right\rfloor,\ y_n = \left\lfloor \frac{yH}{y_{max}} \right\rfloor,\ z_n = \left\lfloor \frac{zW}{z_{max}} \right\rfloor \quad (1)$$

Voxelization. Point cloud is disordered and requires a serial traversal to achieve the outer box mask for each prediction, which is quite slow. To accelerate this process, we voxelize the predicted mask into an ordered representation, as shown in Fig. 3. First, an empty tensor $V^p \in R^{L\times W\times H}$ is generated with all the elements set to 0. Then, for each point in the predicted mask P, we normalize its coordinates (x, y, z) into (x_n, y_n, z_n) by Eq. 1, where $x_{max}, y_{max}, z_{max}$ are the maximum values of the coordinates in P. Finally, we assign all values of P to V^p by $V^p[x_n, y_n, z_n] = P[x, y, z]$ in parallel. Similarly, we voxelize the cube mask C to V^c. Note that only the final predicted mask is voxelized, thus the memory footprint is acceptable.

Projection. After voxelization, we get two ordered tensors V^p and V^c. However, V^c is box-biased which can mislead V^p under direct supervision. To mitigate the mismatch, we construct the outer box mask O^p of V^p by projection and back-projection. In the projection, we reduce V^p to vectors by global max pooling along different axes, as shown in Eq. 2. After projection, the original 3D $V^p \in R^{L\times W\times H}$ is reduced into three 1D vectors: $V^p_x \in R^{L\times 1\times 1}$, $V^p_y \in R^{1\times W\times 1}$ and $V^p_z \in R^{1\times 1\times H}$.

$$\begin{aligned} V^p_x &= \max(V^p, \text{axis} = (1, 2)) \in R^{L\times 1\times 1} \\ V^p_y &= \max(V^p, \text{axis} = (0, 2)) \in R^{1\times W\times 1} \\ V^p_z &= \max(V^p, \text{axis} = (0, 1)) \in R^{1\times 1\times H} \end{aligned} \quad (2)$$

Back Projection. V^p_x, V^p_y and V^p_z no longer contain the shape information of objects, which are perfect for constructing outer box mask O^p, as shown in Eq. 3. Specifically, V^p_x, V^p_y, V^p_z are first repeated into $V^{p'}_x, V^{p'}_y, V^{p'}_z$ with the same size as V^P. Then, we element-wisely take the minimum of $V^{p'}_x, V^{p'}_y, V^{p'}_z$ to achieve O^p. Similarly, V^c is transformed into O^c. Note that, both projection and back-projection are differentiable and therefore can be involved

in gradient back-propagation.

$$\begin{aligned} V_x^{p'} &= \text{repeat}(V_x^p, \text{axis} = (1,2)) \in R^{L\times H\times W} \\ V_y^{p'} &= \text{repeat}(V_y^p, \text{axis} = (0,2)) \in R^{L\times H\times W} \\ V_z^{p'} &= \text{repeat}(V_z^p, \text{axis} = (0,1)) \in R^{L\times H\times W} \\ O^p &= \min(V_x^{p'}, V_y^{p'}, V_z^{p'}) \in R^{L\times H\times W} \end{aligned} \tag{3}$$

Supervision. Instead of using V^c to supervise V^p, we exploit O^c to supervise O^p with binary cross-entropy loss $\mathcal{L}_{BCE} = -[ylog(p) + (1-y)log(1-p)]$ and Dice loss $\mathcal{L}_{DICE} = 1 - \frac{2yp}{p+y}$, as shown in Eq. 4. This design decouples V^p from V^c, thus avoiding the misleading of box bias. As a result, V^p preserves the shape of objects. Meanwhile, V^c constrains the location and scope of objects in V^p through O^p, without making any assumptions about their shape.

$$\mathcal{L}_{Sum} = \mathcal{L}_{BCE}(O^p, O^c) + \mathcal{L}_{DICE}(O^p, O^c) \tag{4}$$

Color Consistency Loss

However, M2B transformation may lead to non-unique predictions, because multiple predictions V^p may correspond to the same outer box mask O^p. Therefore, we introduce the color consistency (CC) loss to disambiguate. By considering the color similarity between points, CC loss achieves a dense supervision through point cloud cropping, similarity calculation and consistency constraining. Without manual labeling, CC loss largely reduces the variability of predictions.

Cropping. Since CC loss focuses on points inside the bounding box, we first crop these points from input S and prediction P. Specifically, given two extreme points $(x_{\min}, y_{\min}, z_{\min})$ and $(x_{\max}, y_{\max}, z_{\max})$ of the bounding box, for each point (x, y, z) in S, we check whether the point satisfies Eq. 5. Those points that meet the condition will be separated into a sub-point cloud S_{sub}. Similarly, a sub-point cloud P_{sub} is separated from P.

$$x_{\min} \le x \le x_{\max}; y_{\min} \le y \le y_{\max}; z_{\min} \le z \le z_{\max} \tag{5}$$

Similarity Calculation. After cropping, we calculate the color similarity between points within S_{sub}. To reduce the computational cost, we adopt a random similarity calculation with $O(n)$ complexity rather than the pairwise similarity calculation with $O(n^2)$ complexity. Concretely, we first shuffle the points in S_{sub} to get S_{sub}^{shu}. Then, we pair the points that have the same position in S_{sub} and S_{sub}^{shu}, and calculate their color similarity with the cosine similarity formula $\rho = \frac{\boldsymbol{\alpha}\cdot\boldsymbol{\beta}}{||\boldsymbol{\alpha}||_2^2 \cdot ||\boldsymbol{\beta}||_2^2}$, where α and β are the RGB color vectors of the points in S_{sub} and S_{sub}^{shu}, respectively.

Supervision. Through the above calculation, we get the color similarity ρ between points. Empirically, points with highly similar colors usually belong to the same objects. As shown in Fig. 3, point p_1 and point p_2 belong to the same object and have a similar color, while p_1 and p_3 differ obviously. Of course, this assumption is not rigorous and would bring some noise. Therefore, we only select the point pairs with large ρ values (*i.e.*, $\rho > 0.9$) to calculate CC loss $\mathcal{L}_{CC}$, as shown in Eq. 6, where (i, j) is a pair of point indexes and N is the number of the selected pairs. Despite its simplicity, CC loss explicitly reduces the variability of predictions and surprisingly improves the robustness of the model. As shown in Eq. 7, taking $\mathcal{L}_{Sum}$ and $\mathcal{L}_{CC}$ together, we achieve the total weak supervision loss $\mathcal{L}_{total}^w$.

$$\mathcal{L}_{CC} = \frac{1}{N}\sum_{(i,j)} |P_{sub}^i - P_{sub}^j|, \quad \rho_{i,j} > 0.9 \tag{6}$$

$$\mathcal{L}_{total}^w = \mathcal{L}_{Sum} + \mathcal{L}_{CC} \tag{7}$$

Self-Training Strategy

Equipped with M2B transformation and CC loss, we achieve the weakly supervised PCSOD model #1. However, it still faces two challenges. First, as shown in Fig. 3, the predictions contain some false positives. Second, the learning paradigm is inconsistent between sparse weak supervision and dense full supervision. Following the paradigm of weakly supervised learning (Wei et al. 2021, 2022), we further introduce the self-training (ST) strategy to alleviate these problems. Specifically, we first generate pseudo masks P for training samples through model #1. To clean P, cube masks C are adopted to remove the background noise by element-wise multiplication (*i.e.*, $P \times C$). Finally, we adopt $P \times C$ as the pseudo masks to train a fully supervised segmentation model #2 from scratch, using traditional $\mathcal{L}_{BCE}$ and $\mathcal{L}_{DICE}$, as shown in Eq. 8.

$$\mathcal{L}_{total}^f = \mathcal{L}_{BCE} + \mathcal{L}_{DICE} \tag{8}$$

With pseudo label filtering and dense supervision, ST greatly boosts the segmentation performance. Note that, both model #1 and model #2 use the same network structure as the baseline model, but they are independent. During inference, only model #2 is used. Besides, pseudo labels are generated for the training set, where bounding boxes are available and the labels of the testing set are not leaked.

Model Summary

In general, WeakPCSOD is a two-stage pipeline. Combining M2B transformation, CC loss, and ST strategy, WeakPCSOD overcomes the box bias of bounding box annotations and realizes noise-free supervision. In the absence of dense labels, WeakPCSOD even achieves comparable performance to full supervision. Notably, the components of WeakPCSOD are independent of network structures. Therefore, WeakPCSOD is general and can be applied to other models. Besides, these components are only used during training, incurring no computational cost to inference.

Experiments and Results

Dataset and Backbone. To evaluate WeakPCSOD, we adopt the large-scale dataset proposed in PointSOD (Fan, Gao, and Li 2022) for experiments, which consists of 2,872 in-/out-door 3D scenes: 2000 for training and 872 for testing. Each scene contains 240,000 points, with dense and bounding box annotations. Note that, during training, only bounding box annotations are involved, and dense annotations are only used to quantify model performance. To verify the universality of WeakPCSOD, two classical networks are

Backbone	Supervision	Testing Set				Training Set			
		$F_m \uparrow$	$E_m \uparrow$	MAE↓	IoU↑	$F_m \uparrow$	$E_m \uparrow$	MAE↓	IoU↑
PointNet++	gt mask	0.791	0.870	0.064	0.683	0.883	0.947	0.029	0.806
	ellipsoid mask	0.653	0.789	0.105	0.512	0.686	0.813	0.091	0.546
	cube mask	0.614	0.747	0.139	0.526	0.627	0.758	0.129	0.544
	WeakPCSOD	0.780	0.869	0.068	0.649	0.836	0.912	0.053	0.718
PointMLP	gt mask	0.847	0.907	0.046	0.753	0.944	0.979	0.013	0.894
	ellipsoid mask	0.675	0.811	0.098	0.525	0.773	0.882	0.071	0.638
	cube mask	0.648	0.781	0.119	0.569	0.694	0.824	0.099	0.637
	WeakPCSOD	0.804	0.896	0.058	0.706	0.863	0.940	0.039	0.794

Table 1: Quantitative comparison between the baseline models and our WeakPCSOD. The gt mask row represents the performance upper bound. The WeakPCSOD row shows our model performance. ↑/↓ means larger/smaller is better.

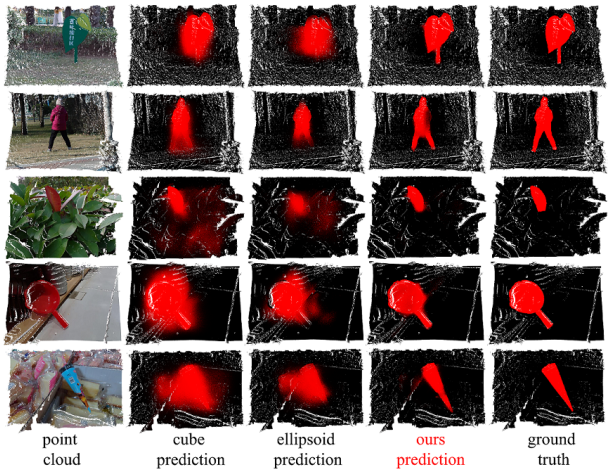

Figure 4: Prediction comparison between WeakPCSOD and weakly supervised models on PointNet++.

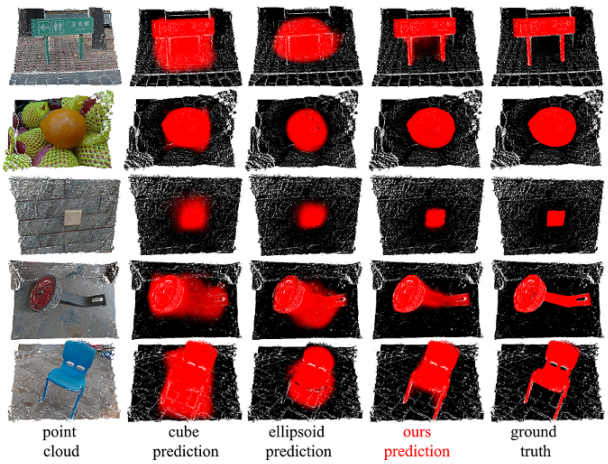

Figure 5: Prediction comparison between WeakPCSOD and weakly supervised models on PointMLP.

adopted as the backbone, including PointNet++ (Qi et al. 2017) and PointMLP (Ma et al. 2022).

Training Details. We use the AdamW optimizer for model training, where the initial learning rate and weight decay are set to 1e-3 and 1e-4, respectively. Cosine annealing is used to adjust the learning rate. To reduce memory footprint, we sample 4096 points from each point cloud during training. We train the model from scratch with Xavier initialization for 500 epochs, and the batch size is set to 64. Following (Fan, Gao, and Li 2022), random rotation is used to augment the input data. Codes will be released.

Evaluation Metrics. For fairness, we follow (Fan, Gao, and Li 2022) to adopt four popular evaluation metrics to quantify the model performance, including mean absolute error (MAE), F-measure (F_m) (Margolin, Zelnik-Manor, and Tal 2014), E-measure (E_m) (Fan et al. 2018), and intersection over union (IoU). MAE estimates the point-wise accuracy between the predicted segmentation map P and corresponding ground truth G, which is formulated as MAE= $\frac{1}{N}\sum_{i=1}^{N}|p_i - g_i|$, where $p_i \in P$ and $g_i \in G$. F_m is the harmonic mean value of the precision ($prec$) and recall ($reca$), *i.e.*, $F_m = \frac{(1-\beta^2)\cdot prec \cdot reca}{\beta^2 prec + reca}$, where β^2 is set to 0.3 for emphasizing the importance of precision. E_m captures both the local matching and region-level matching information of segmentation maps for assessment. IoU describes the extent of overlap between two segmentation maps, defined as IoU = $\frac{inter}{union}$, where $inter$ and $union$ indicate the intersection and union of two segmentation maps, respectively.

Quantitative Comparison. Tab. 1 quantitatively compares the performance between WeakPCSOD and the baseline models on PointNet++ and PointMLP. Among them, the performance on the testing set is the main measure for model generalization. We also list the performance on the training set because higher performance on the training set means a higher quality of pseudo masks, which are beneficial for the fully supervised model #2. In Tab. 1, the fully supervised model (*i.e.*, gt mask row) achieves the best performance which is regarded as the performance upper bound. In contrast, the model supervised by coarse cube masks achieves the worst performance which is regarded as the performance lower bound. Surprisingly, under the settings of PointNet++ backbone and testing set, our WeakPCSOD outperforms the model supervised by cube masks by **16.6%**, and ellipsoid

Model	Years	Sup.	$F_m \uparrow$	$E_m \uparrow$	MAE$\downarrow$	IoU$\uparrow$
PointNet	CVPR_{17}	$\mathcal{F}$	0.634	0.770	0.116	0.520
PointNet++	NIPS_{17}	$\mathcal{F}$	0.741	0.817	0.077	0.610
PointCNN	NIPS_{18}	$\mathcal{F}$	0.244	0.412	0.150	0.152
ShellNet	ICCV_{19}	$\mathcal{F}$	0.756	0.850	0.073	0.650
RandLA	PAMI_{21}	$\mathcal{F}$	0.635	0.742	0.127	0.519
PointSOD	ECCV_{22}	$\mathcal{F}$	0.772	0.853	0.068	0.658
BCM	CVPR_{19}	$\mathcal{W}$	0.653	0.786	0.104	0.552
BBTP	NIPS_{19}	$\mathcal{W}$	0.676	0.802	0.084	0.574
Box2Mask	ECCV_{22}	$\mathcal{W}$	0.721	0.812	0.079	0.604
WeakPCSOD-PointNet++		$\mathcal{W}$	0.780	0.869	0.068	0.649
WeakPCSOD-PointMLP		$\mathcal{W}$	**0.804**	**0.896**	**0.058**	**0.706**

Table 2: Performance comparison with fully-supervised ($\mathcal{F}$) and weakly-supervised ($\mathcal{W}$) models. Best scores are in bold.

Method	M2B	$\mathcal{L}_{CC}$	ST	$F_m \uparrow$	$E_m \uparrow$	MAE$\downarrow$	IoU$\uparrow$
cube mask				0.614	0.747	0.139	0.526
	✓			0.736	0.857	0.082	0.589
	✓	✓		0.748	0.860	0.079	0.603
WeakPCSOD	✓	✓	✓	**0.780**	**0.869**	**0.068**	**0.649**

Table 3: Ablation studies for M2B transformation, color consistency loss $\mathcal{L}_{CC}$ and self-training strategy (ST).

masks by **12.7%** on F_m metric, which largely closes the gap with the fully supervised model and requires no dense annotations. The overall performance ordering is **gt mask** $>$ **WeakPCSOD** $>$ **ellipsoid mask** $>$ **cube mask**.

Qualitative Comparison. Fig. 4 and Fig. 5 compare the predictions of WeakPCSOD with the weakly supervised models being PointNet++ and PointMLP, respectively. Intuitively, WeakPCSOD achieves the closest prediction to the ground truth and works well in keeping the object structure. As the 5^{th} line of Fig. 5 shows, the legs of the chair are precisely segmented. However, the models supervised by cube or ellipsoid masks are misled by the incorrect labels and exhibit a tendency to overfit to background points. Fig. 6 visualizes the predictions of WeakPCSOD and the fully supervised models, where the predictions of WeakPCSOD contain fewer false positives. This is because the fully supervised models are easily affected by wrong labels in dense annotations; but WeakPCSOD requires only coarse bounding boxes, containing less wrong labels.

Compared with Fully/Weakly Supervised Models. To demonstrate the effectiveness of WeakPCSOD, six fully supervised models and three weakly supervised models are adopted as the competitors, including PointNet (Charles et al. 2017), PointNet++ (Qi et al. 2017), PointCNN (Li et al. 2018), ShellNet (Zhang, Hua, and Yeung 2019), RandLA (Hu et al. 2021), PointSOD (Fan, Gao, and Li 2022), BCM (Song et al. 2019), BBTP (Hsu et al. 2019) and Box2Mask (Chibane et al. 2022). All these methods are evaluated on PCSOD testing set. As shown in Tab. 2, WeakPCSOD outperforms previous weakly supervised models by a large margin, even achieving comparable performance to the fully supervised models, which demonstrates the great potential of weakly supervised PCSOD.

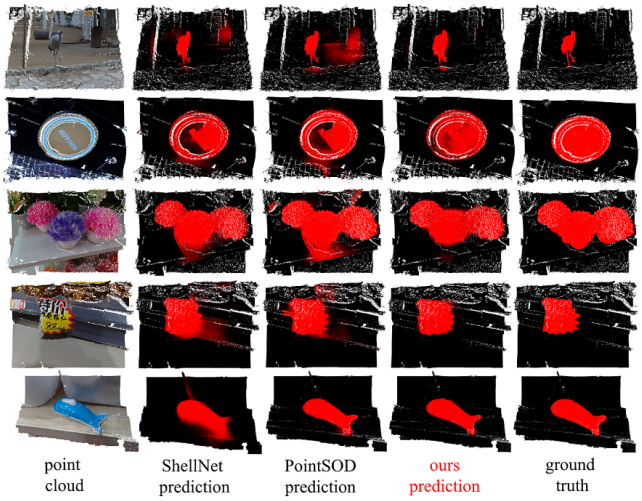

Figure 6: Prediction comparison between WeakPCSOD and fully supervised models with PointNet++ backbone.

Ablation Studies. As shown in Tab. 3, to evaluate the contribution of each module in WeakPCSOD, we conduct ablation studies on the testing set with the PointNet++ backbone. Specifically, with only M2B transformation, our model already improves the performance by **12.2%**, compared to cube supervision on the F_m metric. Furthermore, CC loss reduces the variability of predictions, leading to more robust training and improving F_m by **1.2%**. Besides, ST further cleans up the background noise in pseudo labels and aligns the forms of weak supervision and full supervision. As a result, it brings strong regularization and improves F_m by **3.2%**. Taking all these components together, our WeakPCSOD demonstrates superior performance in weakly supervised point cloud SOD.

Conclusion

To reduce the labeling cost of PCSOD, we contribute the first weakly supervised model (*i.e.*, WeakPCSOD), using only 3D bounding box annotations. However, these annotations have a strong box bias. To remove the bias, we design the mask-to-box transformation to separate predictions from labels. This indirect supervision preserves the object shape free from box bias. Besides, to reduce the variability of predictions, we introduce color consistency loss and self-training strategy. Notably, all these components are used only during training, incurring no computational cost to inference, and are independent of networks, allowing their portability to other models. We conduct extensive experiments with different networks to verify their effectiveness. Surprisingly, our WeakPCSOD model even achieves comparable performance to fully supervised ones, showing its potential for point cloud analysis.

Acknowledgments

This work was supported by NSFC with Grant No. 62293482, by the Basic Research Project No. HZQB-KCZYZ-2021067 of Hetao Shenzhen HK S&T Cooperation Zone, by Shenzhen General Program No. JCYJ20220530143600001, by Shenzhen-Hong Kong Joint Funding No. SGDX20211123112401002, by Shenzhen Outstanding Talents Training Fund, by Guangdong Research Project No. 2017ZT07X152 and No. 2019CX01X104, by the Guangdong Provincial Key Laboratory of Future Networks of Intelligence (Grant No. 2022B1212010001), by the Guangdong Provincial Key Laboratory of Big Data Computing, The Chinese University of Hong Kong, Shenzhen, by the NSFC 61931024&81922046&61902335, by the Shenzhen Key Laboratory of Big Data and Artificial Intelligence (Grant No. ZDSYS201707251409055), and the Key Area R&D Program of Guangdong Province with grant No. 2018B03033800, by Tencent&Huawei Open Fund.

References

Charles, R. Q.; Su, H.; Kaichun, M.; and Guibas, L. J. 2017. PointNet: Deep Learning on Point Sets for 3D Classification and Segmentation. In *CVPR*, 77–85.

Chen, Z.; Xu, Q.; Cong, R.; and Huang, Q. 2020. Global context-aware progressive aggregation network for salient object detection. In *AAAI*, 10599–10606.

Cheng, M.; Hui, L.; Xie, J.; and Yang, J. 2021. SSPC-Net: Semi-supervised Semantic 3D Point Cloud Segmentation Network. In *AAAI*.

Chibane, J.; Engelmann, F.; Anh Tran, T.; and Pons-Moll, G. 2022. Box2Mask: Weakly Supervised 3D Semantic Instance Segmentation using Bounding Boxes. In *ECCV*, 681–699.

Dai, A.; Chang, A. X.; Savva, M.; Halber, M.; Funkhouser, T.; and Nießner, M. 2017. ScanNet: Richly-annotated 3D Reconstructions of Indoor Scenes. In *CVPR*.

Fan, D.-P.; Gong, C.; Cao, Y.; Ren, B.; Cheng, M.-M.; and Borji, A. 2018. Enhanced-alignment measure for binary foreground map evaluation. *arXiv preprint arXiv:1805.10421*.

Fan, D.-P.; Zhai, Y.; Borji, A.; Yang, J.; and Shao, L. 2020. BBS-Net: RGB-D salient object detection with a bifurcated backbone strategy network. In *ECCV*, 275–292.

Fan, S.; Gao, W.; and Li, G. 2022. Salient Object Detection for Point Clouds. In *ECCV*.

Goyal, A.; Law, H.; Liu, B.; Newell, A.; and Deng, J. 2021. Revisiting point cloud shape classification with a simple and effective baseline. In *ICML*, 3809–3820.

Hou, J.; Graham, B.; Nießner, M.; and Xie, S. 2021. Exploring Data-efficient 3D Scene Understanding with Contrastive Scene Contexts. In *CVPR*.

Hsu, C.-C.; Hsu, K.-J.; Tsai, C.-C.; Lin, Y.-Y.; and Chuang, Y.-Y. 2019. Weakly supervised instance segmentation using the bounding box tightness prior. volume 32.

Hu, Q.; Yang, B.; Xie, L.; Rosa, S.; Guo, Y.; Wang, Z.; Trigoni, N.; and Markham, A. 2021. Learning Semantic Segmentation of Large-Scale Point Clouds with Random Sampling. In *PAMI*, 1–1.

Jiang, L.; Shi, S.; Tian, Z.; Lai, X.; Liu, S.; Fu, C.-W.; and Jia, J. 2021. Guided Point Contrastive Learning for Semi-supervised Point Cloud Semantic Segmentation. In *CVPR*.

Li, C.; Cong, R.; Piao, Y.; Xu, Q.; and Loy, C. C. 2020. RGB-D salient object detection with cross-modality modulation and selection. In *ECCV*, 225–241.

Li, Y.; Bu, R.; Sun, M.; Wu, W.; Di, X.; and Chen, B. 2018. Pointcnn: Convolution on x-transformed points. In *NIPS*.

Liu, Z.; Qi, X.; and Fu, C.-W. 2021. One Thing One Click: A Self-Training Approach for Weakly Supervised 3D Semantic Segmentation. In *CVPR*.

Ma, X.; Qin, C.; You, H.; Ran, H.; and Fu, Y. 2022. Rethinking network design and local geometry in point cloud: A simple residual MLP framework. *ICLR*.

Margolin, R.; Zelnik-Manor, L.; and Tal, A. 2014. How to evaluate foreground maps? In *CVPR*, 248–255.

Pang, Y.; Zhao, X.; Zhang, L.; and Lu, H. 2020. Multi-Scale Interactive Network for Salient Object Detection. In *CVPR*, 9410–9419.

Qi, C. R.; Yi, L.; Su, H.; and Guibas, L. J. 2017. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. In *NIPS*.

Qian, G.; Li, Y.; Peng, H.; Mai, J.; Hammoud, H. A. A. K.; Elhoseiny, M.; and Ghanem, B. 2022. PointNeXt: Revisiting PointNet++ with Improved Training and Scaling Strategies. *NIPS*.

Ren, Z.; Misra, I.; Schwing, A. G.; and Girdhar, R. 2021. 3d spatial recognition without spatially labeled 3d. In *CVPR*.

Song, C.; Huang, Y.; Ouyang, W.; and Wang, L. 2019. Box-driven class-wise region masking and filling rate guided loss for weakly supervised semantic segmentation. In *CVPR*, 3136–3145.

Tian, Z.; Shen, C.; Wang, X.; and Chen, H. 2021. BoxInst: High-Performance Instance Segmentation With Box Annotations. In *CVPR*, 5443–5452.

Wang, W.; Lai, Q.; Fu, H.; Shen, J.; Ling, H.; and Yang, R. 2021. Salient Object Detection in the Deep Learning Era: An In-depth Survey. *PAMI*, 1–1.

Wang, Y.; Sun, Y.; Liu, Z.; Sarma, S. E.; Bronstein, M. M.; and Solomon, J. M. 2019. Dynamic graph cnn for learning on point clouds. In *ACM Transactions On Graphics*, 1–12.

Wei, J.; Lin, G.; Yap, K.-H.; Hung, T.-Y.; and Xie, L. 2020a. Multi-path region mining for weakly supervised 3D semantic segmentation on point clouds. In *CVPR*.

Wei, J.; Wang, Q.; Li, Z.; Wang, S.; Zhou, S. K.; and Cui, S. 2021. Shallow feature matters for weakly supervised object localization. In *CVPR*, 5993–6001.

Wei, J.; Wang, S.; and Huang, Q. 2020. F^3Net: fusion, feedback and focus for salient object detection. In *AAAI*, 12321–12328.

Wei, J.; Wang, S.; Wu, Z.; Su, C.; Huang, Q.; and Tian, Q. 2020b. Label decoupling framework for salient object detection. In *CVPR*, 13025–13034.

Wei, J.; Wang, S.; Zhou, S.; Cui, S.; and Li, Z. 2022. Weakly Supervised Object Localization through Inter-class Feature Similarity and Intra-class Appearance Consistency. In *ECCV*.

Zhang, J.; Yu, X.; Li, A.; Song, P.; Liu, B.; and Dai, Y. 2020a. Weakly-supervised salient object detection via scribble annotations. In *CVPR*, 12546–12555.

Zhang, M.; Fei, S. X.; Liu, J.; Xu, S.; Piao, Y.; and Lu, H. 2020b. Asymmetric two-stream architecture for accurate rgb-d saliency detection. In *ECCV*, 374–390.

Zhang, Y.; Qu, Y.; Xie, Y.; Li, Z.; Zheng, S.; and Li, C. 2021. Perturbed Self-Distillation: Weakly Supervised Large-Scale Point Cloud Semantic Segmentation. In *ICCV*.

Zhang, Z.; Hua, B.-S.; and Yeung, S.-K. 2019. ShellNet: Efficient Point Cloud Convolutional Neural Networks Using Concentric Shells Statistics. In *ICCV*, 1607–1616.

Zheng, T.; Chen, C.; Yuan, J.; Li, B.; and Ren, K. 2019. PointCloud Saliency Maps. In *ICCV*, 1598–1606.

Zhou, B.; Khosla, A.; Lapedriza, A.; Oliva, A.; and Torralba, A. 2016. Learning deep features for discriminative localization. In *CVPR*, 2921–2929.