SS3D: Sparsely-Supervised 3D Object Detection from Point Cloud

Chuandong Liu[1,2] Chenqiang Gao[1,2 *] Fangcen Liu[1,2] Jiang Liu[3] Deyu Meng[4,5] Xinbo Gao[1]
[1]School of Communication and Information Engineering, Chongqing University of Posts and Telecommunications, Chongqing, China
[2]Chongqing Key Laboratory of Signal and Information Processing, Chongqing, China
[3]Meta, Menlo Park, USA
[4]Xi'an Jiaotong University, Xi'an, China
[5]Macau Institute of Systems Engineering, Macau University of Science and Technology, Taipa, Macau

Abstract

Conventional deep learning based methods for 3D object detection require a large amount of 3D bounding box annotations for training, which is expensive to obtain in practice. Sparsely annotated object detection, which can largely reduce the annotations, is very challenging since the missing-annotated instances would be regarded as the background during training. In this paper, we propose a sparsely-supervised 3D object detection method, named SS3D. Aiming to eliminate the negative supervision caused by the missing annotations, we design a missing-annotated instance mining module with strict filtering strategies to mine positive instances. In the meantime, we design a reliable background mining module and a point cloud filling data augmentation strategy to generate the confident data for iteratively learning with reliable supervision. The proposed SS3D is a general framework that can be used to learn any modern 3D object detector. Extensive experiments on the KITTI dataset reveal that on different 3D detectors, the proposed SS3D framework with only 20% annotations required can achieve on-par performance comparing to fully-supervised methods. Comparing with the state-of-the-art semi-supervised 3D objection detection on KITTI, our SS3D improves the benchmarks by significant margins under the same annotation workload. Moreover, our SS3D also outperforms the state-of-the-art weakly-supervised method by remarkable margins, highlighting its effectiveness.

1. Introduction

Three-dimensional (3D) object detection, aiming to localize and categorize objects from 3D sensor data (*e.g.*, LiDAR point cloud), has attracted increasing attention due to its diversified applications in autonomous driving, augmented/virtual reality, and indoor robotics. Recently, a number of approaches [1, 17, 18, 34, 35] based on either voxel-wise or point-wise features have been proposed and achieved high performance on large-scale benchmark datasets [2, 21]. However, most of the proposed 3D object detectors require fully supervised learning, implying a fully annotated dataset is required for the model learning. Compared to 2D image objects, annotating 3D point cloud objects is more labor-intensive: annotators have to switch viewpoints or zoom in and out throughout a 3D scene carefully for labeling each 3D object. Therefore, developing 3D detectors with on-par detection performance, while only requiring light-weighted object annotations, is a meaningful problem to tackle for practical applications.

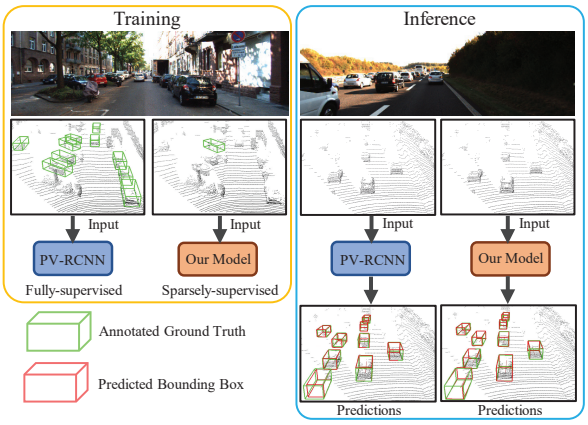

Figure 1. Demonstration of the required annotations of the fully-supervised method and our method. The left case shows the training stage of PV-RCNN [16] which is a high-performance detector with full annotations as input, while our model only annotates one instance for each scene. The right case shows the prediction results of PV-RCNN and our model, indicating that our model achieves comparable performance of the fully-supervised method.

*Corresponding author.

Recently, few works [10, 15, 24, 26, 33] have been proposed to address this issue. In [10], the weakly-supervised learning strategy was adopted. Specifically, the point annotation scheme was used to reduce the burden of annotating bounding boxes. However, the supervision information provided by the point annotation is weak, so that a certain amount of full annotations have to be provided additionally, in order to achieve optimal performance. In [24, 33], the semi-supervised learning strategy was used, where just part of the dataset was annotated with the rest unlabeled. The teacher-student framework was leveraged to transfer information from labeled data to unlabeled data. Nevertheless, the information transfer tends to be ineffective when the gap between labeled and unlabeled data is large. Besides, although just part of the dataset is annotated, it still takes non-negligible labour to label an individual scene, especially for crowded scenarios with many 3D objects, as shown as Fig. 1.

In this paper, we adopt the sparse annotation strategy and just annotate one 3D object in a scene, as illustrated in the left of the Fig. 1. In this way, we are capable of obtaining full supervision information of one 3D object for each scene. Intuitively, this facilitates the learning of information on unlabeled objects, since infra-scene information transfer is much easier than cross-scene knowledge transfer. However, sparsely annotated object detection also raises new challenges: missing-annotated instances will bring incorrect supervision signals (i.e., as negative samples) to disturb the training of the network. During training, due to that the missing-annotated instances and the region near those instances could be incorrectly marked as background, the weight updated of the network will be misguided significantly when gradients back-propagated. This challenge has been tackled in 2D sparse object detection methods [11, 27] by utilizing overlap or hierarchical relation information among 2D objects. However, such information may be absent in 3D datasets *e.g.*, in KITTI [2], which impedes directly applying such methods to 3D applications.

To address the challenge, we propose a novel and effective method for sparsely annotated 3D object detection, namely SS3D, which can be applied to any advanced 3D detector. The main idea of our SS3D is to iteratively mine positive instances and background with high confidence, and further use these generated data to train the 3D detector. We design two effective modules, namely missing-annotated instance mining module and reliable background mining module, to mine reliable missing positive instances and background, respectively. This ensures the 3D detector to be trained with confident supervision data. By this design, compared with the 3D detector trained with the fully annotated dataset, our SS3D can achieve comparable performance, where only 20% annotation is required for the sparsely annotated dataset.

To summarize, our contributions are as follows:

- We propose a novel method for sparsely annotated 3D object detection from point cloud which can be used as a general framework to train any existing 3D fully-supervised detector. To the best of our knowledge, this is the first work to explore the sparsely annotated strategy for the 3D object detection task.
- We design two effective modules to mine reliable missing positive instances and background, respectively, which ensures the 3D detector to be trained with confident supervision data.
- Experimental results show that our method with sparse annotations can achieve comparable performance with fully-supervised methods and highly outperforms state-of-the-art semi-supervised and weakly-supervised 3D object detection methods.

2. Related Work

2.1. Fully-Supervised 3D Object Detection

The existing 3D detection methods can be broadly categorized into two types: voxel-based methods [4,5,28,34,35] and point-based methods [12, 17, 19, 29, 30, 32].

For voxel-based methods, voxelization is a common measure for irregular point clouds to apply traditional 2D or 3D convolution. In voxelNet [36], voxel feature encoding layer was adopted for unified feature representation extraction from point cloud. SECOND [28] effectively extracted features from 3D voxels by modifying the sparse convolution algorithm [3, 7]. TANet [8] leveraged the stacked attention module to exploit the multi-level feature relation. Part-A^2 [18] proposed a two-stage network to explore the spatial relationship by grouping intra-object part features. SE-SSD [35] adopted a pair of teacher and student detectors to improve the performance without introducing extra computation in the inference. Voxel R-CNN [1] designed a voxel RoI pooling to directly aggregate spatial context from 3D voxel feature volumes.

Point-based methods directly take the raw irregular points as input to extract local and global features [13, 14]. PointRCNN [17] fused extracted features and raw points from 3D proposals generated in a bottom-up manner for refinement. STD [30] proposed a novel spherical anchor to reduce the number of anchors and exploited the sparse to dense idea to improve the performance. 3DSSD [29] proposed a fusion sampling strategy based on feature distance for rich information preservation. PV-RCNN [16] utilized voxel-to-keypoint scene encoding and keypoint-to-grid feature aggregation to improve the performance.

Although prior works have made significant progress and show impressive performance, such results deeply depend on the large-scale manual annotations, which are

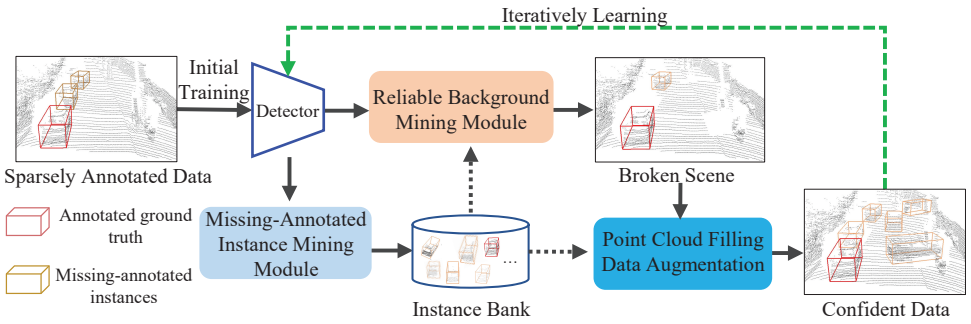

Figure 2. Our SS3D pipeline. The missing-annotated instance mining module searches the missing-annotated instances and stores them in the instance bank. The reliable background mining module leverages the instance bank to further obtain broken scenes with reliable background. Then the point cloud filling data augmentation strategy is used to generate confident data for iteratively learning the detector.

time-consuming and labor-intensive. Our proposed method adopts the sparse annotation strategy which just annotates one object for each scene, while achieving comparable performance with these fully-supervised methods. Moreover, whether voxel-based or point-based detectors, our SS3D can be directly applied.

2.2. Weakly/Semi-Supervised 3D Object Detection

To reduce annotations of 3D objects, the weakly-supervised learning strategy is adopted in WS3D [24], which is achieved by a two-stage architecture based on the click-annotated scheme. WS3D [10] generated cylindrical object proposals by click-annotated scenes in stage-1 and refined the proposals to get cuboids using slight well-labeled instances in stage-2. However, the supervision information provided by the weakly-supervised point annotation is weak, so that a certain amount of full annotations have to be provided additionally. Meanwhile, based on VoteNet [12], SESS [33] firstly proposed a semi-supervised 3D object detection, which leveraged a mutual teacher-student [22] framework to enforce three kinds of consistency losses. Following SESS, 3DIoUMatch [24] was proposed to estimate 3D IoU as a localization metric and set a self-adjusted threshold to filter pseudo labels.

Different from these methods, our proposed method makes precise supervision information of an object existing in each scene, which enables us to transfer reliable supervision information within a scene. Intuitively, this would be superior to transferring supervision information across scenes, especially for largely variable scenes.

2.3. Sparsely-Supervised 2D Object Detection

The sparsely annotated object detection is another way to reduce the dependence of networks on data annotation which only annotates a part of objects. Due to that a part of instances are missing annotated, weight updated of the network may be misguided significantly when gradients back-propagated. To address this issue, existing advanced methods employed re-weight or re-calibrates strategy on the loss of RoIs (regions of interest) to eliminate the effect of unlabeled instances. Soft sampling [27] utilized overlaps between RoIs and annotated instances to re-weight the loss. Background recalibration loss [31] based on focal loss [6] regarded the unlabeled instances as hard-negative samples and re-calibrates their losses, which is only applicable to single-stage detectors. Especially, part-aware sampling [11] ignored the classification loss for part categories by using human intuition for the hierarchical relation between labeled and unlabeled instances. Co-mining [25] proposed a co-generation module to convert the unlabeled instances as positive supervisions.

Above sparsely annotated object detection methods are all for 2D image objects. Due to the modal difference between 2D images and 3D point cloud, these methods can not be applied to our 3D object detection task. For example, in KITTI [2], 3D objects are naturally separated, which means the overlaps among objects are zero and the hierarchical relation between objects does not exist. Comparing with the re-weight and re-calibrates methods, in this paper, we propose a novel method for sparsely annotated 3D object detection which leverages a missing-annotated instance mining module and a simple but effective background mining module to mine confident positive instances and backgrounds, which is key for training detectors with high performance.

3. Method

3.1. Overall Framework

As a general framework, the proposed SS3D aims at facilitating the learning of a 3D detector to obtain the optimal detection performance when training from the scratch based on the sparsely annotated dataset. As shown in Fig. 2, the proposed SS3D is mainly composed of a missing-annotated instance mining module, a reliable background mining module, a point cloud filling data augmentation,

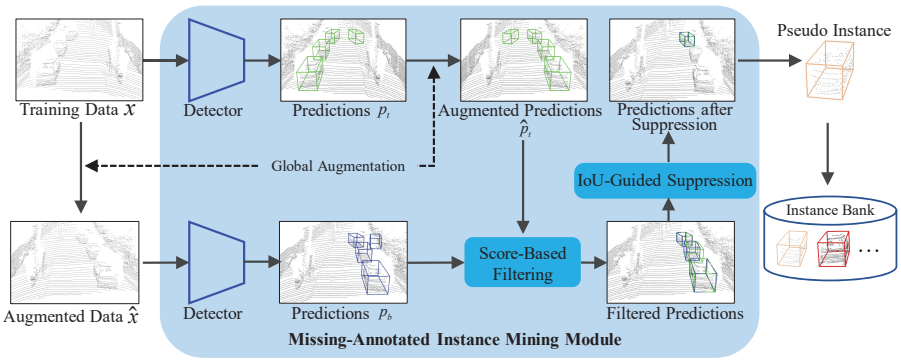

Figure 3. The illustration of our proposed missing-annotated instance mining module. The training data and corresponding augmented data are two different inputs for the detector. Then we leverage the score-based filtering to remove the augmented predictions of raw training data and predictions of augmented data with a low confidence score. Further, the IoU-guided suppression is proposed to filter out low-quality predictions. Lastly, we store the remained predictions as pseudo instances in the instance bank.

and an instance bank. Given a 3D detector, initially, we train the detector from the scratch on the sparsely annotated dataset. Then, we use the detector to mine reliable missing-annotated instances from the point cloud in training data through the missing-annotated instance mining module with strict filtering strategies. We store the mined instances (orange color) and original annotated instances (red color) into the instance bank. Relying on the instance bank, we further use the detector to mine reliable background through the reliable background mining module. Based on the results of these two modules, we leverage the proposed point cloud filling data augmentation to construct a confident data set, which can be further used to retrain the detector. By this iteratively learning style, we can finally obtain a 3D detector with high performance. Details are introduced below.

3.2. Architecture of Detector

Our method is a general framework for training 3D object detectors with the sparsely annotated dataset, which can be directly applied to kinds of detectors. In this paper, we verify our SS3D with state-of-the-art 3D detectors of PointRCNN [17], Part-A2 [18], PV-RCNN [16], and Voxel-RCNN [1]. We take PV-RCNN as an example and briefly review this method. PV-RCNN is a high-performance and efficient two-stage point cloud detector that deeply integrates both the multi-scale 3D voxel Convolutional Neural Network (CNN) features and the PointNet++-based set abstraction features to a small set of keypoints by the novel voxel set abstraction module.

3.3. Missing-Annotated Instance Mining Module

As shown in Fig. 3, we design a missing-annotated instance mining module, which combines IoU-guided suppression and a score-based filtering scheme as a strengthening measure for mining the unlabeled positive instances as high-quality pseudo instances. Then, selected pseudo instances are stored in the instance bank to further guide the reliable background mining module.

Score-based filtering As shown in Fig. 3, to start, the raw input point cloud x goes through the top detector to generate the predictions p_t. Then, we perform a set of global augmentation, which includes a random rotation, flipping, and scaling on x to generate augmented point cloud $\hat{x}$, in synchronizing with p_t to produce augmented predictions $\hat{p}_t$, and the bottom detector generates predictions p_b based on $\hat{x}$. Finally, we set a classification confidence threshold τ_{cls} to filter out predictions of p_b and $\hat{p}_t$ that may contain a wrong category and then obtain the filtered predictions.

IoU-guided suppression Note that only a score-based filtering strategy can not get reliable predictions. Inspired by FixMatch [20], we further propose an effective IoU-guided suppression strategy. After we get the filtered predictions, we calculate the IoU matrix between every pair of bounding boxes from $\hat{p}_t$ and p_b, aiming to match the boxes of two predictions from the irregular point cloud. Then we filter out outmatched paired bounding boxes with IoUs less than threshold τ_{IoU}, thus further improving the quality of pseudo instances.

Final-step instance bank processing Combining score-based filtering and IoU-guided suppression, we can avoid low-quality pseudo instances generation effectively and finally obtain a set of bounding boxes $\{b^r\}_{n=1}^{N}$, where N and r are the numbers of training scenes and bounding boxes remained in a scene, respectively. Then, we compute the IoU between boxes b_n^r and $b_n^{\mathcal{B}}$ (bounding boxes from the instance bank $\mathcal{B}$) of the same scene of index n, and choose b_n^r which does not overlap with $b_n^{\mathcal{B}}$. Finally, the chosen bounding boxes (orange color) along with corresponding predicted class labels and point cloud are stored in the instance bank which also contains all sparsely annotated in-

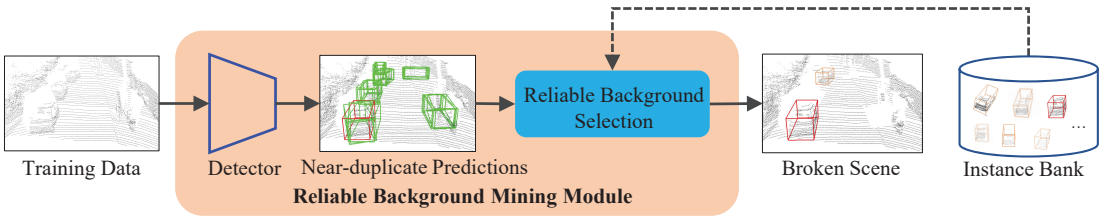

Figure 4. The illustration of our proposed reliable background mining module. To start, we feed the original point cloud to the detector without NMS to produce near-duplicate predictions, and leverage the instances stored in the instance bank to filter out unreliable object points. This will lead to broken scene which is further processed through the point cloud filling strategy.

stances (red color). By this design, with the iteration of the network, our instance bank can store more and more positive instances to guide the reliable background mining module for mining more reliable background.

3.4. Reliable Background Mining Module

Relying on the updated instance bank, we leverage the proposed reliable background mining module to mine background points and further eliminate the negative supervision information due to missing-annotated instances. Compared to the existing re-scale strategy [11, 31] for incorrect supervision, our approach is more simple and effective.

As shown in Fig. 4, to get reliable background point cloud, we adopt the strategy of finding potential foreground points as far as possible. Specifically, we use the detector with a low confidence score threshold τ_l to obtain object detection results. Meanwhile, we remove the Non-Maximum Suppression (NMS) operation from the detector. In this way, we make sure that the results contain potential foreground points as far as possible, which thus means that the rest of the original point cloud tend to be reliable background point cloud. For producing new training data, we remove the point data inside the 3D bounding boxes of the detected objects which do not overlap with the instances within the instance bank.

3.5. Point Cloud Filling Data Augmentation

After the reliable background selection processing, the point cloud scene is broken. Meanwhile, the instances in the scene may be very sparse. These issues will degrade the performance of the network significantly. Inspire by ground truth (GT) sampling augmentation proposed by [28], we further propose a point cloud filling data augmentation strategy to address these issues. For each remained bounding box, we randomly select a bounding box from the instance bank and place the corresponding point cloud inside the selected bounding box at the center of the remained bounding box, if the selected bounding box does not overlap with existing bounding boxes in the broken scene. Then we leverage the GT sampling augmentation [28] to further enhance the current scene. Finally, we obtain the merged point cloud with confident positive instances and reliable background. By this design, we can fix the density unevenness caused by deleting the points before, in the meantime, more ground truth boxes also reduce the negative impact on the network when only a small amount of instances are sparsely annotated in each scene.

Through the previous processing, ambiguous points that may cause negative impact on the network are largely removed, including those missing-annotated instances and not

Algorithm 1 Our SS3D Algorithm.

Input: Detector F trained from the scratch on the sparsely annotated training data D, instance bank $\mathcal{B}$, low score threshold τ_l, iteratively learning times M, training epoch E;

for $m = 1, 2, ..., M$ **do**
  for point cloud x in D **do**
    Preform missing-annotated mining on x;
    Update instance bank $\mathcal{B}$;
  end for
  for $e = 1, 2, ..., E$ **do**
    Shuffle the point cloud in training data D;
    for mini-batch D_k in D **do**
      for point cloud x in D_k **do**
        $P = F(x, W)$, with τ_l and no NMS;
        box_{gt} = boxes from $\mathcal{B}_x$;
        for box_i in P **do**
          if $IoU(box_i, box_{gt}) = 0$ **then**
            Delete points inside box_i in x;
          end if
        end for
        Point cloud filling data augmentation on x;
      end for
      Calculate the loss $\mathcal{L}$ on D_k ;
      Update the weight W of detector F by $\mathcal{L}$;
    end for
  end for
end for

Output: Updated weight parameter W

Method	Data	Car - 3D Detection			Car - BEV Detection			Cyclist - 3D Detection			Cyclist - BEV Detection		
		Easy	Mod	Hard	Easy	Mod	Hard	Easy	Mod	Hard	Easy	Mod	Hard
1. PointRCNN [17]	Full	88.88	78.63	77.38	90.21	87.89	85.51	86.13	69.70	65.40	87.16	73.47	67.61
2. PointRCNN [17]	Sparse (20%)	63.71	53.74	51.87	74.03	69.70	66.23	73.83	62.81	58.26	75.86	65.42	60.26
3. Ours (PointRCNN-based)	Sparse (20%)	87.18	77.10	76.13	89.74	87.41	85.71	86.62	73.22	66.92	87.21	74.27	71.54
4. *Improvements* 2 → 1	-	-25.17	-24.89	-25.51	-16.18	-18.19	-19.28	-12.30	-6.89	-7.14	-11.30	-8.05	-8.90
5. *Improvements* 3 → 1	-	-1.70	-1.53	-1.25	-0.47	-0.48	+0.20	+0.49	+3.52	+1.52	+0.05	+0.80	+3.93
1. Part-A^2 [18]	Full	89.47	79.47	78.54	90.42	88.61	87.31	85.50	69.90	64.48	86.92	73.35	70.77
2. Part-A^2 [18]	Sparse (20%)	72.92	64.41	60.49	79.38	75.38	71.81	74.52	63.39	58.91	76.23	66.26	61.91
3. Ours (Part-A^2-based)	Sparse (20%)	89.26	83.10	78.41	90.09	87.73	87.25	85.15	71.74	69.21	87.11	74.60	71.81
4. *Improvements* 2 → 1	-	-16.55	-15.06	-18.05	-11.04	-13.23	-15.50	-10.98	-6.51	-5.57	-10.69	-7.09	-8.86
5. *Improvements* 3 → 1	-	-0.21	+3.63	-0.13	-0.33	-0.88	-0.06	-0.35	+1.84	+4.73	+0.19	+1.25	+1.04
1. PV-RCNN [16]	Full	89.35	83.90	78.70	90.08	87.90	87.40	86.06	69.47	64.50	88.52	73.32	70.36
2. PV-RCNN [16]	Sparse (20%)	76.38	66.67	66.09	82.24	78.50	72.80	74.65	61.40	56.94	77.19	65.20	60.09
3. Ours (PV-RCNN-based)	Sparse (20%)	89.49	79.30	78.28	90.45	87.98	87.00	88.01	70.35	67.40	89.72	72.33	70.14
4. *Improvements* 2 → 1	-	-12.97	-17.23	-12.61	-7.84	-9.40	-14.60	-11.41	-8.07	-7.56	-11.33	-8.12	-10.27
5. *Improvements* 3 → 1	-	+0.14	-4.60	-0.42	+0.37	+0.08	-0.40	+1.95	+0.88	+2.90	+1.20	-0.99	-0.22
1. Voxel-RCNN [1]	Full	89.41	84.52	78.93	90.21	88.28	87.77	-	-	-	-	-	-
2. Voxel-RCNN [1]	Sparse (20%)	65.70	57.05	57.56	71.67	70.09	63.60	-	-	-	-	-	-
3. Ours (Voxel-RCNN-based)	Sparse (20%)	89.30	84.28	78.23	90.32	88.42	87.47	-	-	-	-	-	-
4. *Improvements* 2 → 1	-	-23.71	-27.47	21.37	-18.54	-18.19	-24.17	-	-	-	-	-	-
5. *Improvements* 3 → 1	-	-0.11	-0.24	-0.70	+0.11	+0.14	-0.30	-	-	-	-	-	-

Table 1. Comparison with different detectors trained with full annotations and extremely sparse split (20% instances of full annotations) on KITTI *val* split. The 3D object detection and bird's eye view detection are evaluated by mean average precision with 11 recall positions.

mined by our missing-annotated instance mining module. Moreover, confident data is generated, which provides vital supervision information to retrain the detector in an iterative manner. Algorithm 1 summarizes our SS3D.

4. Experiments

4.1. Datasets and Evaluation Metrics

Following the state-of-the-art methods [4, 8, 17, 18, 34, 35], we evaluate our SS3D on the KITTI 3D and BEV object detection benchmark [2]. This is a popular dataset widely used for performance evaluation and consists of full annotations for 3D object detection. There are 7,481 samples for training and 7,518 samples for test and we further divide the training samples into *train* split of 3,712 samples and *val* split of 3,769 samples as a common practice [16]. In addition, due to the occlusion and truncation levels of objects, the KITTI benchmark has three difficulty levels in the evaluation: easy, moderate, and hard. Following sparsely annotated dataset generation in [31], we randomly keep one annotated object in each 3D scene from *train* split to generate the extremely sparse split. Compared with the full annotation of all objects on KITTI, the extremely sparse split only need to be annotated with 20% objects. For fair comparisons, we report the mAP with 40 and 11 recall positions, with a 3D overlap threshold of 0.7, 0.5, 0.5 for the three classes: car, pedestrian and cyclist, respectively.

4.2. Implementation Details

At first, we train our detector in a supervised manner following PCDet [23] with the extremely sparse split, and keep the same supervised loss as the used detector. At the training stage, we adopt the ADAM optimizer and cosine annealing learning rate [9] with a batch size of 8 for 6 epochs. We set the low score threshold τ_l as 0.01 in reliable background selection. For score-based filtering and IoU-guided suppression, we set both the confidence score threshold τ_{cls} and the IoU threshold τ_{IoU} as 0.9. Note that we set the times of iteratively learning $M = 10$. In our global augmentation, we randomly flip each scene along X-axis and Y-axis with 0.5 probability, and then scale it with a uniformly sampled factor from $[0.8, 1.2]$. Finally, we rotate the point cloud around Z-axis with a random angle sampled from $\left[-\frac{\pi}{4}, \frac{\pi}{4}\right]$.

4.3. Comparisons with State-of-the-art Methods

Comparison with fully-supervised methods We compare the proposed method with four state-of-the-art fully-supervised methods: PointRCNN [17], Part-A2 [18], PV-RCNN [16], Voxel-RCNN [1], with fully-annotated *train* split and the extremely sparse *train* split, respectively, where these detectors trained on the extremely sparse split are used as the initial detectors of our method. The results of different methods are shown in Tab. 1.

It can be seen from the table, due to the negative impact of missing-annotated instances, the performance of the four detectors trained on extremely sparse split decrease by more than 10% on average. Our method significantly improves the performance of these detectors and makes them close to the performance of full supervision, which indicates that our method has a good effect on mining missing-annotated instances and reliable background.

The visualizations of our SS3D prediction results are il-

Data	Method (PV-RCNN-based)	Car - 3D Detection			Pedstrian - 3D Detection			Cyclist - 3D Detection		
		Easy	Mod	Hard	Easy	Mod	Hard	Easy	Mod	Hard
semi-1%	3DIoUMatch [24]	89.0	76.0	70.8	37.0	31.7	29.1	60.4	36.4	34.3
sparse-1%	Ours SS3D	**96.2**	**88.1**	**86.9**	**61.7**	**58.7**	**54.5**	**85.6**	**62.8**	**58.4**
semi-2%	3DIoUMatch [24]	-	78.7	-	-	48.2	-	-	56.2	-
sparse-2%	Ours SS3D	98.28	**89.2**	88.3	67.5	**62.3**	61.0	90.1	**72.2**	68.3

Table 2. Comparison with 3DIoUMatch on KITTI *val* split under 1% or 2% labeled data. Both our SS3D and 3DIoUMatch are based on PV-RCNN. We report the mAP with 40 recall positions, under IoU thresholds 0.5, 0.25, 0.25 for car, pedestrian, and cyclist, respectively.

Data	Method	Car - 3D Detection		
		Easy	Mod	Hard
weakly* + 534 precisely$^{\#}$	WS3D [10]	84.04	75.10	73.29
534 precisely$^{\#}$	Ours (Voxel-RCNN-based)	88.85	78.53	76.92
	Ours (PointRCNN-based)	85.59	75.85	73.93
	Ours (Part-A^2-based)	88.67	78.17	76.86
	Ours (PV-RCNN-based)	88.29	78.07	76.77

Table 3. Comparison with WS3D on KITTI *val* split. We report the mAP with 11 recall positions. '*' denotes the scenes with center-click and '$^{\#}$' denotes precisely-annotated instance.

lustrated in Fig. 5. For a better view of results, we project the prediction of 3D point cloud onto the corresponding color images. As we can see from this figure, the proposed method has high-quality prediction results.

Comparison with the semi-supervised method We compare the proposed method with the semi-supervised method 3DIoUMatch [24], which is based on the advanced detector PV-RCNN [16]. To make a fair comparison, we also adopt the PV-RCNN as the detector and keep all methods with the same number of annotated objects for training. In KITTI *train* split, there are 3,712 scenes and these scenes contain a total of 17,289 objects for cars, pedestrians, and cyclists. For semi-supervised methods, 1% labeled data means 37 ($3712 \times 1\%$) scenes, which include an average of 172 ($17289 \times 1\%$) labeled objects used for training. So as for 1% labeled data in our extremely sparse split, we randomly select 172 scenes including 172 labeled objects for training. We also test the case of 2% labeled training data for both methods. The results with different ratios of labeled data can be seen from Tab. 2, which illustrates that our SS3D significantly outperforms the current state-of-the-art, 3DIoUMatch, under three classes with all three difficulty levels. Compared with 3DIoUMatch, the greater advantage of our network is that only 172 scenes are used during training. We abandon the remaining scenes, while 3DIoUMatch uses all 3712 scenes in the *train* split for information transfer.

Comparison with the weakly-supervised method In the weakly-supervised method, WS3D [10], 500 scenes with center-click labels and 534 precisely-annotated instances are used to train the network. Since the standard detectors are not applicable with center-click labels, we only use the same 534 precisely-annotated instances to train our proposed SS3D. Tab. 3 shows the comparison results. Obviously, our SS3D with different 3D detectors achieves the highest results for all difficulty levels, outperforming WS3D by a large margin with less labeled efforts.

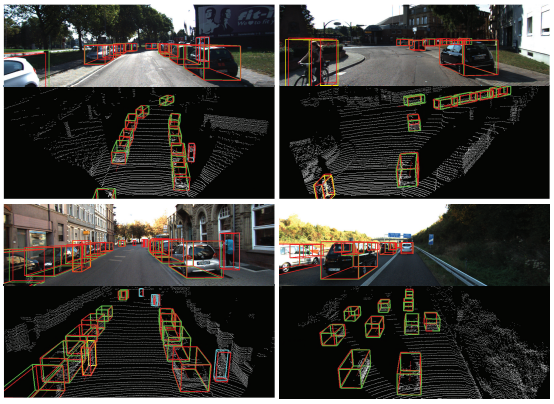

Figure 5. Qualitative results of our SS3D (PV-RCNN-based) on KITTI *val* dataset. The ground truth 3D bounding boxes of cars, cyclists, and pedestrians are drawn in green, yellow, and cyan, respectively. We set the predicted bounding boxes in red and project boxes in point cloud back onto the color images for visualization.

4.4. Ablation Study

In this section, we present a series of ablation studies to analyze the effects of our proposed modules in SS3D. Following the general principles, all models are trained on KITTI extremely sparse split and evaluated on *val* split. We take Voxel-RCNN [1] as our detector to conduct our ablation study due to the fast training speed, and our methods with other detectors are similar. Tab. 4 summarizes the ablation results on our IoU-guided suppression (IoU-GS), score-based filtering (Score-BF), reliable background mining module (RBMM), and point cloud filling data augmentation (PCFD) strategy. All results are with 11 recall points.

Effect of the reliable background mining module In the 1^{st} row of Tab. 4, we remove all modules, so it represents the standard Voxel-RCNN detector trained on the extremely sparse split. In the 2^{nd}, we add the RBMM and replace PCFD with the GT sampling [28]. Moreover, the instance bank only contains sparsely annotated instances without updating. Our reliable background mining module

IoU-GS	Score-BF	RBMM	PCFD	Car-3D Detection		
				Easy	Mod	Hard
-	-	-	-	65.70	57.05	57.56
-	-	✓	-	86.83	78.03	75.32
-	-	✓	✓	87.42	78.12	75.72
-	✓	✓	✓	88.57	82.78	76.12
✓	-	✓	✓	88.27	83.95	77.68
✓	✓	✓	✓	**89.30**	**84.28**	**78.23**

Table 4. Effects of the different components on our designed SS3D network. We report the mAP with 11 recall positions.

significantly boosts the performance in all three difficulty levels. This large improvement shows that mining reliable background can contribute to a better negative supervision removal caused by the missing-annotated instances.

Effect of the point cloud filling data augmentation strategy In the 3^{rd} row of Tab. 4, by combining RBMM and PCFD, our SS3D further improves the performance. This demonstrates our PCFD outperforms GT sampling data augmentation by fixing the structure information of the raw point cloud due to the previous points removal operation.

Effect of the missing-annotated instance mining module As shown in the 4^{th} and 5^{th} rows from Tab. 4, whether to use IoU-GS or Score-BF alone for pseudo instances filtering, it has a certain improvement compared with only using the reliable background mining module, indicating that more positive instances can contribute to a better model optimization. Further, by combining IoU-GS with Score-BF to obtain high-quality pseudo instances, our SS3D boosts the performance of the easy, moderate, and hard by about 1.88, 6.16, and 2.51 percentage points, respectively, as shown in the 3^{rd} and 6^{th} rows. This verifies the effectiveness of the jointly filtering strategy and also shows the importance of high-quality pseudo instances for the network.

4.5. Quality Analysis

In this section, we explore how our SS3D trains on the extremely sparse split and further analyze the quality of pseudo instances in the instance bank. The curves in Fig. 6 show that the coverage rate of the generated pseudo instances increases on the missing-annotated instances during the training process. Here coverage rate at a preset threshold means the percentage of missing-annotated instances that can pair a pseudo label with an IoU larger than the threshold [24]. As we can see from Fig. 6, in the beginning, the coverage rate of the pseudo instances is relatively low due to the strict filtering mechanism. As the training goes on, the improved detector leads to a higher passing rate of filter and hence a higher coverage of the pseudo instances, which in return fuels SS3D. At the end of training, the coverage rate at IoU@0.7 can achieve 0.75, which means our network effectively mines 75% of unlabeled instances.

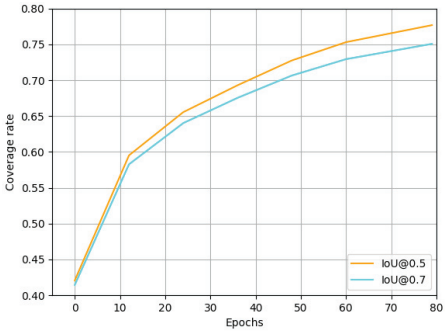

Figure 6. Pseudo instances coverage rate during the training process on extremely sparse split on KITTI.

4.6. Limitation

In principle, the performance of the fully-supervised method is the ceiling of our SS3D. However, in Tab. 1, our method even surpasses the fully-supervised methods in some cases, which may be due to that our method can mine some missing-annotated instances in the original dataset, and these missing instances may cause a negative impact on the training of the fully-supervised methods. For future work, we plan to validate the above hypotheses.

5. Conclusion

In this paper, we propose a novel method, called SS3D, to iteratively learn a 3D object detector from the sparsely annotated point cloud. Through the combination of our missing-annotated instance mining module and reliable background mining module, we largely ensure that each scene possesses confident supervision information when iteratively training the detector, hence eliminating the negative impact of missing-annotated instances of the sparsely annotated strategy. In addition, our SS3D is a general method that can be applied to learn any advanced detector. Extensive experiments validate the effectiveness of our proposed method with only 20% annotations, where our network achieves impressive results, which are close to the detector trained with the fully annotated dataset. Besides, our method exceeds the current semi-supervised and weakly-supervised methods on KITTI by a large margin.

Acknowledgments This work was supported in part by the National Key R&D Program of China (2020YFA0713900), in part by the National Natural Science Foundation of China under Grant 62176035, 61906025, 61721002, U1811461 and 62036007, in part by the Science and Technology Research Program of Chongqing Municipal Education Commission under Grant KJZD-K202100606, in part by the Macao Science and Technology Development Fund under Grant 061/2020/A.

References

[1] Jiajun Deng, Shaoshuai Shi, Peiwei Li, Wengang Zhou, Yanyong Zhang, and Houqiang Li. Voxel r-cnn: Towards high performance voxel-based 3d object detection. In *AAAI*, volume 35, pages 1201–1209, 2021. 1, 2, 4, 6, 7

[2] Andreas Geiger, Philip Lenz, Christoph Stiller, and Raquel Urtasun. Vision meets robotics: The kitti dataset. *Int. J. Rob. Res.*, 32(11):1231–1237, 2013. 1, 2, 3, 6

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

[4] Chenhang He, Hui Zeng, Jianqiang Huang, Xian-Sheng Hua, and Lei Zhang. Structure aware single-stage 3d object detection from point cloud. In *CVPR*, pages 11873–11882, 2020. 2, 6

[5] Alex H Lang, Sourabh Vora, Holger Caesar, Lubing Zhou, Jiong Yang, and Oscar Beijbom. Pointpillars: Fast encoders for object detection from point clouds. In *CVPR*, pages 12697–12705, 2019. 2

[6] Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection. In *ICCV*, pages 2980–2988, 2017. 3

[7] Baoyuan Liu, Min Wang, Hassan Foroosh, Marshall Tappen, and Marianna Pensky. Sparse convolutional neural networks. In *CVPR*, pages 806–814, 2015. 2

[8] Zhe Liu, Xin Zhao, Tengteng Huang, Ruolan Hu, Yu Zhou, and Xiang Bai. Tanet: Robust 3d object detection from point clouds with triple attention. In *AAAI*, volume 34, pages 11677–11684, 2020. 2, 6

[9] Ilya Loshchilov and Frank Hutter. SGDR: stochastic gradient descent with warm restarts. In *ICLR*, 2017. 6

[10] Qinghao Meng, Wenguan Wang, Tianfei Zhou, Jianbing Shen, Luc Van Gool, and Dengxin Dai. Weakly supervised 3d object detection from lidar point cloud. In *ECCV*, pages 515–531, 2020. 2, 3, 7

[11] Yusuke Niitani, Takuya Akiba, Tommi Kerola, Toru Ogawa, Shotaro Sano, and Shuji Suzuki. Sampling techniques for large-scale object detection from sparsely annotated objects. In *CVPR*, pages 6510–6518, 2019. 2, 3, 5

[12] Charles R Qi, Or Litany, Kaiming He, and Leonidas J Guibas. Deep hough voting for 3d object detection in point clouds. In *ICCV*, pages 9277–9286, 2019. 2, 3

[13] Charles R Qi, Hao Su, Kaichun Mo, and Leonidas J Guibas. Pointnet: Deep learning on point sets for 3d classification and segmentation. In *CVPR*, pages 652–660, 2017. 2

[14] Charles Ruizhongtai Qi, Li Yi, Hao Su, and Leonidas J. Guibas. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. In *NeurIPS*, pages 5099–5108, 2017. 2

[15] Zengyi Qin, Jinglu Wang, and Yan Lu. Weakly supervised 3d object detection from point clouds. In *ACM MM*, pages 4144–4152, 2020. 2

[16] Shaoshuai Shi, Chaoxu Guo, Li Jiang, Zhe Wang, Jianping Shi, Xiaogang Wang, and Hongsheng Li. Pv-rcnn: Point-voxel feature set abstraction for 3d object detection. In *CVPR*, pages 10529–10538, 2020. 1, 2, 4, 6, 7

[17] Shaoshuai Shi, Xiaogang Wang, and Hongsheng Li. Pointrcnn: 3d object proposal generation and detection from point cloud. In *CVPR*, pages 770–779, 2019. 1, 2, 4, 6

[18] Shaoshuai Shi, Zhe Wang, Jianping Shi, Xiaogang Wang, and Hongsheng Li. From points to parts: 3d object detection from point cloud with part-aware and part-aggregation network. *IEEE TPAMI*, 43(8):2647–2664, 2021. 1, 2, 4, 6

[19] Weijing Shi and Raj Rajkumar. Point-gnn: Graph neural network for 3d object detection in a point cloud. In *CVPR*, pages 1711–1719, 2020. 2

[20] Kihyuk Sohn, David Berthelot, Nicholas Carlini, Zizhao Zhang, Han Zhang, Colin A Raffel, Ekin Dogus Cubuk, Alexey Kurakin, and Chun-Liang Li. Fixmatch: Simplifying semi-supervised learning with consistency and confidence. *NeurIPS*, 33:596–608, 2020. 4

[21] Pei Sun, Henrik Kretzschmar, Xerxes Dotiwalla, Aurelien Chouard, Vijaysai Patnaik, Paul Tsui, James Guo, Yin Zhou, Yuning Chai, Benjamin Caine, et al. Scalability in perception for autonomous driving: Waymo open dataset. In *CVPR*, pages 2446–2454, 2020. 1

[22] Antti Tarvainen and Harri Valpola. Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results. In *NeurIPS*, pages 1195–1204, 2017. 3

[23] OpenPCDet Development Team. Openpcdet: An open-source toolbox for 3d object detection from point clouds. https://github.com/open-mmlab/OpenPCDet, 2020. 6

[24] He Wang, Yezhen Cong, Or Litany, Yue Gao, and Leonidas J Guibas. 3dioumatch: Leveraging iou prediction for semi-supervised 3d object detection. In *CVPR*, pages 14615–14624, 2021. 2, 3, 7, 8

[25] Tiancai Wang, Tong Yang, Jiale Cao, and Xiangyu Zhang. Co-mining: Self-supervised learning for sparsely annotated object detection. In *AAAI*, volume 35, pages 2800–2808, 2021. 3

[26] Yi Wei, Shang Su, Jiwen Lu, and Jie Zhou. Fgr: Frustum-aware geometric reasoning for weakly supervised 3d vehicle detection. In *ICRA*, pages 4348–4354, 2021. 2

[27] Zhe Wu, Navaneeth Bodla, Bharat Singh, Mahyar Najibi, Rama Chellappa, and Larry S. Davis. Soft sampling for robust object detection. In *BMVC*, page 225, 2019. 2, 3

[28] Yan Yan, Yuxing Mao, and Bo Li. Second: Sparsely embedded convolutional detection. *Sensors*, 18(10):3337, 2018. 2, 5, 7

[29] Zetong Yang, Yanan Sun, Shu Liu, and Jiaya Jia. 3dssd: Point-based 3d single stage object detector. In *CVPR*, pages 11040–11048, 2020. 2

[30] Zetong Yang, Yanan Sun, Shu Liu, Xiaoyong Shen, and Jiaya Jia. Std: Sparse-to-dense 3d object detector for point cloud. In *ICCV*, pages 1951–1960, 2019. 2

[31] Han Zhang, Fangyi Chen, Zhiqiang Shen, Qiqi Hao, Chenchen Zhu, and Marios Savvides. Solving missing-annotation object detection with background recalibration loss. In *ICASSP*, pages 1888–1892, 2020. 3, 5, 6

[32] Yanan Zhang, Di Huang, and Yunhong Wang. Pc-rgnn: Point cloud completion and graph neural network for 3d object detection. In *AAAI*, volume 35, pages 3430–3437, 2021. 2

[33] Na Zhao, Tat-Seng Chua, and Gim Hee Lee. Sess: Self-ensembling semi-supervised 3d object detection. In *CVPR*, pages 11079–11087, 2020. 2, 3

[34] Wu Zheng, Weiliang Tang, Sijin Chen, Li Jiang, and Chi-Wing Fu. Cia-ssd: Confident iou-aware single-stage object detector from point cloud. In *AAAI*, volume 35, pages 3555–3562, 2021. 1, 2, 6

[35] Wu Zheng, Weiliang Tang, Li Jiang, and Chi-Wing Fu. Se-ssd: Self-ensembling single-stage object detector from point cloud. In *CVPR*, pages 14494–14503, 2021. 1, 2, 6

[36] Yin Zhou and Oncel Tuzel. Voxelnet: End-to-end learning for point cloud based 3d object detection. In *CVPR*, pages 4490–4499, 2018. 2