2025 International Conference on Intelligent Computing
July 26-29, Ningbo, China
https://www.ic-icc.cn/2025/index.php

ObjectContrast: Self-supervised Point Cloud Pre-training via Object Feature Contrast

Nuo Xu[1,2], Qinghong Yang[1,2(✉)] and Weiguang Zhuang[2]

[1] School of Software, Beihang University, Beijing 100191, China
[2] Hangzhou International Innovation Institute, Beihang University, Hangzhou 311115, China
kyono@buaa.edu.cn, yangqh@buaa.edu.cn

Abstract. Current point cloud object detection methods rely on expensive manual annotation. Utilizing contrastive learning for self-supervised pre-training on unlabeled large-scale point clouds can reduce annotation costs and improve model performance. However, selecting effective features for instance discrimination is crucial for contrastive learning. Previous methods have constructed instances for pre-training at different levels, such as points, proposals, and scenes, but the features of these instances differ from the objects to be detected. Considering that instance discrimination tasks based on object-level features align with downstream object detection tasks, we propose a novel and efficient self-supervised point cloud object detection pre-training framework called ObjectContrast. To learn more effective point cloud representations, this framework constructs two self-supervised pre-training modules: object-level instance discrimination contrast (ObCo) and bounding box geometric contrast prediction (BoxCo). ObCo drives the model to learn general object representations to locate object foregrounds and determine categories. BoxCo enhances the model's geometric perception capabilities regarding the dimension and orientation of 3D bounding boxes. Extensive experiments on various detectors and datasets validate the efficiency and transferability of ObjectContrast. Compared with the state-of-the-art self-supervised pre-training methods, ObjectContrast demonstrates superior performance.

Keywords: Self-supervised, Point Cloud, Object Detection.

1 Introduction

Please note that the first paragraph of a section or subsection is not indented. The first paragraphs that follows a table, figure, equation etc. does not have an indent, either. Point cloud object detection is crucial for understanding 3D scenes and has attracted much attention in the fields of autonomous driving and robotics [1]. However, existing supervised learning-based methods for point cloud object detection rely on expensive and time-consuming manually annotated data, which limits their practical applications. In contrast, raw point cloud data is readily available, and pre-training object detection

✉ Corresponding author

models using large-scale unlabeled point clouds have shown great potential and research value. Self-supervised pre-training aims to learn invariant representation from large-scale unlabeled data and transfer it to downstream tasks, thereby significantly enhancing model performance and reducing data annotation costs. By designing specific pretext tasks, self-supervised learning (SSL) could capture general feature representations from unlabeled data and has achieved great success in fields such as natural language processing and computer vision [2,3].

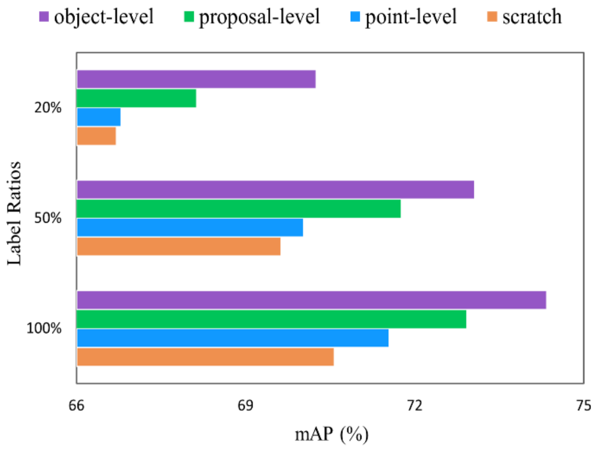

Fig. 1. Constructing different levels of contrast pairs for instance discrimination. Point-level [4] excessively emphasizes fine-grained details and cannot construct comprehensive object features. Proposal-level [5] randomly selects contrast regions, similarly failing to cover complete and accurate object features. Object-level directly utilizes object instances for discrimination, achieving more efficient learning of object representations. The mAP of the object-level is 3.0% higher than the point-level and 1.6% higher than the proposal-level on average.

Recently, significant progress has been made in SSL for point clouds, particularly through contrastive learning for model pre-training. Contrastive learning designs specific instance discrimination tasks to bring features of the same instance closer and push features of different instances apart, thereby learning and distinguishing general representation. Therefore, designing appropriate instance discrimination tasks is crucial for self-supervised pre-training. Existing SSL methods construct instances for discrimination at different levels, such as points [4], proposals [5], supervoxels [6], and scenes [7]. However, there is a certain gap between these instance features and the real object features. Point-level methods focus too much on fine-grained features and lack a holistic object description; scene-level methods are too coarse and fail to accurately locate objects; supervoxel-level and proposal-level methods select local regions of the point clouds as instances, but the randomly generated regions could not cover complete and accurate object instances. As a result, these methods usually need to construct a large number of contrast pairs to learn general representations from background features. Due to the difference between the pre-training instance discrimination tasks and downstream object detection tasks, we rethink the impact of instance features on the data-efficient transfer learning. We pre-train PV-RCNN on Waymo and transfer to KITTI with different label ratios in fine-tuning, as shown in Figure 1, by comparison, directly using object instances for discrimination can more effectively learn object-level representations, facilitating data-efficient transfer learning.

Contrastive learning first pre-trains the models through instance discrimination and then fine-tunes the models at downstream tasks. Given that object-level features differ from point-level, region-level, or scene-level features, the feature gap between upstream and downstream tasks could impair the data-efficient transfer learning. Based on the above analysis, we propose an efficient self-supervised framework called ObjectContrast. The framework uses object-level features for Contrastive learning, bridging the feature gap between pre-training and fine-tuning tasks. Specifically, ObjectContrast learns object-level point cloud representations through two modules: class-related object-level instance discrimination contrast (ObCo) and boundTheing box geometric contrast prediction (BoxCo), thereby achieving data-efficient transfer for object detection task. Firstly, since the number of object classes is very limited and objects often belong to the same class, the contrastive loss constructed by InfoNCE [8] can cause negative pairs of the same class to be pushed apart, which is detrimental to model convergence. ObCo addresses this by constructing a class-related object instance contrastive loss, treating augmented instances of the same object as positive pairs and other class objects and backgrounds as negative pairs, aiding the models in distinguishing foreground and class representations. Secondly, besides locating objects and determining classes, the object detection task also requires accurate prediction of 3D bounding boxes. BoxCo enhances the point cloud objects through random scaling and rotation, and by combining the augmented features with the original features, predicts scaling ratios and rotation angles to enhance the model's geometric prediction and perception capabilities. We conduct extensive experiments on the public datasets Waymo [9], ONCE [10], KITTI [11], as well as an autonomous driving engineering dataset, validating popular point cloud object detectors such as PV-RCNN [12], CenterPoint [13], and SECOND [14]. We compare ObjectContrast with point-level, region-level, and scene-level methods, the experimental results demonstrate the efficiency and superiority of the proposed self-supervised pre-training methods.

The main contributions of this work include the following three aspects:

- We rethink the impact of instance features on data-efficient transfer learning and propose a framework that utilizes object-level features for instance discrimination to bridge the feature gap between pre-training and fine-tuning tasks.
- We design two self-supervised tasks for object detection to enhance the classification and bounding box prediction capabilities of pre-trained models, including class-related object instance discrimination contrast and bounding box geometric contrast prediction.
- Our method achieves state-of-the-art performance compared to other self-supervised pre-training methods. Extensive experiments demonstrate the superior data-efficient transfer learning capabilities of the proposed method.

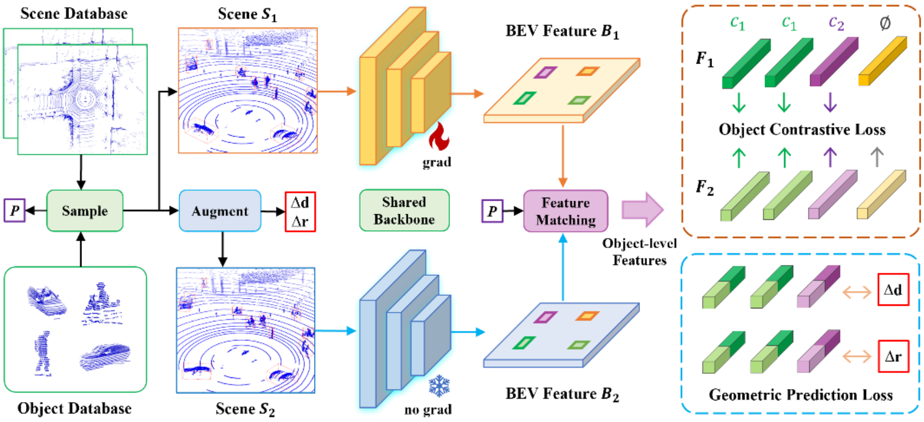

Fig. 2. Overview of the proposed ObjectContrast framework. The framework consists of four main components: data generation and augmentation, feature extraction and matching, object-level instance discrimination, and bounding box geometry prediction. Sample several objects O and a scene S. O undergoes rotation and scaling augmentation to obtain O', S combine with O to form S_1, and then combine with O' to form S_2. S_1 and S_2 are processed through a shared backbone network to obtain BEV features B_1 and B_2, ultimately extracting object-level features F_1 and F_2. The models are pre-trained through two self-supervised pretext tasks: ObCo and BoxCo, to learn object-level representations, facilitating the transfer of the pre-trained model to object detection.

2 Related Work

2.1 3D Object Detection in Point Clouds

Please note that the first paragraph of a section or subsection is not indented. The first paragraphs that follows a table, figure, equation etc. does not have an indent, either. With the rapid development of deep learning and the continuous emergence of large-scale datasets, 3D object detection algorithms based on point clouds have also proliferated. Inspired by research in the image domain, the point cloud domain has successively proposed anchor-based methods [14,15], center-based methods [13], and query-based methods [16]. Compared to images, point clouds are characterized by sparsity and disorder, necessitating encoding them into a structured form before feature extraction. According to the different coding schemes, the point cloud object detection methods can be divided into point-based [17], voxel-based [18], point-voxel hybrid methods [12], and graph-based [19]. Although point cloud object detection algorithms have made significant progress, the aforementioned methods usually rely on a large amount of labeled data for supervised learning. Therefore, it is crucial to reduce the dependency of detectors on labeled data by using self-supervised methods. Following the first principle thinking, object detection can be divided into two fundamental sub-tasks: object classification and bounding box prediction. We construct two self-supervised pretext tasks, ObCo and BoxCo, to enhance the pre-trained model's classification and geometric prediction capabilities, respectively. ObCo enhances the model's ability to distinguish between foreground and background representations and learn inter-class representations through contrastive learning. BoxCo improves the geometric perception of the models by predicting rotation and scaling.

2.2 Contrastive Learning for Self-Supervised Pre-training

In recent years, SSL has developed rapidly and has become an important research direction, especially with the emergence of many research achievements based on contrastive learning discriminative methods [20,21]. In the image domain, [22,23] extract the overall features of images as instances for discrimination, demonstrating excellent performance in classification tasks. As deep learning tasks become more complex, instances are designed with increasing sophistication. Considering that object detection tasks require predicting the position and size of bounding boxes, [24,25] construct object-level instance discrimination tasks to learn more precise object representations. In the point cloud domain, [7,26] treat the overall features of point clouds as instances for discrimination, which is relatively coarse and struggles to learn fine-grained object-level representations from scene-level features; [4,27] achieve point-level instance discrimination, which focuses too much on fragmented point-level features, ignoring the relevance of overall object features; [5,6,28,29,30] achieve contrastive learning by constructing super-voxels, proposals, and other region-level instances, which requires constructing a large number of contrast pairs to learn general representations from background features. We observe that existing contrastive learning pre-training methods construct instance discrimination tasks using point-level, scene-level, or region-level features, but there is a significant feature gap between these instances and the objects to be detected in downstream tasks. This prompts us to rethink the impact of instance design on transfer learning. Therefore, we propose an object-level point cloud instance discrimination method to bridge the feature gap between pre-training and fine-tuning, facilitating a smoother transfer of the pre-trained models to object detection task.

3 Methodology

SSL improves the performance of detectors and reduces reliance on labeled data through pre-training and fine-tuning. However, previous self-supervised contrastive learning methods exhibit discrepancies between pre-training and fine-tuning tasks. As shown in Figure 2, we propose an efficient self-supervised pre-training framework for point cloud object detection to unify the granularity of the instance features between pre-training and fine-tuning, bridge the feature gap between upstream and downstream tasks, and promote data-efficient transfer learning.

3.1 Data Generation and Augmentation

Existing self-supervised pre-training methods for point clouds construct region-level instances that lack explicit semantic information, introducing a large number of background features as contrast pairs. For example, [5] generates proposals through farthest point sampling, and [28] obtains regions via over-segmentation of point clouds, requiring the construction of 4096 pairs of positive and negative samples for each frame of point clouds. As shown in Figure 3, we obtain pseudo-labels through processes such as ground removal, clustering, and fitting 3D boxes to construct a point cloud object database, from which object-level instances are sampled during pre-training. The method of generating object-level instances described above has the following main advantages: first, the process is unsupervised and can be automated through rule-based methods; second, the generated point cloud objects have explicit semantic information, which facilitates the model in learning comprehensive object-level representations.

To avoid the influence of unknown point cloud objects in the scene, it is necessary to collect empty point cloud scenes that do not contain any point cloud objects, thereby constructing a point cloud scene database. For public datasets, we remove the point clouds within the annotated boxes for our experiments. For the proposed engineering dataset, we collected a large number of empty point cloud scenes during periods of low traffic flow. During pre-training, scenes S and objects $O = \{o_1, o_2, \dots, o_n\}$ are sampled from the point cloud database. The objects O are augmented through rotation and scaling to obtain $O' = \{o_1', o_2', \dots, o_n'\}$, where the rotation angles are $\Delta r = \{\Delta r_1, \Delta r_2, \dots, \Delta r_n\}$ and the scaling factors are $\Delta d = \{\Delta d_1, \Delta d_2, \dots, \Delta d_n\}$. S and O are combined to form S_1, and S and O' are combined to form S_2. To ensure the most realistic point clouds possible, the positions $P = \{p_1, p_2, \dots, p_n\}$ of the point cloud objects in S_1 and S_2 are kept as they were during collection.

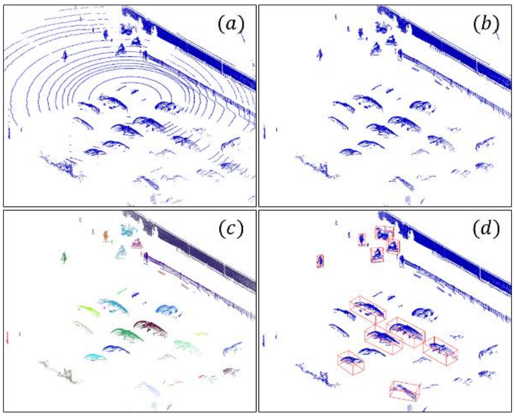

Fig. 3. Generate pseudo-labels of point cloud objects. (a) Original point cloud (b) Ground removal (c) Clustering (d) 3D box fitting.

3.2 Feature Extraction and Matching

The point cloud S_1 and S_2 are fed into the pre-trained model to obtain BEV features $B_1 \in \mathbb{R}^{C\times H\times W}$ and $B_2 \in \mathbb{R}^{C\times H\times W}$, respectively. After positional matching and filtering, the objects features $F = \{f_1, f_2, \dots, f_n\}$ and $F' = \{f_1', f_2', \dots, f_n'\}$ are obtained:

$$F_1 = \phi\big(g_q(S_1), P\big), \quad F_2 = \phi(g_k(S_2), P) \tag{1}$$

where g_q and g_k are backbone networks with shared parameters, and the parameters of g_k are frozen, not involved in the backpropagation process. ϕ is the object feature selection function, which extracts features F_1 and F_2 based on the position P of the point cloud objects in the scene.

In previous methods of contrastive learning for images and point clouds, the discrimination capability of the pre-trained models is improved by creating large-scale contrast pairs. However, in this study, the sparsity of point cloud object instances requires the extraction of background features as negative pairs to help the models better distinguish between foreground and background. We use similar feature mining to select hard samples in background features and enhance the efficiency of negative pair discrimination [31]. First, following [13], the heatmap ground truth $Y \in [0, 1]^{K\times H\times W}$ is constructed, where K is the number of point cloud object categories, and $H\times W$ is the feature map size. The ground truth of the heatmap for the category k is $Y_k \in [0, 1]^{H\times W}$. The meta-feature $E_k \in \mathbb{R}^C$ for each category is obtained by weighted averaging:

$$E_k = \frac{\sum_{i,j}^{H,W} Y_k(i,j)\cdot B_2(i,j)}{\sum_{i,j}^{H,W} Y_k(i,j)} \quad \text{for} \quad Y_k(i,j) = 1 \tag{2}$$

Background features are selected from $B = \{B_2(i, j) \mid Y_k\ (i, j) \neq 1,\ k = 1, \ldots, K\}$ as negative pairs. Since the point cloud scene S is empty, the ground truth category of B must be background, denoted as $\emptyset$, meaning it doesn't belong to any category. We use cosine similarity to select background features $U = \{u_1, u_2, \ldots, u_m\}$.

$$U = \mathrm{top}\big(\mathrm{sim}(E_k, B)\big), \quad k = 1, \ldots, K \tag{3}$$

where sim is the cosine similarity function, and top represents selecting the top M background features with the highest similarity to the category meta-features.

3.3 Object Instance Discrimination

Points and proposals often lack explicit semantic significance, whereas object-level instances exhibit a strong correlation with their corresponding categories. By using pseudo-labels, the object category of point clouds can be obtained. Within a batch of instances, there are multiple objects of the same category, and treating all other objects as negative pairs besides the instance itself would cause the same category objects to repel each other. Conversely, treating objects with the same category label as positive pairs can introduce noise due to the errors in pseudo-labels. Therefore, due to the limited number of categories and the presence of label errors, conventional contrastive learning loss functions would treat other objects of the same category as negative pairs, which hinders model convergence. We aim to cluster object instances of the same category together and separate instances of different categories. To achieve this, we propose ObCo, which effectively utilizes category prior information while mitigating the impact of incorrect category pseudo-labels.

The instances participating in contrastive learning include object features $F_1 = \{f_1, f_2, \ldots, f_n\}$, $F_2 = \{f_1', f_2', \ldots, f_n'\}$, and background features $U = \{u_1, u_2, \ldots, u_m\}$, with the object category set $C = \{c_1, c_2, \ldots, c_k, \emptyset\}$, where $c(f_i)$ denotes the category pseudo-label for the object feature f_i, and all elements in U belong to the category label $\emptyset$. Following the idea from [23] of viewing contrastive learning as a dictionary look-up task, we define the query instance set as $Q = F_1$ and the key instance set as $K = F_2 \cup U$. For a query instance f_i in Q, the positive pair is defined as $f_+ = f_i'$ where $f_i' \in F_2$, and the negative pair set is defined as $F_- = \{f_j \in F_2 \mid c(f_j) \neq c(f_i)\} \cup D$. The ObCo loss $\mathcal{L}_{obco}$ for all query instances is defined as:

$$\mathcal{L}_{obco} = -\sum_{-\,\mathrm{f_q} \in \mathrm{Q}} \log\left(\frac{\exp(\mathrm{f_q} \cdot \mathrm{f_+}/\tau)}{\exp(\mathrm{f_q} \cdot \mathrm{f_+}/\tau) + \sum_{\mathrm{f_-} \in \mathrm{F_-}} \exp(f_q \cdot f_-/\tau)}\right)) \tag{4}$$

where τ is the temperature coefficient, used to control the model's ability to distinguish hard negative samples [32]. In Figure 4, including $\{f_j \in F_2 \mid c(f_j) \neq c(f_i)\}$ as negative pairs helps the models to differentiate between object categories, while including D as negative pairs helps the model to separate foreground from background. $\{f_j \in F_2 \mid c(f_j) = c(f_i)\}$ represents the other objects in the same category, which are excluded from instance discrimination to avoid introducing noise that could disrupt model convergence.

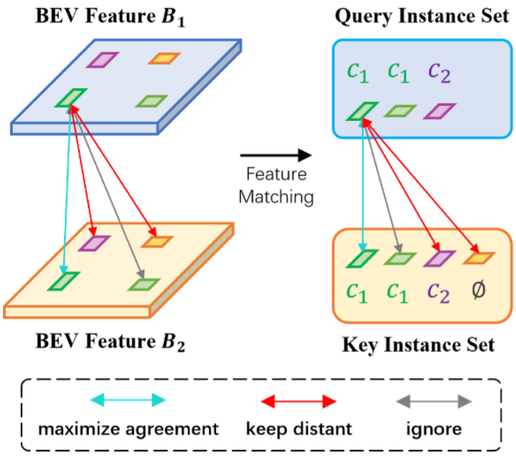

Fig. 4. The class-related object-level instance discrimination contrast

3.4 Bounding Box Geometric Prediction

In object detection task, it is essential not only to accurately identify and classify foreground objects in a scene but also to predict precise object bounding boxes. This requires the models to have geometric awareness of the object's dimension and orientation. We propose BoxCo, which directly constructs object-level geometric prediction tasks to efficiently learn object-level geometric representations. Compared to [27], which uses point-level features to predict scene-level geometric information, BoxCo's design of using object-level features to predict object-level geometric information is more reasonable and more easily transferable to downstream object detection task.

We do not treat the augmented object O' as the ground truth for prediction, because there exists a distributional difference in the absolute values of the bounding box data between O' and the source object O. Predicting O' directly through supervised learning could negatively impact the performance of the models in the actual data distribution [33]. BoxCo uses the relative differences in orientation and dimension, Δr and Δd, between O and O' as the ground truth for self-supervised learning. This approach makes the models more sensitive to differences in orientation and dimension without changing the distribution of the real point cloud data.

BoxCo first connects the object features corresponding to F_1 and F_2 to obtain the joint features, and then regresses the predicted values through a fully connected layer. Δr and Δd are the rotation angle and scaling factor during data augmentation. Using Δr and Δd as ground truths guides the model to learn object-level geometric representations, the final BoxCo loss $\mathcal{L}_{boxco}$ as:

$$\mathcal{L}_{boxco} = \mathrm{MSE}\big(\mathrm{MLP}([F_1, F_2]), \triangle r\big) + \mathrm{MSE}\big(\mathrm{MLP}([F_1, F_2]), \triangle d\big) \tag{5}$$

where MLP denotes the fully connected layer, and MSE represents the mean squared error.

3.5 Self-supervised Pre-training Losses

We design a specific self-supervised loss function for object detection tasks to enable the pre-trained models to be more smoothly transferred to downstream tasks. First, considering that the ultimate goal of object detection tasks is to locate objects and predict categories, our proposed ObCo emphasizes both intra-class differentiation and foreground-background differentiation. Second, object detection tasks require accurate prediction of bounding box parameters, and BoxCo enhances the pre-train model's geometric awareness of bounding box dimension and orientation. The total losses of the proposed ObjectContrast self-supervised pre-training framework are:

$$\mathcal{L} = \alpha\mathcal{L}_{obco} + \beta\mathcal{L}_{boxco} \tag{6}$$

α and β are weighting parameters used to balance the relative importance of the two losses.

4 Experiment

4.1 Datasets and Evaluation Metrics

Waymo Open Dataset. [9] contains a training set with 798 sequences and a validation set with 202 sequences. We use the entire point clouds from the training set without the labels for self-supervised pre-training.

KITTI Dataset. [11] contains 7,481 annotated point cloud scenes, with 3,712 scenes in the training set and 3,769 scenes in the validation set. We use the mean average precision (mAP) under forty recall thresholds (R40) as the evaluation metric for the detectors.

ONCE Dataset. [10] is an autonomous driving dataset proposed for semi/self-supervised learning, with the evaluation metric being orientation-aware AP. The labeled data are split into a training set of 5k scenes and a validation set of 3k scenes. The unlabeled data are divided into 3 subsets: U_{small}, U_{medium}, and U_{large}, having 100k, 500k, and 1M scenes, respectively. We use the U_{small} to pre-train models to ensure consistency with previous research.

Bus Engineering Dataset. To demonstrate the applicability of ObjectContrast, we also conduct experiments on a real-world autonomous driving engineering dataset, which is available upon request. The dataset is collected by an autonomous driving bus equipped with 4 LiDAR sensors. The annotation format and evaluation metrics adhere to those of KITTI. The dataset is divided into a training set with 10k scenes, a test set with 4k scenes, and an additional 50k unlabeled point cloud scenes for self-supervised pre-training.

4.2 Implementation Details

To evaluate the proposed ObjectContrast, we follow the previous experimental protocol, primarily evaluating the transfer learning ability and the data-efficient learning ability. We pre-train PV-RCNN on unlabeled point clouds from the Waymo Open Dataset, then fine-tune the detector on the KITTI dataset. We compare the performance with scene-level, point-level, and region-level self-supervised pre-training methods to evaluate the data-efficient transfer learning ability. Additionally, we pre-train CenterPoint and SECOND on the U_{small} unannotated subset from the ONCE dataset and then fine-tuned the detectors on the ONCE training set. Similarly, we pre-train CenterPoint on unlabeled point clouds from the Bus Engineering Dataset and then fine-tune the detector on the training set to evaluate the data-efficient learning ability.

The hyperparameter settings for ObjectContrast are as follows, the total number of positive/negative contrast pairs is 4096, the upper limit for point cloud object instance sampling is 100, and the remaining instances are supplemented with background features. The weighting parameters are set as $\alpha = \beta = 1$, the temperature parameter for $\mathcal{L}_{obco}$ is set to 0.1, the rotation angle $\Delta r \in (-\pi/2, \pi/2)$, and the scaling factor $\Delta d \in (0.85, 1.15)$. Other experimental hyperparameters follow [5,28,34].

4.3 Data-efficient Transfer Learning

We pre-train the backbone network on Waymo and fine-tune it on KITTI to evaluate the data-efficient learning performance of ObjectContrast. Using PV-RCNN as the point cloud object detector, we fine-tune it with 20% and 100% of the annotated data. As Table 1 shows, the experimental results indicate that ObjectContrast consistently outperforms scene-level, point-level, and region-level self-supervised pre-training methods. This demonstrates that object-level pre-trained models can more smoothly transfer to downstream object detection tasks. We observe that region-level methods generally perform better than scene-level and point-level methods. Since point cloud objects can be considered as more precise region-level proposals, this highlights that constructing instance discrimination tasks closely aligned with object features is beneficial for transferring pre-trained models to object detection tasks.

Table 1. Data-efficient 3D Object Detection on KITTI. We pre-train the backbone network of PV-RCNN on Waymo and transfer to KITTI with 20% and 100% label ratios in fine-tuning, [35] and [29] were pre-trained on NuScenes [36]. ObjectContrast achieves state-of-the-art performance in two label ratios, compared to the scene-level method STRL [7], the point-level method PointContrast [4], the region-level methods ProposalContrast [5] and FAC [28]. "Scratch" denotes the model trained from scratch.

Label Ratios	Pre-training Schedule	mAP (Mod)	Car			Pedestrian			Cyclist		
			Easy	Mod	Hard	Easy	Mod	Hard	Easy	Mod	Hard
20%	Scratch	66.71	91.81	82.52	80.11	58.78	53.33	47.61	86.74	64.28	59.53
	[5]	68.13	91.96	82.65	80.15	62.58	55.05	50.06	88.58	66.68	62.32
	[28]	69.73	92.87	83.68	82.32	64.15	56.78	51.29	89.65	68.65	65.63
	Ours	**70.25**	**93.26**	**84.36**	**82.46**	**64.25**	**57.21**	**52.04**	**90.57**	**69.19**	**65.67**
80%	Scratch	70.57	-	84.50	-	-	57.06	-	-	70.14	-
	[7]	71.46	-	84.70	-	-	57.80	-	-	71.88	-
	[4]	71.55	91.40	84.18	82.25	65.73	57.74	52.46	91.47	72.72	67.95
	[5]	72.92	92.45	84.72	82.47	68.43	60.36	55.01	92.77	73.69	69.51
	[28]	73.95	92.98	86.33	83.82	69.39	61.27	56.36	**93.75**	74.85	71.23
	[35]	72.50	-	84.90	-	-	57.80	-	-	75.00	-
	[29]	72.10	-	84.80	-	-	57.30	-	-	74.20	-
	Ours	**74.34**	**93.42**	**86.45**	**84.75**	**70.16**	**61.48**	**57.09**	93.69	**75.11**	**71.82**

Table 2. 3D Object Detection Performance on ONCE validation set. We pre-train the backbone networks of the detectors on the U_{small} unannotated set from ONCE and fine-tune the detectors on the ONCE training set. ObjectContrast achieves state-of-the-art performance in two detectors, compared to other self-supervised methods.

Detector	Methods	mAP	Orientation-aware AP		
			Vehicle	Pedestrian	Cyclist
CenterPoint [13]	Scratch	64.24	75.26	51.65	65.79
	ProposalContrast [5]	66.24	78.00	52.56	68.17
	Ours	**68.08**	**79.95**	**54.82**	**69.48**
SECOND [14]	Scratch	51.89	71.19	26.44	58.04
	BYOL [37]	46.04	68.02	19.50	50.61
	PointContrast [4]	49.98	71.07	22.52	56.36
	SwAV [38]	51.96	72.71	25.13	58.05
	ALSO [35]	52.58	71.73	28.16	58.13
	Ours	**53.43**	**72.41**	**29.02**	**58.87**

We also investigate intra-domain transfer and data-efficient learning capabilities. For a fair comparison with previous works, we pre-train the backbone network of CenterPoint and SECOND on the U_{small} unlabeled point cloud from ONCE and fine-tune them on the training set. As shown in Table 2, ObjectContrast consistently outperformed existing self-supervised methods. We also conduct experiments on a real-world autonomous driving engineering dataset. We pre-train the backbone network of CenterPoint and fine-tune the model using 1%, 10%, and 100% of the training dataset to evaluate the model's performance under extremely scarce data conditions. As Table 3 shows, the experimental results indicate that compared to training from scratch, the self-supervised pre-trained CenterPoint shows a significant performance improvement. The lower the annotation rate, the more pronounced the improvement, particularly with an annotation rate of only 1%, where the mAP increased by 15.14%.

Table 3. Data-efficient 3D Object Detection on the Bus Engineering Dataset. We pre-train the backbone network of CenterPoint on the unannotated set and fine-tune CenterPoint using 1%, 10%, and 100% of the training set. the CenterPoint pre-trained with ObjectContrast shows significant performance improvement compared to training from scratch, and the lower the label ratio, the more noticeable the performance gain.

Label Ratios	Pre-training Schedule	mAP (Mod)	Car	Pedestrian	Cyclist
1%	Scratch	28.23	48.12	10.73	25.85
	Ours	**43.37**	**63.28**	**26.47**	**40.36**
10%	Scratch	45.93	63.20	33.22	41.38
	Ours	**56.78**	**72.93**	**43.62**	**53.80**
100%	Scratch	67.52	84.45	52.89	65.23
	Ours	**69.60**	**85.12**	**56.76**	**66.92**

4.4 Ablation Study and Analysis

Effectiveness of the components. We conduct ablation experiments on ObjectContrast to verify the effectiveness of each component. The baseline is CenterPoint trained from scratch on 10% of the training set from the Bus Engineering Dataset. Case 1 and Case 2 are pre-trained using only ObCo or BoxCo, respectively. As shown in Table 4, both ObCo and BoxCo improve baseline performance to varying degrees, especially ObCo. Benefiting from the class-related object-level instance discrimination task, ObCo facilitates the pre-trained model to learn comprehensive and accurate object representations, thereby achieving smoother transfer to object detection tasks. Furthermore, contrastive learning typically constructs a large number of negative pairs. To investigate the impact of negative pairs, we reduce the number of negative pairs by removing background features from the set of key instances, resulting in ObCo-. The experimental results indicate that even though object-level instances can more accurately reflect target features, it is still necessary to construct large-scale negative pairs.

ObCo introduces richer visual representations and enhances the ability to distinguish between foreground and background by constructing large-scale negative pairs.

Table 4. The effectiveness of the different components.

Case	ObCo	ObCo-	BoxCo	MAP
Baseline	-	-	-	45.93
Case1	✓	-	-	53.06
Case2	-	-	✓	48.12
Case3	-	✓	✓	52.48
ObjectContrast	✓	-	✓	**56.78**

Object-level query instances. ObjectContrast can be considered a specialized region-level contrastive learning method. Unlike proposals and super-voxels, the instances constructed by ObjectContrast are closer to complete and accurate point cloud objects, which can improve contrastive learning efficiency and bridge the gap between upstream and downstream tasks. To verify whether object-level instances have more efficient representation learning, we adjust the proportion of object features in the object instance query set and visualized the features using t-SNE. Specifically, ObCo set the number of point cloud object instances sampled to 100, meaning that the object instance query set contained 100 object features. By replacing object features with background features, the proportion of object instances was reduced. As shown in Figure 5, the four subplots represent the results of the feature visualization when the proportion of objects in the query set is 0%, 20%, 50%, and 100%, respectively. The experiments demonstrate that increasing the proportion of object instances is beneficial for enlarging the inter-class distance. Compared to randomly selected background regions, object instances with clear semantic information are more conducive to the model learning discriminative representations.

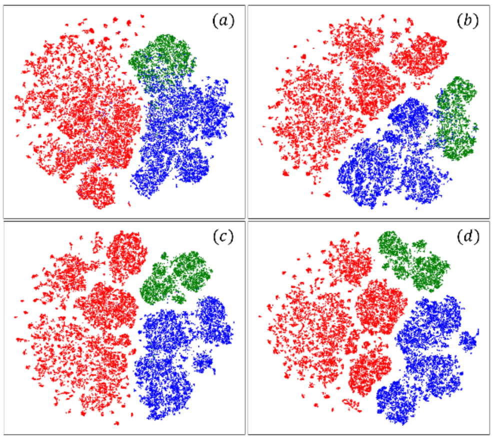

Fig. 5. The t-SNE visualization of object instance proportion. (a) 0% (b) 20% (c) 50% (d) 100%. Red represents car, blue represents cyclist and green represents pedestrian.

5 Conclusion

We propose ObjectContrast, a self-supervised pre-training framework designed for point cloud object detection. Despite previous research for point-level, region-level, and scene-level instance discrimination, the features of these instances differ from the real object features. Selecting critical features for instance discrimination is crucial for contrastive learning. Considering the upstream and downstream feature gap between pre-training and fine-tuning, we rethink the impact of instance design on task transfer and specifically construct two self-supervised pretext tasks: ObCo and BoxCo. ObCo achieves class-related instance discrimination, enabling the models to learn general object representations for locating object foregrounds and determining categories. BoxCo predicts the relative differences in 3D bounding boxes, making the model more sensitive to variations in orientation and dimension. Extensive experiments demonstrate that ObjectContrast exhibits superior data-efficient transfer learning capabilities compared to existing point-level, region-level, and scene-level self-supervised pre-training methods.

References

1. Mao, J., Shi, S., Wang, X., Li, H.: 3d object detection for autonomous driving: A comprehensive survey. International Journal of Computer Vision **131**(8), 1909–1963 (2023)
2. Huang, J., Zhao, K., Li, C., Lin, Y., Liu, Z., Wang, K., Lian, S.: Self-supervised visual anomaly detection with image patch generation and comparison networks. In: International Conference on Intelligent Computing. pp. 96–113. Springer (2024)
3. Yuan, X., Zhang, H., Li, T., Zhang, S., Zhang, X.: Multilingual knowledge graph completion with negative sample balance based adaptive self-supervised graph alignment. In: International Conference on Intelligent Computing. pp. 346–358. Springer (2024)
4. Xie, S., Gu, J., Guo, D., Qi, C.R., Guibas, L., Litany, O.: Pointcontrast: Unsupervised pre-training for 3d point cloud understanding. In: Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part III 16. pp. 574–591. Springer (2020)
5. Yin, J., Zhou, D., Zhang, L., Fang, J., Xu, C.Z., Shen, J., Wang, W.: Proposalcontrast: Unsupervised pre-training for lidar-based 3d object detection. In: European conference on computer vision. pp. 17–33. Springer (2022)
6. Chen, Z., Xu, H., Chen, W., Zhou, Z., Xiao, H., Sun, B., Xie, X., et al.: Pointdc: Unsupervised semantic segmentation of 3d point clouds via cross-modal distillation and super-voxel clustering. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 14290–14299 (2023)
7. Huang, S., Xie, Y., Zhu, S.C., Zhu, Y.: Spatio-temporal self-supervised representation learning for 3d point clouds. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 6535–6545 (2021)
8. Oord, A.v.d., Li, Y., Vinyals, O.: Representation learning with contrastive predictive coding. arXiv preprint arXiv:1807.03748 (2018)
9. Sun, P., Kretzschmar, H., Dotiwalla, X., Chouard, A., Patnaik, V., Tsui, P., Guo, J., Zhou, Y., Chai, Y., Caine, B., et al.: Scalability in perception for autonomous driving: Waymo

open dataset. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 2446–2454 (2020)

10. Mao, J., Niu, M., Jiang, C., Liang, H., Chen, J., Liang, X., Li, Y., Ye, C., Zhang, W., Li, Z., et al.: One million scenes for autonomous driving: Once dataset. arXiv preprint arXiv: 2106.11037 (2021)
11. Geiger, A., Lenz, P., Urtasun, R.: Are we ready for autonomous driving? the kitti vision benchmark suite. In: 2012 IEEE conference on computer vision and pattern recognition. pp. 3354–3361. IEEE (2012)
12. Shi, S., Guo, C., Jiang, L., Wang, Z., Shi, J., Wang, X., Li, H.: Pv-rcnn: Point-voxel feature set abstraction for 3d object detection. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 10529–10538 (2020)
13. Yin, T., Zhou, X., Krahenbuhl, P.: Center-based 3d object detection and tracking. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 11784–11793 (2021)
14. Yan, Y., Mao, Y., Li, B.: Second: Sparsely embedded convolutional detection. Sensors **18**(10), 3337 (2018)
15. Lang, A.H., Vora, S., Caesar, H., Zhou, L., Yang, J., Beijbom, O.: Pointpillars: Fast encoders for object detection from point clouds. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 12697–12705 (2019)
16. Misra, I., Girdhar, R., Joulin, A.: An end-to-end transformer model for 3d object detection. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 2906–2917 (2021)
17. Shi, S., Wang, X., Li, H.: Pointrcnn: 3d object proposal generation and detection from point cloud. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 770–779 (2019)
18. Li, J., Luo, C., Yang, X.: Pillarnext: Rethinking network designs for 3d object detection in lidar point clouds. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 17567–17576 (2023)
19. Shi, W., Rajkumar, R.: Point-gnn: Graph neural network for 3d object detection in a point cloud. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 1711–1719 (2020)
20. Fei, B., Yang, W., Liu, L., Luo, T., Zhang, R., Li, Y., He, Y.: Self-supervised learning for pre-training 3d point clouds: A survey. arXiv preprint arXiv:2305.04691 (2023)
21. Xiao, A., Huang, J., Guan, D., Zhang, X., Lu, S., Shao, L.: Unsupervised point cloud representation learning with deep neural networks: A survey. IEEE Transactions on Pattern Analysis and Machine Intelligence **45**(9), 11321–11339 (2023)
22. Chen, T., Kornblith, S., Norouzi, M., Hinton, G.: A simple framework for contrastive learning of visual representations. In: International conference on machine learning. pp. 1597–1607. PMLR (2020)
23. He, K., Fan, H., Wu, Y., Xie, S., Girshick, R.: Momentum contrast for unsupervised visual representation learning. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 9729–9738 (2020)
24. Li, M., Wu, J., Wang, X., Chen, C., Qin, J., Xiao, X., Wang, R., Zheng, M., Pan, X.: Aligndet: Aligning pre-training and fine-tuning in object detection. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 6866–6876 (2023)
25. Yang, C., Wu, Z., Zhou, B., Lin, S.: Instance localization for self-supervised detection pretraining. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 3987–3996 (2021)

26. Zhang, Z., Girdhar, R., Joulin, A., Misra, I.: Self-supervised pretraining of 3d features on any point-cloud. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 10252–10263 (2021)
27. Shi, W., Rajkumar, R.R.: Self-supervised pretraining for point cloud object detection in autonomous driving. In: 2022 IEEE 25th International Conference on Intelligent Transportation Systems (ITSC). pp. 4341–4348. IEEE (2022)
28. Liu, K., Xiao, A., Zhang, X., Lu, S., Shao, L.: Fac: 3d representation learning via foreground aware feature contrast. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 9476–9485 (2023)
29. Sautier, C., Puy, G., Boulch, A., Marlet, R., Lepetit, V.: Bevcontrast: Self-supervision in bev space for automotive lidar point clouds. In: 2024 International Conference on 3D Vision (3DV). pp. 559–568. IEEE (2024)
30. Shrout, O., Nitzan, O., Ben-Shabat, Y., Tal, A.: Patchcontrast: Self-supervised pre-training for 3d object detection. arXiv preprint arXiv:2308.06985 (2023)
31. Xia, Q., Deng, J., Wen, C., Wu, H., Shi, S., Li, X., Wang, C.: Coin: Contrastive instance feature mining for outdoor 3d object detection with very limited annotations. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 6254–6263 (2023)
32. Wang, F., Liu, H.: Understanding the behaviour of contrastive loss. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 2495–2504 (2021)
33. Reuse, M., Simon, M., Sick, B.: About the ambiguity of data augmentation for 3d object detection in autonomous driving. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 979–987 (2021)
34. Team, O., et al.: Openpcdet: An open-source toolbox for 3d object detection from point clouds. https://github.com/open-mmlab/OpenPCDet (2020)
35. Boulch, A., Sautier, C., Michele, B., Puy, G., Marlet, R.: Also: Automotive lidar self-supervision by occupancy estimation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 13455–13465 (2023)
36. Caesar, H., Bankiti, V., Lang, A.H., Vora, S., Liong, V.E., Xu, Q., Krishnan, A., Pan, Y., Baldan, G., Beijbom, O.: nuscenes: A multimodal dataset for autonomous driving. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 11621–11631 (2020)
37. Grill, J.B., Strub, F., Altché, F., Tallec, C., Richemond, P., Buchatskaya, E., Doersch, C., Avila Pires, B., Guo, Z., Gheshlaghi Azar, M., et al.: Bootstrap your own latent-a new approach to self-supervised learning. Advances in neural information processing systems **33**, 21271–21284 (2020)
38. Caron, M., Misra, I., Mairal, J., Goyal, P., Bojanowski, P., Joulin, A.: Unsupervised learning of visual features by contrasting cluster assignments. Advances in neural information processing systems **33**, 9912–9924 (2020)