Computational Visual Media
https://doi.org/10.1007/s41095-023-0366-0 Vol. 9, No. 4, December 2023, 875–892

Research Article

EFECL: Feature encoding enhancement with contrastive learning for indoor 3D object detection

Yao Duan[1], Renjiao Yi[1], Yuanming Gao[1], Kai Xu[1], and Chenyang Zhu[1] (✉)

Abstract Good proposal initials are critical for 3D object detection applications. However, due to the significant geometry variation of indoor scenes, incomplete and noisy proposals are inevitable in most cases. Mining feature information among these “bad” proposals may mislead the detection. Contrastive learning provides a feasible way for representing proposals, which can align complete and incomplete/noisy proposals in feature space. The aligned feature space can help us build robust 3D representation even if bad proposals are given. Therefore, we devise a new contrast learning framework for indoor 3D object detection, called *EFECL*, that learns robust 3D representations by contrastive learning of proposals on two different levels. Specifically, we optimize both instance-level and category-level contrasts to align features by capturing instance-specific characteristics and semantic-aware common patterns. Furthermore, we propose an enhanced feature aggregation module to extract more general and informative features for contrastive learning. Evaluations on ScanNet V2 and SUN RGB-D benchmarks demonstrate the generalizability and effectiveness of our method, and our method can achieve 12.3% and 7.3% improvements on both datasets over the benchmark alternatives. The code and models are publicly available at `https://github.com/YaraDuan/EFECL`.

Keywords indoor scene; object detection; contrastive learning; feature enhancement

1 Introduction

In recent years, RGB-D cameras and LiDAR devices are widely adopted for 3D data collection, which produces massive large-scale indoor or outdoor datasets for 3D object detection, such as ScanNet [1], SUN RGB-D [2], KITTI [3], and so on. The detection of objects holds paramount significance in various fields of 3D vision, including augmented reality, robot navigation, robot grasping, etc. Nevertheless, the direct detection of objects from raw point clouds continues to present substantial challenges.

A main line of research focuses on object detection by generating candidate proposals and subsequently performing box regression and object classification tasks [4–10]. Certain methods adopt object center prediction and point aggregation to generate high-quality proposals. For example, VoteNet [4] utilizes deep Hough voting for object centers, H3DNet [5] employs a hybrid set of geometric primitives for more accurate proposals, and BRNet [11] back-traces representative points from the centers. Additionally, numerous other approaches concentrate on extracting high-dimensional abstract features from proposals through the design of complicated networks or the incorporation of supplementary input information, including MLCVNet [12], Imvotenet [13], and GroupFreed3D [8], among others. However, these advanced approaches inherently produce incomplete and noisy proposals, which is particularly evident in the diverse and complex objects found in indoor scenes (see Fig. 1). To mitigate the impact of incomplete proposals, GroupFree3D selectively samples proposals based on their high objectness scores.

ProposalContrast [14] recently introduced contrastive learning to align the features of proposals obtained from multiple views within an instance. Similarly, proposals derived from

1 School of Computing, National University of Defense Technology, Changsha 410000, China. E-mail: Y. Duan, duanyao16@nudt.edu.cn; R. Yi, yirenjiao@nudt.edu.cn; Y. Gao, 356232063@qq.com; K. Xu, kevin.kai.xu@gmail.com; C. Zhu, zhuchenyang07@nudt.edu.cn (✉).

Manuscript received: 2023-04-07; accepted: 2023-07-02

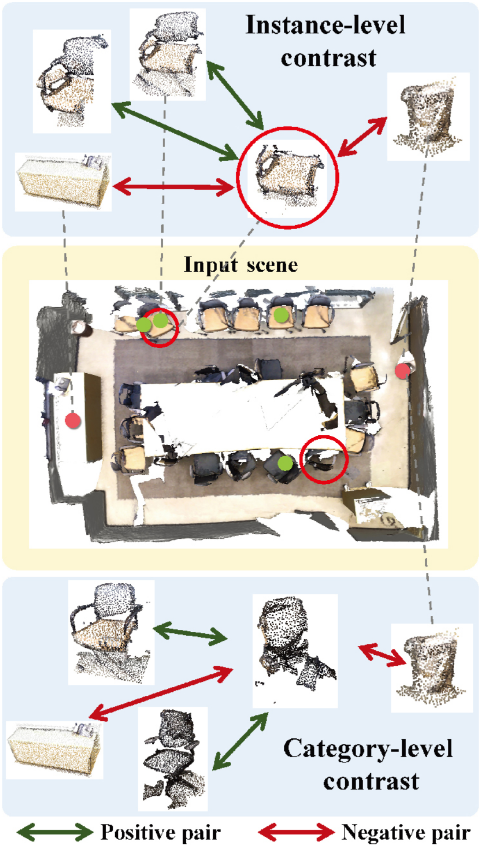


Fig. 1 Main idea of *EFECL*. In the figure's top, the proposal usually contains part of the objects which is difficult for the detector to recognize. Our instance-level contrast (ICL) module computes the contrasts of the same instance and different objects to provide the instance-aware characteristics. In addition, objects in the same category usually differ in structure. The instance-aware contrasts are limited. Therefore, the category-level contrast (CCL) module extracts the common pattern of the category for the incomplete object by computing the contrast of objects in the same category and objects from different categories.

multiple views are also susceptible to incompleteness. ProposalContrast aims to maximize the agreement of feature embeddings between two differently augmented views of the same data instance while simultaneously minimizing the agreement between different instances.

Inspired by the method, we take advantage of contrastive learning to learn the 3D representations for 3D detection. While ProposalContrast also enhances the proposal representations by sharpening the discriminativeness of object instances, it primarily emphasizes feature alignment among distinct views of the instance for LiDAR-based 3D object detection in outdoor scenarios like Waymo Open Dataset [15] and KITTI. Our method emphasizes feature alignments among proposals originating from the same instance within scanned indoor scenes. Consequently, our approaches do not require data augmentation, whereas ProposalContrast incurs additional time costs for data augmentation and corresponding computations. Additionally, our method differs from GroupFree3D, which selects proposals based on high objectness. The fundamental concept of our approach revolves around aligning both complete and incomplete/noisy proposals utilizing contrastive learning to effectively enhance feature encoding.

Having the proposals belonging to the same instance which are not only incomplete but also structurally unclear and contain much noise, we compute the contrasts of the proposals from the same instance to align the features by *instance-level contrastive learning (ICL)* module. The proposal representations are thus encouraged to gather instance-aware properties. However, objects in indoor scenes usually have the same semantic category but differ in shape and structure. The contrasts of instance have a limit in aligning the features within the category. Therefore, we introduce the *category-level contrastive learning (CCL)* module to capture inherent object invariance within the same category and semantic-aware common patterns of the category. Contrasts in the same category reserve the intrinsic property and thus learn the features by pulling the proposals with the same category label close and pushing the proposals in different categories apart. Both of the contrasts help the detector align the features and thus facilitate the semantic recognition of objects.

Additionally, in order to extract more comprehensive and informative features for contrastive learning, we introduce an *Enhanced Feature Aggregation Module* to aggregate the features of the proposal. The module computes the max-pooled features and uniform features to preserve both crucial and general information. Therefore, the module can provide more detailed and richer geometric and semantic features through feature aggregation.

In summary, the contributions of this paper include:

- We propose a contrastive learning framework named EFECL for indoor 3D object detection that does not rely on data augmentation. The method computes instance-level and category-level contrasts to facilitate feature alignment of proposals. Based on the two different contrasts, the approach encourages the network to capture instance-specific characteristics and semantic-aware common patterns.

- We introduce an *Enhanced Feature Aggregation Module* to aggregate informative features for proposals.
- Our method is straightforward yet effective. The proposed method achieves improvements of 12.3% and 7.3% in terms of mAP@0.5 on the ScanNet V2 and SUN RGB-D datasets, respectively.

2 Related work

Point cloud based indoor 3D object detection. Directly detecting 3D objects from point clouds poses challenges due to their sparse, unordered, and irregular point distribution. Recent advancements in 3D object detection can be categorized into two main approaches: voxel-based or grid projection methods [6, 9, 16–20] and point-based methods [4, 5, 7, 8, 10, 11, 21–23]. Voxel-based or grid projection methods are commonly employed in outdoor autonomous driving scenarios. They involve projecting 3D volumes onto 2D grids to detect bird's-eye view (BEV) bounding boxes or converting points into voxels and utilizing 3D ConvNets for 3D box generation. However, these projection/voxelization-based methods all suffer from large memory and computational cost.

After PointNet [24] and PointNet++ [25] having been proposed to learn features of point clouds, point-based methods are widely used to process point clouds directly and predict 3D bounding boxes in indoor scenes. Most of these methods assign a group of points to each object candidate (proposal) and then compute object features from each point group.

VoteNet is a point-based method that first groups the points to each object candidate (proposal) according to their voted center and extracts the object features from the groups. The approach with high performance and few computational costs has achieved great success. Consequently, lots of follow-up works [5, 8, 10–12, 26, 27] have been proposed. MLCVNet incorporates multi-level contextual information for voting and classification. BRNet back-traces the representative points from the vote centers and also revisits complementary seed points around these generated points to capture the fine local structural features. GroupFree3D computes the feature of an object from all the points with the help of an attention mechanism in the Transformers [28] to generate more accurate object detection results. DisARM [10] introduces a plug-and-play module for most detection methods which improves the performance of detection by encoding the weighted relations between objects and relation anchors as context information.

Most of these point-based methods deliver impressive results but are constrained by their design. While these methods strive to generate high-quality proposals or extract high-level features for proposals, they continue to encounter challenges with incomplete proposals. Therefore, we propose the *EFECL* framework for 3D object detection, which leverages contrast learning at the instance and category levels to align proposal features.

Contrastive learning in 3D object detection. Self-supervised learning (*SSL*) has gained popularity in various vision tasks as it enables the learning of expressive feature representations without manual annotations. Consequently, contrastive learning-based SSL algorithms [14, 29–33] have demonstrated impressive results across a wide range of downstream tasks. Lots of 2D image tasks benefit from the availability of free supervision signals derived from the data itself, enabling representations in tasks such as video representations [34], object detection [35–42], image generation [43], scene boundary detection [44], and so on.

However, the extent of its usefulness in 3D point cloud understanding remains largely unexplored. Recently, contrastive learning has emerged as a successful approach for learning 3D feature representation. CrossPoint [45] facilitates establishing a correspondence between 3D and 2D representations of objects by employing cross-modal contrastive learning to maximize the agreement between point clouds and 2D images for 3D object classification.

The learned representations in 3D detection also show excellent performance. PointContrast [46] proposes a self-supervised method to build representations of scene-level point clouds which relies on the complete 3D construction of a scene with point-wise correspondences between the different views of a point cloud. Inspired by PointContrast, Contrastive Scene Contexts [47] explores data-efficient learning by making use of both point-level correspondences and spatial context contrasts in a scene for limited data or supervision. FAC [48] constructs region-level contrast to enhance the local coherence and better foreground awareness in the learned representations for segmentation tasks. RondomRooms [49] learns the 3D scene representation only by applying object-level contrastive learning on two random scenes generated from the synthetic objects to improve the performance of detection. ProposalConstrast devises a pre-training framework for LiDAR-based detection which sharps the discriminativeness of proposals across objects and clusters in different views.

Different from objects in outdoor autonomous driving scenarios, instances in indoor scenes exhibit diverse shapes and complex structures, and they are densely arranged. The presence of similar and incomplete objects frequently leads to false detections. Moreover, the discriminability of individual instances is limited. To address these issues, we enhance feature encoding by minimizing feature agreement between instances and categories through the utilization of contrastive learning.

3 Method

3.1 Overview

In this section, we introduce the proposed *EFECL* in detail. The overall pipeline is shown in Fig. 2. Firstly, we briefly describe our baseline 3D object detection model and the detection tasks in Section 3.2. Then we propose an *Enhanced Feature Aggregation Module* to compute informative and general features for proposals in Section 3.3. Subsequently, we introduce two different contrastive learning components to explore the instance-level discrimination (Section 3.4) and category-level discrimination (Section 3.5) to capture the instance-aware properties and semantic-aware patterns, respectively. Finally, the joint optimization of detection and contrastive learning is formulated in Section 3.6.

3.2 Preliminaries

Inputs and goals. The input is a point cloud $\mathcal{X} \in \mathbb{R}^{N\times 3}$ of size $N \times 3$ with 3D coordinate for N points, where each point is described as $\boldsymbol{x}_i = [x_i, y_i, z_i]^{\mathrm{T}}, i \in \{0, \cdots, N\}$. The detection goal is to produce a set of 3D (oriented) bounding boxes $\mathcal{B}$ with categorization scores to cover all ground-truth objects. Each box $b \in \mathcal{B}$ is associated with a category label $l_b \in \mathcal{C}$, a center $\boldsymbol{c}_b = [c_b^x, c_b^y, c_b^z]^{\mathrm{T}} \in \mathbb{R}^3$ in a world coordinate

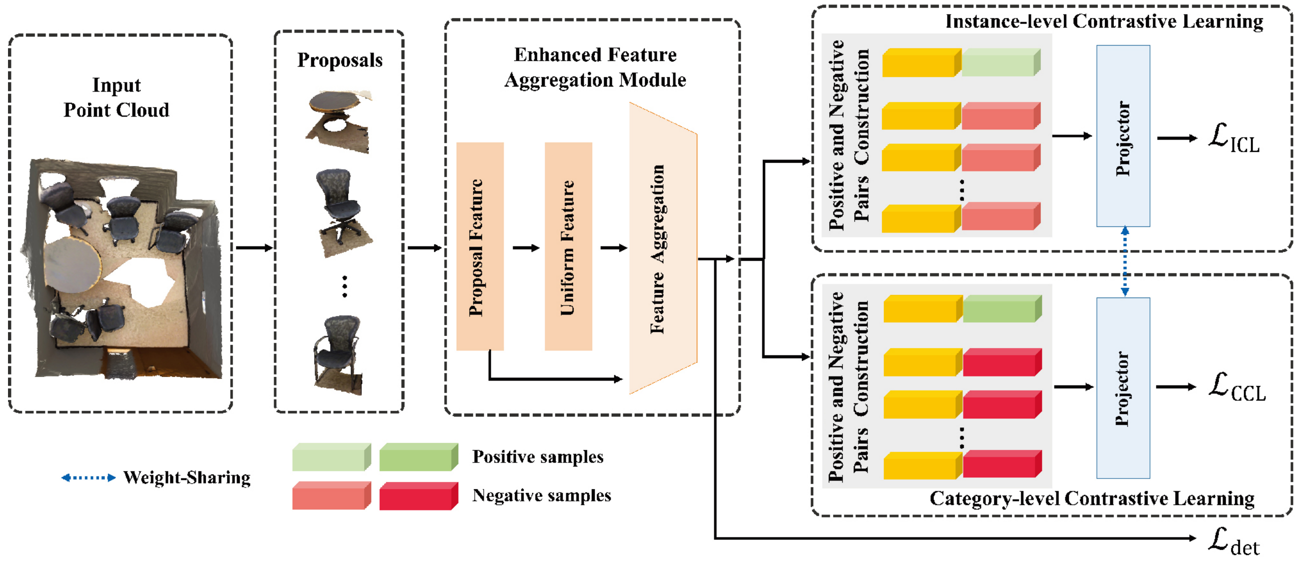

Input
Point Cloud
Proposals
Enhanced Feature
Aggregation Module
Proposal Feature
Uniform Feature
Feature Aggregation
Instance-level Contrastive Learning
Positive and Negative Pairs Construction
Projector
$\mathcal{L}_{\mathrm{ICL}}$
Positive and Negative Pairs Construction
Projector
$\mathcal{L}_{\mathrm{CCL}}$
Category-level Contrastive Learning
$\mathcal{L}_{\mathrm{det}}$
Weight-Sharing
Positive samples
Negative samples

Fig. 2 Framework of EFECL. Taking proposals generated by the backbone as input, we first compute the informative and general features for proposals by *Enhanced Feature Aggregation Module*. The module consists of a uniform feature extractor and feature aggregation function. For each of proposals, we compute the $\mathcal{L}_{\mathrm{ICL}}$ loss and $\mathcal{L}_{\mathrm{CCL}}$ loss. Specifically, we construct positive pairs from the same instance and negative pairs from different objects for ICL to compute contrast and align the features of proposals. At the same time, we construct positive pairs within the same category and negative pairs from different classes for CCL to compute contrast which captures the common pattern of the category. At last, the two different level contrastive learning and detection tasks are together optimized to help the network better understand and recognize the objects.

system, the size of bounding box $\boldsymbol{s}_b = [s_b^x, s_b^y, s_b^z]^{\mathrm{T}} \in \mathbb{R}^3$, and an orientation angle θ_b in the xy-plane of the same world coordinate system.

Proposal generation. Given the input point cloud, we adopt VoteNet as the baseline detection method. Firstly, The approach leverages PointNet++ as the backbone to sample seeds and extract high-dimensional features for the seeds. VoteNet then takes the seed points with extracted features as input to the voting module to regress object centers which simulate the Hough Voting procedure. Clusters are then generated by grouping votes around the cluster centers to form object candidates (proposals). To compute the feature of the proposal, votes from each cluster are processed by an MLP before being max-pooled to a single feature vector and passed to another MLP where information from different votes is further combined. As a result, each proposal in $\{\boldsymbol{p}_i\}_{i=1}^M$ consists of its geometric position $\boldsymbol{z}_i \in \mathbb{R}^3$ in the 3D space and extracted feature $\boldsymbol{f}_i \in \mathbb{R}^C$.

At last, proposals are leveraged to generate 3D bounding boxes and be classified through detection head as Eq. (1), where σ is a detection function implemented by utilizing multi-layer perceptrons (MLP).

$$\{l_i, \boldsymbol{c}_i, \boldsymbol{s}_i, \theta_i\} = \sigma(\boldsymbol{p}_i), \quad \boldsymbol{p}_i = [\boldsymbol{z}_i, \boldsymbol{f}_i]^{\mathrm{T}} \tag{1}$$

Classification and box regression. The loss $\mathcal{L}_{\mathrm{det}}$ in the detection head consists of objectness, bounding box estimation, and semantic classification losses. Objectness loss $\mathcal{L}_{\mathrm{obj}}$ is supervised for the proposals that are located either close to a ground truth object center. $\mathcal{L}_{\mathrm{reg}}$ decouples the box loss to center regression, heading angle estimation, and box size estimation. For semantic classification, VoteNet uses the standard cross-entropy loss. After that, a 3D IoU procedure is operated on the estimated boxes to compute the final detection results.

3.3 Enhanced feature aggregation module for proposal

Although VoteNet has achieved great success in 3D detection by processing point clouds directly and outputting the bounding boxes of the objects. The features of proposals extracted by VoteNet are coarse and lack general information on surrounding points. As mentioned in Section 3.2, VoteNet conducts a max-pooling operation on the votes to aggregate the features to retain remarkable signals. However, the points in indoor scenes usually contain much noise. As a result, some votes in a cluster can be unessential or belong to other objects, and then the max-pooling operation collects inappropriate information for the proposal. The feature aggregated with this inappropriate information makes it more difficult for the network to understand the object. To extract more general proposal representations, we leverage an *Enhanced Feature Aggregation Module* (EFAM) to capture the more uniform features and aggregate informative features for proposals.

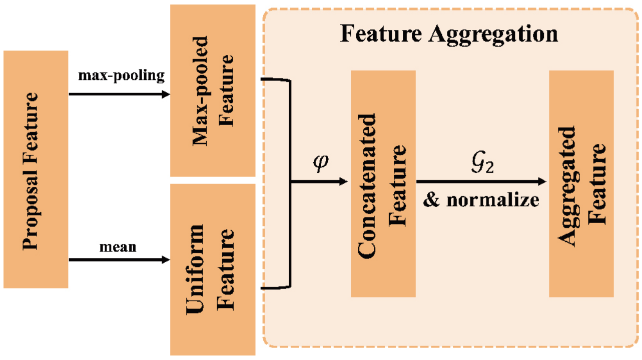


Fig. 3 EFAM architecture. To capture informative features for proposals, we compute the max-pooled feature $\boldsymbol{f}_i$ and uniform feature $\boldsymbol{f}_i'$ through max-pooling and mean operations. Subsequently, the features are concatenated by adopting the function φ. Finally, the aggregated feature $\boldsymbol{f}_{\mathrm{aggr}}$ is obtained through the MLP network $\mathcal{G}_2$ and a normalization operation.

We take the votes generated by VoteNet as input. A vote cluster is denoted as $\mathcal{C} = \{\boldsymbol{v}_q\}$ with $q = 1, \cdots, v$, where each vote $\boldsymbol{v}_q$ consists of its location information and feature. Different from VoteNet, we conduct mean operations on votes within the cluster to compute a uniform feature that represents the average distribution of votes in terms of geometric and feature space. The uniformed feature $\boldsymbol{f}_i'$ of proposal i is obtained by Eq. (2), where $\mathcal{G}_1$ is a perception function given by an MLP network. Note that the uniform feature is used to enrich the feature representations, which provides the general information of the proposal.

$$\boldsymbol{f}_i' = \underset{q=1,\cdots,v}{\mathrm{mean}} \{\mathcal{G}_1(\boldsymbol{v}_q)\} \tag{2}$$

Therefore, we aggregate the informative feature by concatenating the max-pooled feature $\boldsymbol{f}_i$ and the uniform feature $\boldsymbol{f}_i'$ together and projecting the concatenated feature to l_2-normalized embedding space as Eq. (3) showing, where φ is a function which concatenates the two feature in channel-wise, and $\mathcal{G}_2$ aggregates the features which are given by an MLP network. As a result, the aggregated informative feature of proposal $\boldsymbol{p}_i$ is computed by $\boldsymbol{f}_{\text{aggr}}$.

$$\boldsymbol{f}_{\text{aggr}} = \frac{\mathcal{G}_2(\varphi(\boldsymbol{f}_i, \boldsymbol{f}_i'))}{\|\,\mathcal{G}_2(\varphi(\boldsymbol{f}_i, \boldsymbol{f}_i'))\,\|} \tag{3}$$

3.4 Instance-level contrastive learning

Lots of proposals are generated during the detection process, exhibiting a range of qualities. Some proposals contain significant noise points originating from the wall, clutter, and floor, while others capture only a portion of the object or incorporate points from multiple objects. In an effort to produce high-quality proposals, many methods employ complex network architectures and additional supervision signals. However, these approaches also inevitably generate incomplete proposals.

Therefore, we propose instance-level contrastive learning (ICL) and category-level contrastive learning (CCL) to leverage contrasts at different levels for aligning proposal features and enhancing feature encoding. In this section, we first introduce ICL. The key insight is that the network may generate multiple proposals for an object, some of which may only contain partial patches of the object. These incomplete proposals with partial structures are similar to other objects, which can confuse the detector. For instance, a proposal that only includes the lower part of a *chair* may resemble a *table*. Figure 4(a) provides a visualization of this example. To address this, we associate each proposal with positive proposals from the same instance, thereby providing additional information about the object to facilitate feature alignment. Additionally, we pair the proposal with negative proposals from other categories to compute discrimination, enabling the detector to differentiate the incomplete proposal from other similar objects.

Positive and negative pairs construction. Specifically, given the proposals with their features, we try to enforce the features of positive samples to be close and the features of negative samples to be distant. Firstly, we construct one positive pair and N negative pairs for each proposal. Note that we follow Ref. [50] and only consider one positive pair. For proposal $\boldsymbol{p}_i$, we sample the proposal that is closest in 3D space and has the same category label to form the positive set $\mathcal{P}_i^{\text{pos}} = \{\boldsymbol{p}_i^+\}$. The sampled positive $\boldsymbol{p}_i^+$ and the proposal $\boldsymbol{p}_i$ belong to the same instance. We then take ones with different category labels as negative set $\mathcal{P}_i^{\text{neg}} = \{\boldsymbol{p}_{i_n}^-\}^N$. The computation of the pairs is detailed in Algorithm 1.

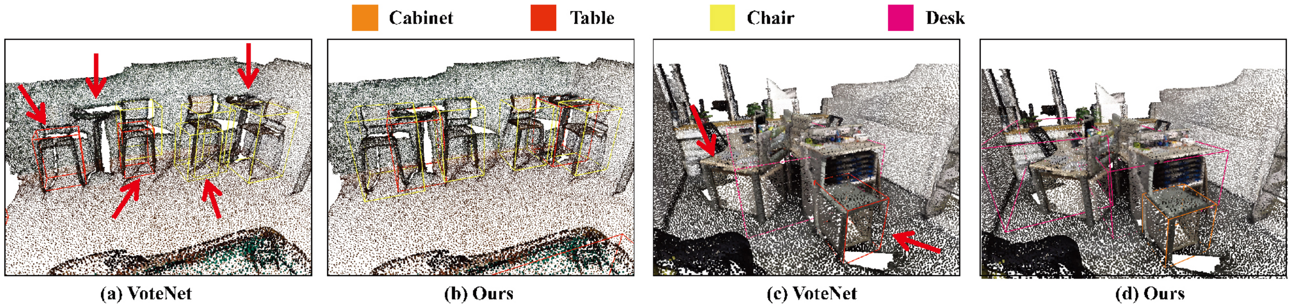
Feature projection. After constructing the positive and negative pairs, we then adopt a projection layer to project all proposal pairs to l_2-normalized embedding space. As a result, the projected positive set $\mathcal{D}_i^+$ and projected negative set $\mathcal{D}_i^-$ are calculated in Eq. (4) and Eq. (5).

$$\mathcal{D}_i^+ = \left\{ \frac{g_{\text{proj}}(\boldsymbol{p}_i^+)}{\|\,g_{\text{proj}}(\boldsymbol{p}_i^+)\,\|} \right\}, \quad \boldsymbol{p}_i^+ \in \mathcal{P}_i^{\text{pos}} \tag{4}$$

$$\mathcal{D}_i^- = \left\{ \frac{g_{\text{proj}}(\boldsymbol{p}_{i_n}^-)}{\|\,g_{\text{proj}}(\boldsymbol{p}_{i_n}^-)\,\|} \right\}, \quad \boldsymbol{p}_{i_n}^- \in \mathcal{P}_i^{\text{neg}} \tag{5}$$

Here g_{proj} is an MLP network consisting of linear transformation layers and activation function.

ICL loss. At last, the ICL loss is designed in the form of the InfoNCE loss [50]. As shown in Eq. (6), where $\boldsymbol{y}_i$, $\boldsymbol{y}_i^+$, $\boldsymbol{y}'$ denote the projected feature



(a) VoteNet (b) Ours (c) VoteNet (d) Ours

Fig. 4 Illustration of the importance of ICL and CCL. We visualize the detection results of VoteNet and EFECL on the ScanNet V2 dataset. The proposals containing part of the *chair* in (a) are mistaken as *table* by VoteNet. With the help of ICL which captures the instance-specific characteristics and provides richer information about the objects by instance-aware contrasts, the network recognizes the proposals and regresses the bounding of boxes accurately as shown in (b). In addition, the objects with incomplete structures, e.g., *tables* in (a) and *desk* in (c), are hard to be detected. Our method can detect objects correctly as well. Note that the *cabinet* in (c) with a small size which is different from the other *cabinets* is similar to *table* which confuses VoteNet. However, our approach captures the semantic-aware common patterns of the *cabinet* category for the object. As a result, we recognize the cabinet accurately in (d).

Algorithm 1 General framework of EFECL

```
**Input**: propoals $\mathcal{P} = \{\boldsymbol{p}_i\}^M$.
**for** $i \in \{1, \cdots, M\}$ **do**
    //Construct the pos/neg pairs for ICL
    sample instance aware positives $\mathcal{P}_i^{\text{pos}} = \{\boldsymbol{p}_i^+\}$:
    $\boldsymbol{p}_i^+ = (\underset{\boldsymbol{p}^+ \in \mathcal{P}, \boldsymbol{p}^+ \neq \boldsymbol{p}_i}{\arg\min} \text{dist}(\boldsymbol{p}_i, \boldsymbol{p}^+)) \vee (l_i == l^+)$
    sample negatives $\mathcal{P}_i^{\text{neg}} = \{\boldsymbol{p}_{i_n}^-\}^N$:
    $\{\boldsymbol{p}_{i_n}^-\} = \{(\underset{\boldsymbol{p}_{i_n}^- \in \mathcal{P} - \{\boldsymbol{p}_i, \boldsymbol{p}_i^+\}}{\text{random}} \boldsymbol{p}_{i_n}^-) \vee (l_i \neq l_n)\}$
    //projection
    $\boldsymbol{y}_i = \text{norm}(g_{\text{proj}}(\boldsymbol{p}_i))$
    $\mathcal{D}_i^+ = \{\boldsymbol{y}_i^+\} = \{\text{norm}(g_{\text{proj}}(\boldsymbol{p}_i^+))\}$
    $\mathcal{D}_i^- = \{\boldsymbol{y}_{i_n}^-\}^N = \{\text{norm}(g_{\text{proj}}(\boldsymbol{p}_{i_n}^-))\}$
    //Construct the pos/neg pairs for CCL
    sample category aware positives $\mathcal{P}_i^{\text{pos}'} = \{\boldsymbol{p}_i^{+'}\}$:
    $\boldsymbol{p}_i^{+'} = (\underset{\boldsymbol{p}^{+'} \in \mathcal{P}, \boldsymbol{p}^{+'} \neq \boldsymbol{p}_i}{\arg\max} \text{dist}(\boldsymbol{p}_i, \boldsymbol{p}^{+'})) \vee (l_i == l^{+'})$
    sample negatives $\mathcal{P}_i^{\text{neg}'} = \{\boldsymbol{p}_{i_n}^{-'}\}^{N'}$:
    $\{\boldsymbol{p}_{i_n}^{-'}\} = \{(\underset{\boldsymbol{p}_{i_n}^{-'} \in \mathcal{P} - \{\boldsymbol{p}_i, \boldsymbol{p}_i^{+'}\}}{\text{random}} \boldsymbol{p}_{i_n}^{-'}) \vee (l_i \neq l_{n'})\}$
    //projection
    $\mathcal{D}_i^{+'} = \{\boldsymbol{y}_i^{+'}\} = \{\text{norm}(g_{\text{proj}}(\boldsymbol{p}_i^{+'}))\}$
    $\mathcal{D}_i^{-'} = \{\boldsymbol{y}_{i_n}^{-'}\}^{N'} = \{\text{norm}(g_{\text{proj}}(\boldsymbol{p}_{i_n}^{-'}))\}$
**end for**
compute ICL loss: $\mathcal{L}_{\text{ICL}}$;
compute CCL loss: $\mathcal{L}_{\text{CCL}}$.
```

of proposal $\boldsymbol{p}_i$, projected feature of corresponding positive proposal $\boldsymbol{p}_i^+$, and projected feature of proposal in the union of D_i^+ and D_i^-, respectively. $\mathcal{D}$ set collects the projected feature of proposal $\boldsymbol{p}_i$ and M indicates the number of proposals. τ is a temperature hyper-parameter.

$$\mathcal{L}_{\text{ICL}} = -\frac{1}{M} \sum_{\boldsymbol{y}_i \in \mathcal{D}} \log \frac{\exp(\boldsymbol{y}_i \cdot \boldsymbol{y}_i^+ / \tau)}{\sum_{\boldsymbol{y}' \in \mathcal{D}_i^+ \cup \mathcal{D}_i^-} \exp(\boldsymbol{y}_i \cdot \boldsymbol{y}' / \tau)} \tag{6}$$

3.5 Category-level contrastive learning

We argue that contrasts between proposals of the same category are also crucial. Different from objects in some outdoor scenes, objects in indoor scenes exhibit diversity and complexity. Many objects share the same category label but differ in terms of their structures, size, and component details. These distinctive characteristics of indoor objects pose challenges for the detection network. For instance, as illustrated in Fig. 4(c), a small-sized *cabinet* can be easily misclassified as a *table*. The limited contrasts between instances of the same object fail to capture the general rules within the category. Furthermore, it is challenging for the network to directly learn the semantic-aware common patterns.

Therefore, to further release the power of the category-level information, we propose category-level contrastive learning (CCL) module by which the object representations from the same category are aligned and the features from different categories are pushed away.

Different pairs construction. Similar to the ICL, we first construct one positive pair and N' negative pairs. The difference is that we form the positive set $\mathcal{P}_i^{\text{pos}\prime} = \{\boldsymbol{p}_i^{+'}\}$ for proposal $\boldsymbol{p}_i$ by sampling the proposal with a same category label and largest distance in 3D space. The operation ensures that the two proposals do not come from the same instance. After that, we take the proposals with different category labels as negatives $\mathcal{P}_i^{\text{neg}\prime} = \{\boldsymbol{p}_{i_n}^{-'}\}^{N'}$. The sampled positive and negative pairs thus provide the category-level information for the network to explore the semantic-aware common pattern within the same class and discrimination between different categories. Projection and normalization are then conducted on the features of positive proposals and negative proposals to construct the sets $\mathcal{D}_i^{+'}$ and $\mathcal{D}_i^{-'}$, respectively. Note that the weights in the projection layer are shared between ICL and CCL.

CCL loss. The CCL loss is formulated as Eq. (7), where $\boldsymbol{y}_i^{+'}$, $\boldsymbol{y}''$ denote the projected features of the corresponding positive proposal $\boldsymbol{p}_i^{+'}$ and the projected feature of proposal in the union of $\mathcal{D}_i^{+'}$ and $\mathcal{D}_i^{-'}$. τ' is the temperature hyper-parameter for category-level contrasts.

$$\mathcal{L}_{\text{CCL}} = -\frac{1}{M} \sum_{\boldsymbol{y}_i \in \mathcal{D}} \log \frac{\exp(\boldsymbol{y}_i \cdot \boldsymbol{y}_i^{+'} / \tau')}{\sum_{\boldsymbol{y}'' \in \mathcal{D}_i^{+'} \cup \mathcal{D}_i^{-'}} \exp(\boldsymbol{y}_i \cdot \boldsymbol{y}'' / \tau')} \tag{7}$$

3.6 Joint optimization of EFECL

The property of contrast learning, together with the motivation to help the detector better align the features of the same instance or same category, inspires us to perform contrast learning simultaneously during the detection training procedure. The joint optimization thus helps the network enhance the feature encoding in a fine-tuned way. Therefore, the overall object function is defined by detection target and contrastive learning considering both instance-level contrast and category-level distinction. The total loss $\mathcal{L}$ is defined in Eq. (8), where α and β are the balancing coefficients, respectively. We show the analysis of the effectiveness of the optimization in Section 5.3.

$$\mathcal{L} = \mathcal{L}_{\text{det}} + \alpha\mathcal{L}_{\text{ICL}} + \beta\mathcal{L}_{\text{CCL}} \tag{8}$$

4 Implementation details

4.1 EFECL's main learning algorithm

We have described the purpose, methodology, and formulation of ICL and CCL in Section 3. To more intuitively elaborate the design, we summarize the whole procedure in Algorithm 1 which contains positive and negative pairs construction, feature projection, and loss computation for each contrast learning task.

4.2 Network architecture and training details

To evaluate the efficacy and versatility of our approach, we investigate the performance of EFECL on VoteNet as 3D detection backbone architecture. We take the $M = 256$ output proposals of VoteNet with 128-dimensional features as the input of ICL and CCL. The projection g_{proj} is realized with two linear layers output size of 256, 256, and a ReLU as activation function. The number of negative proposals N and temperature parameter τ in ICL are set to 20 and 0.2, respectively. Hyper-parameter N' and τ' in CCL are set to 15 and 0.3, respectively. The coefficients α and β in Eq. (8) are set to 1 and 0.1, respectively.

We first train VoteNet with our EFAM to generate informative features of proposals for 36 epochs. The initial learning rate is 0.008. The decay steps are 24 and 32. After that, we fine-tune the entire network by objective function Eq. (8) with the pre-trained model equipped with EFAM for 80 epochs. The initial learning rate is 0.008. The decay steps are 56 and 68. EFECL is optimized by adopting the Adam [51] optimizer with the batch size of 8. We implement our method on MMDetection3D [52] which is an open-source 3D object detection toolbox with one NVIDIA TITAN V GPU.

5 Experiments

5.1 Dataset and metrics

ScanNet V2 dataset. We utilize the widely adopted ScanNet V2 dataset, which offers extensive 3D indoor scenes. ScanNet V2 is a dataset consisting of RGB-D video recordings capturing indoor scenes, and it provides rich annotations of 3D reconstructed meshes. The dataset comprises approximately 1.5k scans that are annotated with both semantic segmentation and object instance labels, covering a total of 18 categories. Point clouds are sampled from the reconstructed meshes following the approach introduced in Ref. [4].

SUN RGB-D dataset. The SUN RGB-D dataset [2] is a well-known public dataset specifically designed for single-view RGB-D scene understanding tasks. It comprises approximately 5000 training images that are annotated with oriented 3D bounding boxes and semantic labels across 10 categories. To process the point data in our method, we adopt the approach presented in Ref. [4] to convert the depth images into point clouds by utilizing the camera parameters provided with the dataset. Furthermore, we assess the performance of our method on this challenging dataset, known for its significant occlusion challenges.

Metrics. Average precision is employed as the evaluation metric to assess the accuracy of the detected object bounding boxes against the ground truth bounding boxes. In our experiments, we utilize two IoU thresholds: 0.5 and 0.25. The mean average precision (mAP) is calculated as the macro-average of the average precision values across all test categories. The mAP values computed using the two thresholds are referred to as mAP@0.5 and mAP@0.25, respectively.

5.2 Comparisons

In this section, we evaluate our method with previous approaches on the ScanNet V2 dataset in Table 1 to demonstrate its effectiveness, such as VoteNet and its

Table 1 3D object detection results on ScanNet V2 dataset. Notations: Ours[1] indicates that we only apply the $\mathcal{L}_{\text{CCL}}$ and $\mathcal{L}_{\text{ICL}}$ to baseline method. * denotes the average performance of training and test trial in GroupFree3D. We show the results of GroupFree3D as reported in MMDetection3D

Method	mAP@0.25	mAP@0.5
VoteNet	58.7	33.5
PointContrast	59.2	38.0
RandomRooms	61.3	36.2
DepthContrast	64.0	42.9
Ours	**64.3**	**45.8**
BRNet	66.1	50.9
Ours[1] (BRNet)	**66.7**	**51.7**
GroupFree3D (L6, O256)	66.3 (65.7*)	47.8 (47.7*)
Ours[1] (GroupFree3D (L6, O256))	**66.8**	**50.1**
GroupFree3D (L12, O256)	66.6 (66.2*)	48.2 (49.0*)
Ours[1] (GroupFree3D (L12, O256))	**67.4**	**50.5**

successors BRNet and GroupFree3D. To demonstrate the generalization on different indoor scenes, we also show the results on the SUN RGB-D dataset in Table 3.

Quantitative results. The detection results of ScanNet V2 are shown in Table 1. Taking VoteNet as the baseline model, our EFECL achieves 64.3 on mAP@0.25 and 45.8 on mAP@0.5, which is 5.6 and 12.3 higher than the performance of VoteNet without designing complex network architectures and augmenting any additional data. Note that we also compare the performance with PointContrast, RandomRooms, and DepthContrast [53]. The three excellent works also take VoteNet as a baseline method and introduce contrastive learning for detection. EFECL outperforms the approaches on both mAP@0.25 and mAP@0.5 which demonstrates the effectiveness of our two different contrasts.

We then take BRNet and GroupFree3D as the harder baselines which extract more abstract features for proposals, and our approach also obtains 0.6, 0.5 improvements on mAP@0.25 and 0.8, 2.3 improvements on mAP@0.5 respectively. Note that the harder baselines have different proposal generation modules. Therefore, we only equip the baselines with our $\alpha\mathcal{L}_{\text{ICL}} + \beta\mathcal{L}_{\text{CCL}}$ but also improve the performances on them. The larger improvements on mAP@0.5 demonstrate that EFECL helps the detectors to learn the representations of objects more precisely.

As presented in Table 3, our method is evaluated on the SUN RGB-D dataset. In this evaluation, EFECL outperforms the baseline method with improvements of 3.8 in mAP@0.25 and 7.3 in mAP@0.5. These results highlight the effectiveness of our proposed two-level contrastive learning approach, which facilitates feature alignment in incomplete proposals and enhances the distinction between category representations.

Table 8 presents the results for the 10 categories in terms of mAP@0.25 and mAP@0.5. Our method exhibits improved performance in mAP@0.25 for most categories, specifically 9 out of 10. Furthermore, when compared to VoteNet, our method outperforms across all categories in terms of mAP@0.5, demonstrating its superiority. Significantly, we achieve over 10% improvement in mAP@0.5 for the *sofa* category, characterized by larger structures where proposals often contain parts of the objects. These results underscore the ability of EFECL to enhance feature encoding for objects, even in cases where they are incomplete.

Notably, our method works well on both datasets, which indicates its outstanding generalization ability for different detection scenarios with different detectors.

Table 2 3D object detection results on ScanNet V2 dataset with mAP@0.5. Notations: Ours^1 indicates that we apply both $\mathcal{L}_{\text{CCL}}$ and $\mathcal{L}_{\text{ICL}}$ to the baseline method; we utilize bounding boxes to mark the numbers and indicate the categories that each component or combination of components excels at processing

Method	cab	bed	chair	sofa	tabl	door	wind	bkshf	pic	cntr	desk	curt	fridg	showr	toil	sink	bath	ofurn	mAP
VoteNet	8.1	76.1	67.2	68.8	42.4	15.3	6.4	28.0	1.3	9.5	37.5	11.6	27.8	10.0	86.5	16.8	78.9	11.7	33.5
CCL	17.6	78.2	72.7	75.2	50.4	24.7	14.7	38.9	3.8	27.6	44.9	$\boxed{24.2}$	31.6	17.2	$\boxed{87.0}$	$\boxed{36.4}$	89.9	19.8	41.9
ICL	$\boxed{22.1}$	81.1	73.2	78.7	50.8	23.5	15.6	$\boxed{46.2}$	3.9	29.1	44.4	19.8	30.7	$\boxed{33.7}$	82.9	28.4	92.1	21.6	43.2
Ours^1	$\boxed{18.8}$	80.2	72.1	75.9	$\boxed{52.9}$	$\boxed{25.3}$	$\boxed{15.8}$	$\boxed{48.9}$	$\boxed{4.9}$	$\boxed{33.4}$	$\boxed{45.2}$	$\boxed{29.2}$	$\boxed{33.9}$	$\boxed{36.5}$	$\boxed{87.0}$	$\boxed{35.4}$	86.6	$\boxed{23.3}$	44.7

Table 3 EFECL with different components. The first row indicates the baseline method (VoteNet) without our components. We denote the CCL, ICL, and EFAM as the baseline method adopting our $\mathcal{L}_{\text{CCL}}$, $\mathcal{L}_{\text{ICL}}$, and the enhanced feature aggregation module respectively

CCL	ICL	EFAM	ScanNet		SUN RGB-D	
			mAP@0.25	mAP@0.5	mAP@0.25	mAP@0.5
×	×	×	58.7	33.5	57.7	32.9
✓			63.5 (+4.8)	41.9 (+8.4)	60.9 (+3.2)	39.1 (+6.2)
	✓		63.5 (+4.8)	43.2 (+9.7)	61.1 (+3.4)	39.2 (+6.3)
✓	✓		64.2 (+5.5)	44.7 (+11.2)	61.0 (+3.3)	39.6 (+6.7)
		✓	63.6 (+4.9)	43.4 (+9.9)	61.3 (+3.6)	39.6 (+6.7)
✓	✓	✓	**64.3 (+5.6)**	**45.8 (+12.3)**	**61.5 (+3.8)**	**40.2 (+7.3)**

Qualitative results. In Fig. 6 and Fig. 8, we visualize the representative 3D object results from our method and the baseline method (VoteNet) on ScanNet V2 dataset and SUN RGB-D dataset. These results demonstrate that applying our EFECL to the baseline method achieves more reliable detection results with more accurate bounding boxes. The *other furniture* in the first row with the partial structure of objects is mistaken as *cabinet* by VoteNet but recognized correctly by our method. We attribute the success to the instance properties captured by ICL. The patches of the wall in the second row are classified as *door* by VoteNet which does not learn the common pattern of the *door* category, such as the average size and the handle component of the door. EFECL with CCL helps the network deal with incomplete objects by the contrasts from categories. Note that VoteNet even treats the proposal containing noise points in the third row as *cabinet*. All these results prove that our method can help the detector recognize and localize the objects more effectively.

5.3 Ablation study

We conduct extensive ablation experiments to analyze the effectiveness of different components of EFECL. All experiments are trained and evaluated on the ScanNet V2 dataset and take VoteNet as the baseline method.

Analysis on each component of EFECL. We evaluate the contribution of each component in Table 3. The proposed $\mathcal{L}_{\mathrm{CCL}}$ and $\mathcal{L}_{\mathrm{ICL}}$ both help detectors obtain better performance. Although $\mathcal{L}_{\mathrm{ICL}}$ is more effective in improving performance on mAP@0.5, the contrasts provided by it with instance properties are not enough for the objects in the indoor scene. Together with $\mathcal{L}_{\mathrm{CCL}}$, the detector achieves higher performance. The results demonstrate that $\mathcal{L}_{\mathrm{ICL}}$ and $\mathcal{L}_{\mathrm{CCL}}$ provide different concepts of information by instance-level and category-level contrasts.

Compared to VoteNet, our EFAM module also obtains better results on the ScanNet V2 dataset and SUN RGB-D dataset in terms of mAP@0.25 and mAP@0.5. We attribute the success to the EFAM collecting more general features of the proposal which provides richer information than the VoteNet only adopting the MaxPooling function. Applying all the modules of EFECL, we get the best results.

The impact of CCL, ICL, and CCL+ICL on the 18 categories is illustrated in Table 2. CCL significantly improves the performance for *curt*, *toil*, and *sink*. This can be attributed to the inherent challenges faced by categories with limited samples and low-quality point cloud representations, making it difficult for the detector to learn common patterns within these categories. However, CCL effectively provides class-aware information through category-level contrasts. Furthermore, ICL outperforms CCL in terms of performance for *cab*, *bkshf*, and *showr*. These objects are characterized by their large structures, and the proposals typically include parts of the objects. The observed improvements can be attributed to the instance-specific characteristics facilitated by instance-level contrasts. Leveraging both CCL and ICL leads to improved performance in most categories, including *tabl*, *door*, *cntr*, and others. These results highlight the different contrasts provided by ICL and CCL, and the combined utilization of both contrasts enables the network to achieve superior results.

To further analyze why EFECL can help the detector improve its performance, we draw the curves of loss with different detection tasks on the ScanNet V2 dataset in Fig. 5. EFECL performs better in center regression and objectness classification tasks which demonstrates that EFECL helps the detector focus on the target object by exploring the instance-specific characteristics. Good convergence also can be found in semantic classification tasks which indicates that EFECL helps the detector understand and recognize the objects in terms of the contrast from categories. Besides, EFECL not only has an effect in semantic aware tasks but also has effectiveness in box regression tasks.

Hyper-parameters in ICL. The effectiveness of each hyperparameter in $\mathcal{L}_{\mathrm{ICL}}$ is analyzed, as presented in Table 4. Through multiple experiments,

Table 4 Ablation studies of hyper-parameter settings in ICL module. Notations: τ denotes the temperature parameter in Eq. (6) and N denotes the number of negative pairs

Setting	mAP@0.25	mAP@0.5
$\tau = 0.2, N = 10$	62.8	42.5
$\tau = 0.2, N = 12$	63.0	42.5
$\tau = 0.2, N = 15$	63.2	**43.3**
$\tau = 0.2, N = 20$	**63.5**	43.2
$\tau = 0.2, N = 32$	63.1	41.9
$\tau = 0.07, N = 20$	62.9	41.0
$\tau = 0.1, N = 20$	63.3	41.2
$\tau = 0.2, N = 20$	**63.5**	**43.2**
$\tau = 0.3, N = 20$	63.3	42.5
$\tau = 0.5, N = 20$	63.0	42.8

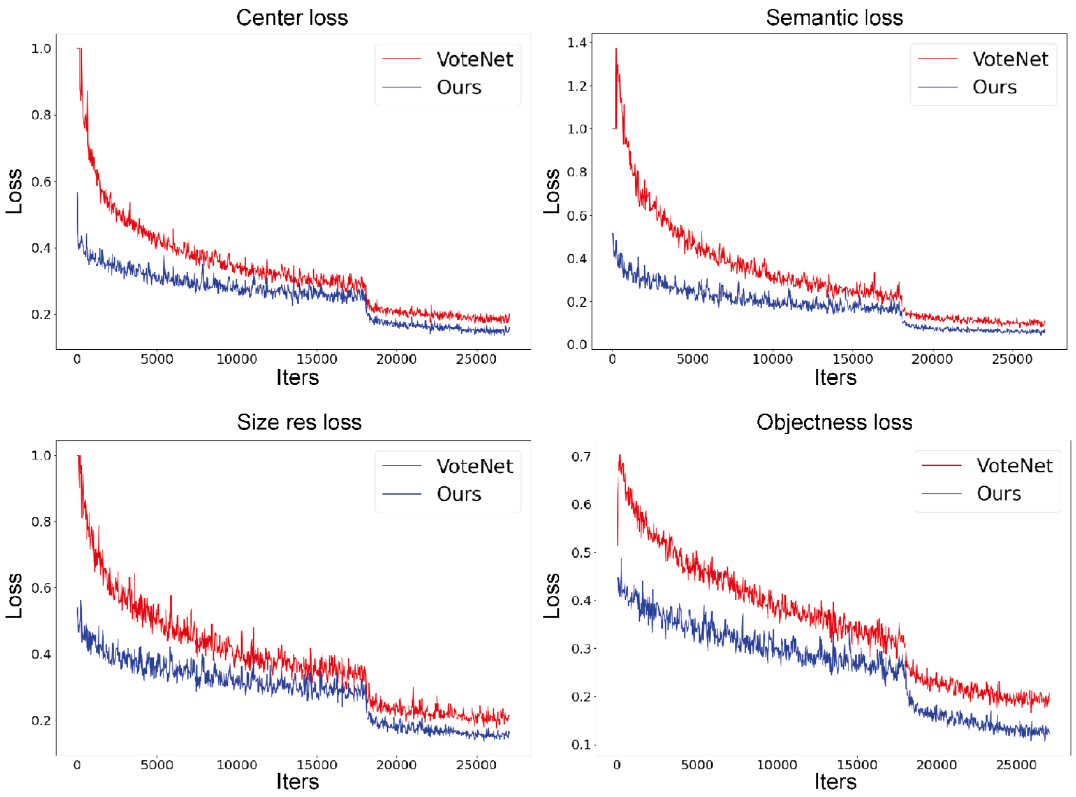


Fig. 5 Loss curves of center regression, semantic category classification, box size regression, and objectness classification tasks for VoteNet and EFECL on ScanNet V2 dataset.

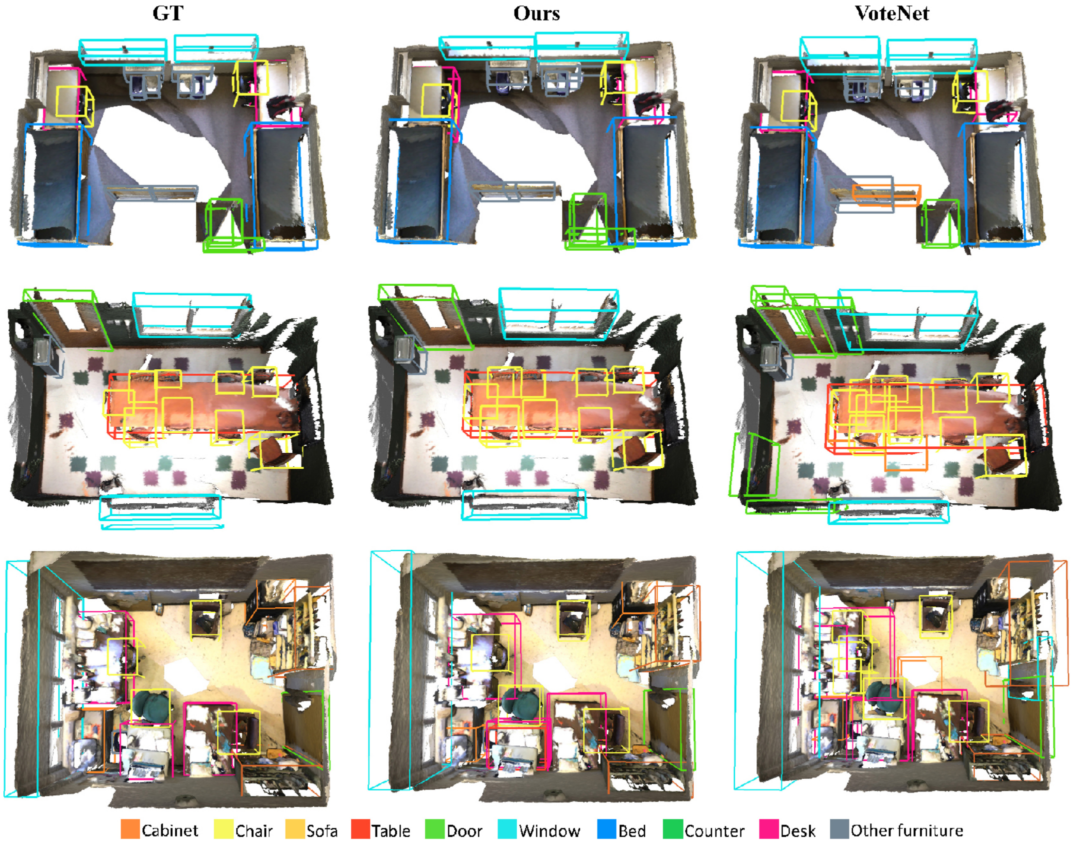


Fig. 6 Qualitative results on ScanNet V2 dataset. The first column is ground truth and the rest columns are detections of our EFECL and VoteNet. Best viewed on screen.

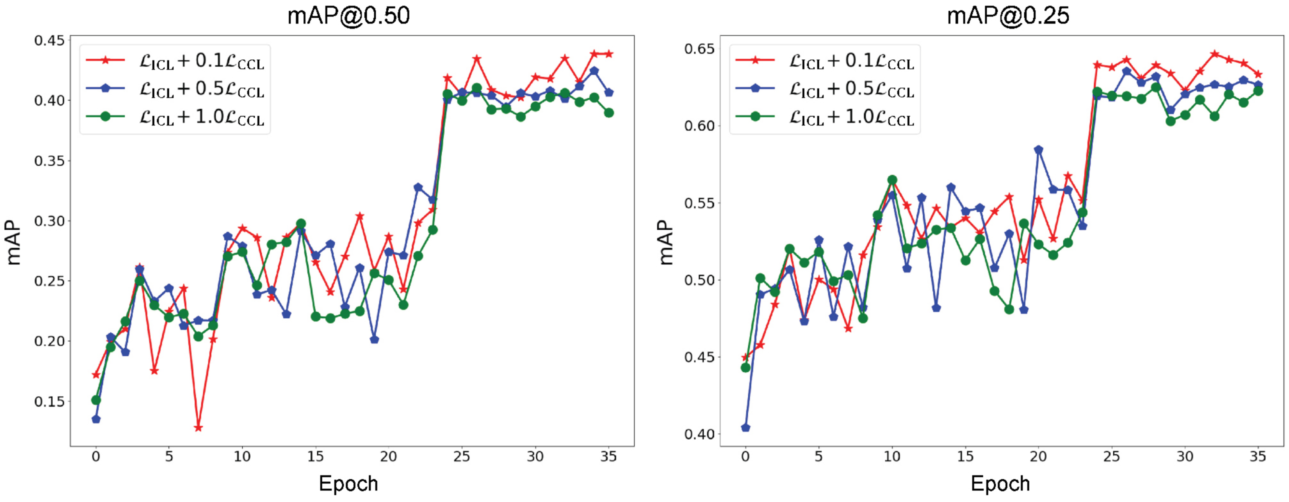


Fig. 7 Mean average precision curves of different weights of $\mathcal{L}_{\text{ICL}}$ and $\mathcal{L}_{\text{CCL}}$ on ScanNet V2 dataset. Notations: We plot the results of evaluation on the validation set after each training epoch.

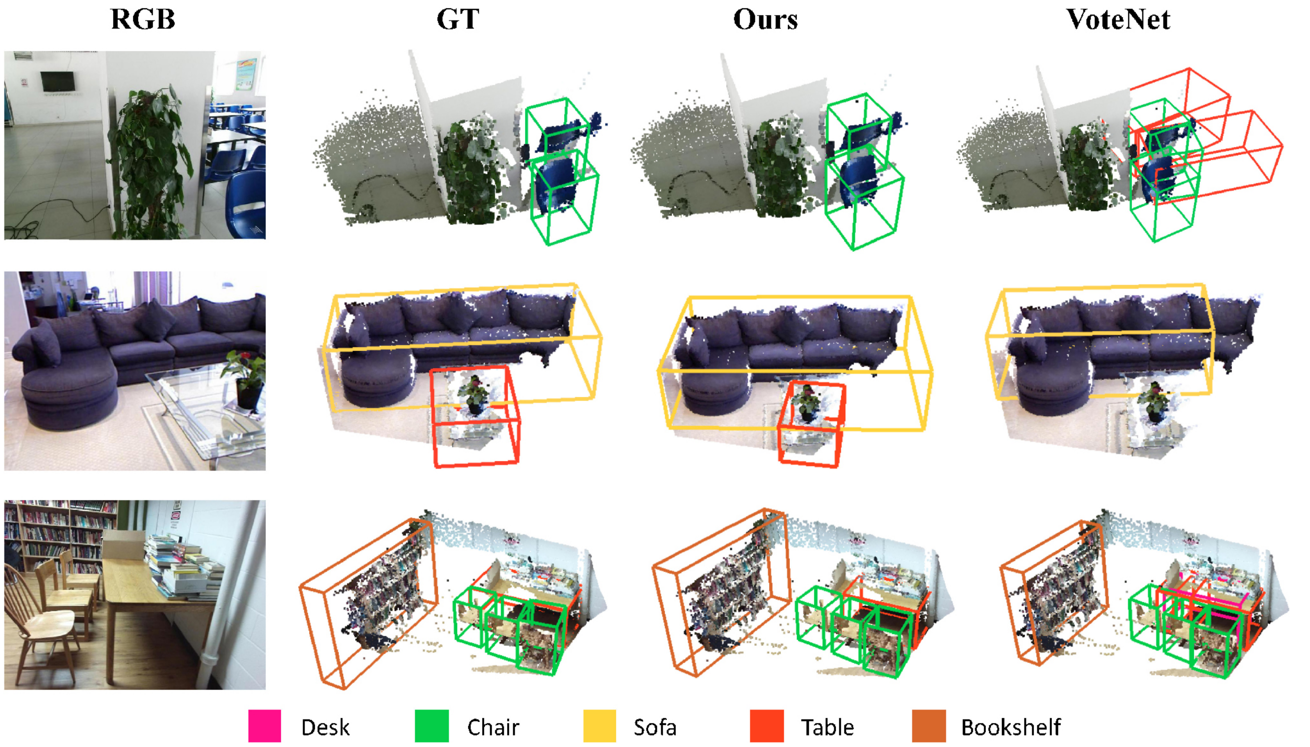


Fig. 8 Qualitative results on SUN RGB-D dataset. The first column is ground truth and the rest columns are detections of different methods. Best viewed on screen.

it is found that the number of negative pairs should not be too small. Instance-level contrast provides strong signals for features and representations, and a smaller number of negatives hinders effective discrimination computation and limits information retention. Furthermore, an excessive number of negative pairs hampers the network's ability to accurately recognize objects. Additionally, the results demonstrate that an excessively high value of the temperature hyperparameter, τ, causes the network to overlook samples that are challenging to distinguish. In the case of instance-level contrast, smaller temperature hyperparameters are necessary to prioritize the most similar and challenging samples, enabling the computation of more robust representations.

Hyper-parameters in CCL. The effectiveness of different numbers of negative pairs for each proposal and various values of the temperature hyperparameter, τ', is demonstrated in Table 5. It is important to note that the provision of more negative samples does not necessarily result in a higher number of contrasts across different categories. On the contrary, an abundance of negative proposals complicates the learning process and leads to confusion for the detector. The role of the temperature hyperparameter, τ', is to regulate the attention given to challenging samples. In the case of category-level contrast, objects from different classes may exhibit similar representations. Consequently, several challenging negative samples may possess potentially related features. Overly compelling feature extractors to separate proposals from such difficult samples would result in a loss of underlying semantic information. Therefore, it is advised not to set the temperature coefficient too small, as it may compromise the ability to effectively capture the distinctions between challenging samples.

Analysis of the weights of different contrasts. As shown in Table 7, we evaluate the different weights of $\mathcal{L}_{\text{ICL}}$ and $\mathcal{L}_{\text{CCL}}$ mentioned in Eq. (8). As shown in Table 2, ICL and CCL have different contributions on different categories and they also can be leveraged together to obtain better results. However, we find that ICL achieves better performance than CCL on the mean average precision of the categories. The result shows that ICL provides principal contrasts and plays a more important role. CCL further improves the performance by computing a different contrast. Therefore, we first set α as 1.0 and observe the effects on different weights of $\mathcal{L}_{\text{CCL}}$. Increasing the value of β, the performances on mAP@0.25 and mAP@0.5 gradually decrease which suggests that much category-aware contrast will distract the network from learning the representations of objects and only attempt to find the difference between the categories.

Table 5 Ablation studies of hyper-parameter settings in CCL module. Notations: τ' denotes the temperature parameter in Eq. (7) and N' denotes the number of negative pairs

Setting	mAP@0.25	mAP@0.5
$\tau' = 0.3, N' = 10$	62.8	40.6
$\tau' = 0.3, N' = 12$	62.1	41.0
$\tau' = 0.3, N' = 15$	**63.5**	**41.9**
$\tau' = 0.3, N' = 20$	63.0	40.5
$\tau' = 0.3, N' = 32$	62.9	40.2
$\tau' = 0.07, N' = 15$	62.2	40.6
$\tau' = 0.1, N' = 15$	61.6	40.0
$\tau' = 0.2, N' = 15$	62.8	41.2
$\tau' = 0.3, N' = 15$	**63.5**	**41.9**
$\tau' = 0.5, N' = 15$	62.2	40.6

Table 6 Efficiency of different components and methods

Module	Params (M)	GFLOPs
EFAM	0.117	0.288
$\alpha\mathcal{L}_{\text{ICL}} + \beta\mathcal{L}_{\text{CCL}}$	0.033	0.0
Method	Params (M)	GFLOPs
VoteNet	0.93	5.78
Ours	1.04 (**+0.11**)	5.79 (**+0.01**)

Table 7 Comparison of efficiency for different weights of $\mathcal{L}_{\text{ICL}}$ loss and $\mathcal{L}_{\text{CCL}}$ loss. Notations: α denotes the weight of $\mathcal{L}_{\text{ICL}}$ and β denotes the weight of $\mathcal{L}_{\text{CCL}}$

Setting	mAP@0.25	mAP@0.5
$\alpha = 1.0, \beta = 0.1$	64.2	**44.7**
$\alpha = 1.0, \beta = 0.5$	62.6	40.6
$\alpha = 1.0, \beta = 1.0$	61.5	40.2
$\alpha = 0.1, \beta = 1.0$	61.6	41.5
$\alpha = 0.5, \beta = 0.5$	61.5	40.4
$\alpha = 0.8, \beta = 0.2$	63.3	43.7
$\alpha = 0.9, \beta = 0.1$	**64.8**	42.5

We also plot the curves of mAP@0.25 and mAP@0.5 during training in Fig. 7, and the curves with setting $\beta = 0.1$ reaching the highest points also prove our conclusion. Therefore, to take advantage of both contrastive losses, we set a lower CCL weight. We also set the values of α and β with the constraint of $\alpha + \beta = 1.0$. Although we obtain the best result on mAP@25 in the last row of the table, the experiment with setting $\alpha = 0.9, \beta = 0.1$ does not perform well on both metrics. Note that we give lower weights to CCL which does not demonstrate the category-aware contrast is useless. On the contrary, the contrast captured by CCL is different from the instance-aware characteristics which is also important for the feature encoding enhancement.

Training parameters and computational complexity. Table 6 presents the efficiency of each component and various methods. To ensure a fair comparison, all experiments are conducted on a single Titan V GPU workstation utilizing the MMDetection3D toolbox. Initially, we evaluate the efficiency of each component in EFECL. EFAM and the two contrastive learning modules have only 0.117M and 0.033M training parameters, respectively. Furthermore, the contrastive learning

Table 8 3D object detection results of 10 categories on SUN RGB-D val dataset with mAP@0.25 and mAP@0.5

Method	bathtub	bed	bkshf	chair	desk	drser	nigtstd	sofa	table	toilet	mAP@0.25
VoteNet	74.4	83.0	28.8	75.3	22.0	29.8	62.2	64.0	47.3	90.1	57.7
Ours	**79.7**	**86.8**	**34.4**	**78.5**	**30.3**	**33.1**	**65.7**	**67.8**	**49.3**	89.2	**61.5**
Method	bathtub	bed	bkshf	chair	desk	drser	nigtstd	sofa	table	toilet	mAP@0.5
VoteNet	47.0	50.1	7.2	53.9	5.3	11.5	40.7	42.4	19.5	59.8	33.7
Ours	**53.3**	**56.8**	**12.8**	**58.6**	**9.5**	**19.7**	**49.9**	**54.1**	**23.2**	**64.6**	**40.2**

modules do not incur any additional computational cost (0 GFLOPs) on the GPU. We also present a comparison of efficiency between the baseline method and EFECL. Our proposed method is highly effective, requiring only a slight increase in training parameters (0.11M) compared to the backbone method (VoteNet) and incurring an extremely small computational cost (0.01 GFLOPs) while achieving significant performance improvements (5.6% and 12.3%). These findings demonstrate that our lightweight modules offer substantial performance enhancements for 3D object detection.

6 Conclusions

This paper presents a novel contrastive learning framework aimed at enhancing the performance of indoor 3D object detection. Unlike previous methods that heavily rely on multi-view data augmentation and solely focus on learning contrasts between identical instances, our approach computes diverse contrasts for proposals without employing any augmentation. Firstly, to extract highly informative proposal features for contrastive learning, we introduce an Enhanced Feature Aggregation Module that combines uniform features and max-pooled features. Subsequently, we compute both instance-level and category-level contrasts for the proposals. The network is guided by these contrasts to align proposal features by learning instance-specific characteristics and semantic-aware common patterns. Our method enables the detector to more accurately recognize incomplete proposals that only contain partial objects and noise points. Experimental evaluations conducted using diverse benchmarks and datasets demonstrate the effectiveness and generalizability of our approach.

Limitation. Our approach is more applicable to objects within indoor scenes. Furthermore, our method captures the common patterns shared among objects of the same category, which exhibit complexity and diversity. However, objects of the same category in certain outdoor scenes, such as autonomous driving scenes, exhibit similar structures and sizes. This similarity poses a challenge for the detector to learn category-level contrasts. For instance, this is evident with *cars*, *pedestrians*, *cyclists*, and other similar objects. In future work, we aim to explore a more comprehensive contrastive learning framework that encompasses both indoor and outdoor scenes.

Availability of data and materials

All datasets included in the manuscript are deposited in publicly available repositories (`https://github.com/ScanNet/ScanNet` and `http://rgbd.cs.princeton.edu/`).

Author contributions

Yao Duan: Methodology, Software, Writing Draft, Visualization, Results Analysis; Renjiao Yi: Methodology, Supervision, Results Analysis; Yuanming Gao: Supervision, Results Analysis; Kai Xu: Methodology, Supervision; Chenyang Zhu: Methodology, Supervision, Results Analysis.

Acknowledgements

We thank Yuqing Lan for visualizing the results. This work is supported in part by the National Key R&D Program of China (2018AAA0102200), National Natural Science Foundation of China (62002375, 62002376, 62132021), Natural Science Foundation of Hunan Province of China (2021RC3071, 2022RC1104, 2021JJ40696), and NUDT Research Grants (ZK22-52).

Declaration of competing interest

The authors have no competing interests to declare that are relevant to the content of this article.

References

[1] Dai, A.; Chang, A. X.; Savva, M.; Halber, M.; Funkhouser, T.; Nießner, M. ScanNet: Richly-annotated 3D reconstructions of indoor scenes. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2432–2443, 2017.

[2] Song, S. R.; Lichtenberg, S. P.; Xiao, J. X. SUN RGB-D: A RGB-D scene understanding benchmark suite. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 567–576, 2015.

[3] Geiger, A.; Lenz, P.; Urtasun, R. Are we ready for autonomous driving? The KITTI vision benchmark suite. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 3354–3361, 2012.

[4] Qi, C. R.; Litany, O.; He, K. M.; Guibas, L. Deep Hough voting for 3D object detection in point clouds. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, 9276–9285, 2019.

[5] Zhang, Z. W.; Sun, B.; Yang, H. T.; Huang, Q. X. H3DNet: 3D object detection using hybrid geometric primitives. In: *Computer Vision – ECCV 2020. Lecture Notes in Computer Science, Vol. 12357.* Vedaldi, A.; Bischof, H.; Brox, T.; Frahm, J. M. Eds. Springer Cham, 311–329, 2020.

[6] Yan, Y.; Mao, Y. X.; Li, B. SECOND: Sparsely embedded convolutional detection. *Sensors* Vol. 18, No. 10, 3337, 2018.

[7] Yang, H.; Shi, C.; Chen, Y. H.; Wang, L. W. Boosting 3D object detection via object-focused image fusion. *arXiv preprint* arXiv:2207.10589, 2022.

[8] Liu, Z.; Zhang, Z.; Cao, Y.; Hu, H.; Tong, X. Group-free 3D object detection via transformers. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2929–2938, 2021.

[9] Yin, T. W.; Zhou, X. Y.; Krähenbühl, P. Center-based 3D object detection and tracking. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 11779–11788, 2021.

[10] Duan, Y.; Zhu, C. Y.; Lan, Y. Q.; Yi, R. J.; Liu, X. W.; Xu, K. DisARM: Displacement aware relation module for 3D detection. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 16959–16968, 2022.

[11] Cheng, B. W.; Sheng, L.; Shi, S. S.; Yang, M.; Xu, D. Back-tracing representative points for voting-based 3D object detection in point clouds. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 8959–8968, 2021.

[12] Xie, Q.; Lai, Y. K.; Wu, J.; Wang, Z. T.; Zhang, Y. M.; Xu, K.; Wang, J. MLCVNet: Multi-level context VoteNet for 3D object detection. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 10444–10453, 2020.

[13] Qi, C. R.; Chen, X. L.; Litany, O.; Guibas, L. J. ImVoteNet: Boosting 3D object detection in point clouds with image votes. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 4403–4412, 2020.

[14] Yin, J. B.; Zhou, D. F.; Zhang, L. J.; Fang, J.; Xu, C. Z.; Shen, J. B.; Wang, W. G. ProposalContrast: Unsupervised pre-training for LiDAR-based 3D object detection. In: *Computer Vision – ECCV 2022. Lecture Notes in Computer Science, Vol. 13699.* Avidan, S.; Brostow, G.; Cissé, M.; Farinella, G. M.; Hassner, T. Eds. Springer Cham, 17–33, 2022.

[15] Sun, P.; Kretzschmar, H.; Dotiwalla, X.; Chouard, A.; Patnaik, V.; Tsui, P.; Guo, J.; Zhou, Y.; Chai, Y. N.; 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, 2443–2451, 2020.

[16] Zhou, Y.; Tuzel, O. VoxelNet: End-to-end learning for point cloud based 3D object detection. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 4490–4499, 2018.

[17] Yang, B.; Luo, W. J.; Urtasun, R. PIXOR: Real-time 3D object detection from point clouds. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 7652–7660, 2018.

[18] Shi, S. S.; Wang, Z.; Shi, J. P.; Wang, X. G.; Li, H. S. From points to parts: 3D object detection from point cloud with part-aware and part-aggregation network. *IEEE Transactions on Pattern Analysis and Machine Intelligence* Vol. 43, No. 8, 2647–2664, 2021.

[19] Shi, S. S.; Guo, C. X.; Jiang, L.; Wang, Z.; Shi, J. P.; Wang, X. G.; Li, H. S. PV-RCNN: Point-voxel feature set abstraction for 3D object detection. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 10526–10535, 2020.

[20] Chen, X. Z.; Ma, H. M.; Wan, J.; Li, B.; Xia, T. Multi-view 3D object detection network for autonomous driving. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 6526–6534, 2017.

[21] Wang, H. Y.; Shi, S. S.; Yang, Z.; Fang, R. Y.; Qian, Q.; Li, H. S.; Schiele, B.; Wang, L. W. RBGNet: Ray-based grouping for 3D object detection. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 1100–1109, 2022.

[22] Lan, Y. Q.; Duan, Y.; Shi, Y. F.; Huang, H.; Xu, K. 3DRM: Pair-wise relation module for 3D object detection. *Computers & Graphics* Vol. 98, 58–70, 2021.

[23] Lan, Y. Q.; Duan, Y.; Liu, C. Y.; Zhu, C. Y.; Xiong, Y. S.; Huang, H.; Xu, K. ARM3D: Attention-based relation module for indoor 3D object detection. *Computational Visual Media* Vol. 8, No. 3, 395–414, 2022.

[24] Charles, R. Q.; Hao, S.; Mo, K. C.; Guibas, L. J. PointNet: Deep learning on point sets for 3D classification and segmentation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 77–85, 2017.

[25] Qi, C. R.; Yi, L.; Su, H.; Guibas, L. J. PointNet++: Deep hierarchical feature learning on point sets in a metric space. In: Proceedings of the 31st International Conference on Neural Information Processing Systems, 5105–5114, 2017.

[26] Chen, J. T.; Lei, B. W.; Song, Q. Y.; Ying, H. C.; Chen, D. Z.; Wu, J. A hierarchical graph network for 3D object detection on point clouds. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 389–398, 2020.

[27] Xie, Q.; Lai, Y. K.; Wu, J.; Wang, Z. T.; Lu, D. N.; Wei, M. Q.; Wang, J. VENet: Voting enhancement network for 3D object detection. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, 3692–3701, 2021.

[28] Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A. N.; Kaiser, Ł; Polosukhin, I. Attention is all you need. In: Proceedings of the 31st International Conference on Neural Information Processing Systems, 6000–6010, 2017.

[29] Chen, T.; Kornblith, S.; Norouzi, M.; Hinton, G. A simple framework for contrastive learning of visual representations. In: Proceedings of the 37th International Conference on Machine Learning, 1597–1607, 2020.

[30] He, K. M.; Fan, H. Q.; Wu, Y. X.; Xie, S. N.; Girshick, R. Momentum contrast for unsupervised visual representation learning. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 9726–9735, 2020.

[31] Hjelm, R. D.; Fedorov, A.; Lavoie-Marchildon, S.; Grewal, K.; Bachman, P.; Trischler, A.; Bengio, Y. Learning deep representations by mutual information estimation and maximization. *arXiv preprint* arXiv:1808.06670, 2018.

[32] Wang, W. G.; Zhou, T. F.; Yu, F.; Dai, J. F.; Konukoglu, E.; Van Gool, L. Exploring cross-image pixel contrast for semantic segmentation. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, 7283–7293, 2021.

[33] Yin, J. B.; Fang, J.; Zhou, D. F.; Zhang, L. J.; Xu, C. Z.; Shen, J. B.; Wang, W. G. Semi-supervised 3D object detection with proficient teachers. In: *Computer Vision – ECCV 2022. Lecture Notes in Computer Science, Vol. 13698.* Avidan, S.; Brostow, G.; Cissé, M.; Farinella, G. M.; Hassner, T. Eds. Springer Cham, 727–743, 2022.

[34] Purushwalkam, S.; Gupta, A. Demystifying contrastive self-supervised learning: Invariances, augmentations and dataset biases. In: Proceedings of the 34th International Conference on Neural Information Processing Systems, Article No. 287, 3407–3418, 2020.

[35] Hénaff, O. J.; Koppula, S.; Alayrac, J. B.; van den Oord, A.; Vinyals, O.; Carreira, J. Efficient visual pretraining with contrastive detection. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, 10066–10076, 2021.

[36] Yang, C. Y.; Wu, Z. R.; Zhou, B. L.; Lin, S. Instance localization for self-supervised detection pretraining. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 3986–3995, 2021.

[37] Xiao, T. T.; Reed, C. J.; Wang, X. L.; Keutzer, K.; Darrell, T. Region similarity representation learning. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, 10519–10528, 2021.

[38] Wei, F. Y.; Gao, Y.; Wu, Z. R.; Hu, H.; Lin, S. Aligning pretraining for detection via object-level contrastive learning. In: Proceedings of the 35th Conference on Neural Information Processing Systems, 22682–22694, 2021.

[39] Bai, Y. T.; Chen, X. L.; Kirillov, A.; Yuille, A.; Berg, A. C. Point-level region contrast for object detection pre-training. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 16040–16049, 2022.

[40] Van Gansbeke, W.; Vandenhende, S.; Georgoulis, S.; Van Gool, L. Revisiting contrastive methods

for unsupervised learning of visual representations. In: Proceedings of the 35th Conference on Neural Information Processing Systems, 16238–16250, 2021.

[41] Xie, E. Z.; Ding, J.; Wang, W. H.; Zhan, X. H.; Xu, H.; Sun, P. Z.; Li, Z. G.; Luo, P. DetCo: Unsupervised contrastive learning for object detection. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, 8372–8381, 2021.

[42] Sun, B.; Li, B. H.; Cai, S. C.; Yuan, Y.; Zhang, C. FSCE: Few-shot object detection via contrastive proposal encoding. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 7348–7358, 2021.

[43] Zhan, F. N.; Yu, Y. C.; Wu, R. L.; Zhang, J. H.; Lu, S. J.; Zhang, C. G. Marginal contrastive correspondence for guided image generation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 10653–10662, 2022.

[44] Chen, S. X.; Nie, X. H.; Fan, D.; Zhang, D. Q.; Bhat, V.; Hamid, R. Shot contrastive self-supervised learning for scene boundary detection. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 9791–9800, 2021.

[45] Afham, M.; Dissanayake, I.; Dissanayake, D.; Dharmasiri, A.; Thilakarathna, K.; Rodrigo, R. CrossPoint: Self-supervised cross-modal contrastive learning for 3D point cloud understanding. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 9892–9902, 2022.

[46] Xie, S. N.; Gu, J. T.; Guo, D. M.; Qi, C. R.; Guibas, L.; Litany, O. PointContrast: Unsupervised pre-training for 3D point cloud understanding. In: *Computer Vision – ECCV 2020. Lecture Notes in Computer Science, Vol. 12348*. Vedaldi, A.; Bischof, H.; Brox, T.; Frahm, J. M. Eds. Springer Cham, 574–591, 2020.

[47] Hou, J.; Graham, B.; Nießner, M.; Xie, S. N. Exploring data-efficient 3D scene understanding with contrastive scene contexts. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 15582–15592, 2021.

[48] Liu, K. C.; Xiao, A. R.; Zhang, X. Q.; Lu, S. J.; Shao, L. FAC: 3D representation learning via foreground aware feature contrast. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 9476–9485, 2023.

[49] Rao, Y. M.; Liu, B. L.; Wei, Y.; Lu, J. W.; Hsieh, C. J.; Zhou, J. RandomRooms: Unsupervised pre-training from synthetic shapes and randomized layouts for 3D object detection. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, 3263–3272, 2021.

[50] Van den Oord, A.; Li, Y. Z.; Vinyals, O. Representation learning with contrastive predictive coding. *arXiv preprint* arXiv:1807.03748, 2018.

[51] Kingma, D. P.; Ba, J. Adam: A method for stochastic optimization. *arXiv preprint* arXiv:1412.6980, 2014.

[52] MMDetection3D Contributors. OpenMMLab's next-generation platform for general 3D object detection. 2020. Available at https://github.com/open-mmlab/mmdetection3d.

[53] Zhang, Z. W.; 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, 10232–10243, 2021.

Yao Duan is a Ph.D. candidate in the School of Computing, National University of Defense Technology (NUDT), China. Her research interests include 3D vision, scene understanding, etc.

Renjiao Yi is an associate professor in the School of Computing, NUDT. She is interested in 3D vision problems such as inverse rendering and image-based relighting.

Yuanming Gao is an assistant professor at NUDT. Her research area is computer graphics.

Kai Xu is a professor in the School of Computing, NUDT, where he received his Ph.D. degree in 2011. He serves on the editorial boards of *ACM Transactions on Graphics*, *Computer Graphics Forum*, *Computers & Graphics*, etc.

Chenyang Zhu is an associate professor in the School of Computing, NUDT. His current directions of interest include data-driven shape analysis and modeling, 3D vision, robot perception, navigation, etc.