Semantic-Context Graph Network for Point-Based 3D Object Detection

Shuwei Dong, Xiaoyu Kong, Xingjia Pan, Fan Tang, Wei Li, Yi Chang, *Senior Member, IEEE*, and Weiming Dong, *Member, IEEE*

***Abstract*— Point-based indoor 3D object detection has received increasing attention with the large demand for augmented reality, autonomous driving, and robot technology in the industry. However, the detection precision suffers from inputs with semantic ambiguity, i.e., shape symmetries, occlusion, and texture missing, which would lead that different objects appearing similar from different viewpoints and then confusing the detection model. Typical point-based detectors relieve this problem via learning proposal representations with both geometric and semantic information, while the entangled representation may cause a reduction in both semantic and spatial discrimination. In this paper, we focus on alleviating the confusion from entanglement and then enhancing the proposal representation by considering the proposal's semantics and the context in one scene. A semantic-context graph network (SCGNet) is proposed, which mainly includes two modules: a category-aware proposal recoding module (CAPR) and a proposal context aggregation module (PCAg). To produce semantically clear features from entanglement representation, the CAPR module learns a high-level semantic embedding for each category to extract discriminative semantic clues. In view of further enhancing the proposal representation and leveraging the semantic clues, the PCAg module builds a graph to mine the most relevant context in the scene. With few bells and whistles, the SCGNet achieves SOTA performance and obtains consistent gains when applying to different backbones (0.9% ∼ 2.4% on ScanNet V2 and 1.6% ∼ 2.2% on SUN RGB-D for mAP@0.25). Code is available at https://github.com/dsw-jlu-rgzn/SCGNet.**

***Index Terms*— 3D object detection, graph neural networks, information entanglement.**

I. Introduction

3D OBJECT detection has attracted a surge of attention recently and plays an important role in many intelligent industrial applications such as autonomous driving and robotics manipulation. Although providing accurate geometry information, 3D point clouds lack color and texture information and are irregular and sparse, which makes point cloud-based 3D object detection much more challenging compared with natural image-based detection methods [1], [2], [4].

As the first indoor 3D detection work to process point clouds, VoteNet [3] directly leverages the PointNet/PointNet++ [5], [6] to learn the point cloud representation and proposes a voting mechanism to generate box proposals. However, as shown in Fig. 1(a), it tends to get poor performance (red boxes) in the presence of incomplete, distant, or ambiguous input data. Following VoteNet, recent works focus on improving the performance by encoding multi-level contextual information [7], [8], using geometric primitives for proposal refinement [9], or learning semantics via hierarchical point graph modelling [10].

These methods usually entangle geometric and semantic information in the entire process of the proposal representation learning by directly concatenating point coordinates and their features, which we argue that it damages the discrimination of the learned representation. Directly adopting coordinates of points makes the model susceptible to cardinality [11] and the learned inaccurate geometric relationship would, in turn, mislead the semantic aggregation. As illustrated in Fig. 1(b), in the entangled feature space, the distance of proposals in the same category may be large due to the influence of the improperly encoded spatial information.

To alleviate the confusion sourcing from entanglement and make full use of semantic information, we propose a graph-based proposal feature enhancement module for mining and aggregating the object semantics in one scene. The node features of the graph are abstracted only from semantic information, and the relation among nodes is obtained by taking both semantic and geometric information into account. The most related works to this paper are those which apply the attention mechanism or Transformer architectures to 2D/3D object detection tasks [8], [12], [13], [14], [15], [16]. Compared with these methods, we avoid confusion by decoupling semantic learning from geometry.

In this study, we propose a *semantic-context graph network* (SCGNet), which includes a category-aware proposal recoding (CAPR) module and a proposal context aggregation (PCAg) module. Different from RGB/RGB-D based 3D object detection, the appearance properties are hard to observe from point cloud inputs. To this end, we propose the CAPR to extract discriminative semantic clues for each proposal by learning a high-level semantic embedding for each category.

Manuscript received 27 November 2022; revised 28 March 2023; accepted 19 April 2023. Date of publication 28 April 2023; date of current version 30 October 2023. This work was supported in part by the Beijing Natural Science Foundation under Grant L221013; and in part by the National Natural Science Foundation of China under Grant 62102162, Grant 61832016, Grant 62172126, Grant 62106063, Grant 61976102, Grant U20B2070, and Grant U19A2065. This article was recommended by Associate Editor X. Cao. *(Corresponding authors: Fan Tang; Yi Chang.)*

Shuwei Dong and Yi Chang are with the School of Artificial Intelligence, Jilin University, Changchun 130012, China (e-mail: dongsw21@mails.jlu.edu.cn; yichang@lu.edu.cn).

Xiaoyu Kong is with the School of Computer Science and Technology, Harbin Institute of Technology (Shenzhen), Shenzhen, Guangdong 518055, China (e-mail: 21S151102@stu.hit.edu.cn).

Xingjia Pan is with Momenta, Beijing 215100, China (e-mail: xjia.pan@gmail.com).

Fan Tang is with the Institute of Computing Technology, Chinese Academy of Sciences, Beijing 100190, China (e-mail: tangfan@ict.ac.cn).

Wei Li is with Didiglobal, Beijing 100193, China.

Weiming Dong is with the Institute of Automation, Chinese Academy of Sciences, Beijing 100190, China (e-mail: weiming.dong@ia.ac.cn).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TCSVT.2023.3271318.

Digital Object Identifier 10.1109/TCSVT.2023.3271318

1051-8215 © 2023 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission.
See https://www.ieee.org/publications/rights/index.html for more information.
Authorized licensed use limited to: INSTITUTE OF COMPUTING TECHNOLOGY CAS. Downloaded on October 19,2026 at 08:42:45 UTC from IEEE Xplore. Restrictions apply.

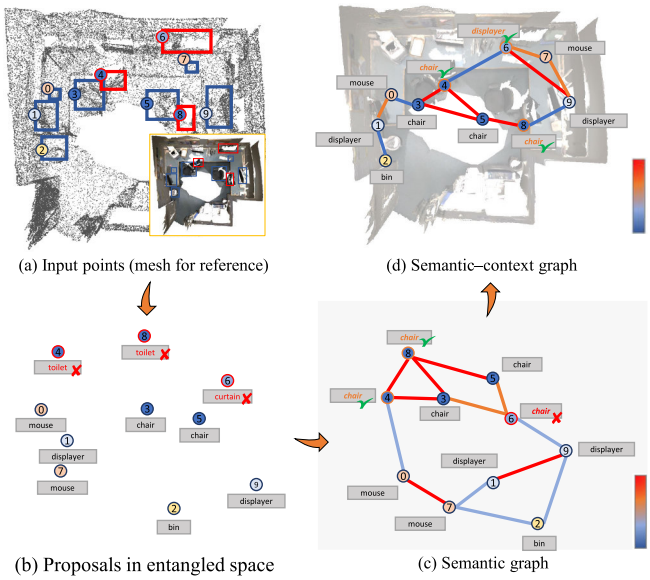


Fig. 1. Illustration of the proposed SCGNet. Original methods directly process point clouds (a) and obtains entangled features (b). Compared with the toy semantic graph where the relation is only based on semantic information (c), the proposed SCGNet (d) proposes the semantic-context graph considering context relationship to enhance the discriminative ability. In (a), rectangles are the ground truth of the object, and the red color means that the object is difficult to detect correctly. Labels in (b), (c), and (d) are the predicted semantic results in different ways. The color bar indicates the relationship strength between proposals.

The decoupled semantic information could be used to correct some misclassified proposals by building the semantic graph (Fig. 1(c)). In contrast with the semantic scene graph which some recent works [17], [18] have discussed, our built semantic graph only contains abstract semantic information and lacks context information which can describe the whole layout in the specific scene. To further enhance the proposal feature, the proposed PCAg adopts a graph-based mechanism to effectively aggregate information with the help of the learned context (Fig. 1(d)). Different from Transformer-based methods, which aggregate all proposals, the PCAg instead selects top-K nodes with original context features to identify relevant object proposals and avoid unnecessary overhead or negative proposals. Detailed comparisons for different SCGNet configurations in terms of node/edge types prove the effectiveness of decoupling geometric and semantic information and the necessity of ensembling multiple kinds of information for 3D object detection on point clouds (Sec. IV-F). Moreover, the proposed SCGNet can be easily plugged into most existing point-based 3D object detection methods to enhance the perception ability. In summary, the main contributions of this study are as follows:

- We consider decoupling semantic features from point cloud inputs and introduce a semantic-context graph network that can work in a plug-and-play way to existing 3D object detection pipelines.
- We propose a category-aware proposal recoding module to learn proposal semantic clues and a proposal context aggregation module to enhance context relations among proposals.
- Experiments show that the proposed SCGNet could achieve the SOTA results and obtain consistent and substantial gains across different backbones on mainstream 3D object detection datasets.

II. Related Work

In this section, we review the related literature in the field of 3D object detection. We first discuss the grid/voxel-based 3D object detection method and then discuss the point-based 3D detection methods, which can process point clouds directly. Recently, graph neural networks (GNNs) [19] have been successfully applied to tackle problems in the 3D domain, and some works have adopted GNNs for 3D object detection. Thus we finally discuss graph methods and their applications to 3D object detection.

A. Grid/Voxel-Based 3D Detection

The point cloud inputs for 3D detection are usually sparse, irregular, and orderless and thus not suitable for directly applying convolutional neural networks (CNNs). Earlier attempts usually project point clouds to 2D grids or 3D voxels. Yang et al. [20] processed point clouds in a fully convolutional manner by using a bird-eye-view projection. Gao et al. [21] employed a multi-scale bird-eye-view projection to generate multiple geometry-aware features. Meng et al. [22] proposed easily-acquired center-click supervision on BEV maps to learn a robust 3D object detector. Chen et al. [23] projected point clouds into frontal views for applying 2D CNNs. As for voxel-based methods, they first converted the irregular point cloud data to regular voxels and then used mature 3D CNNs to process these voxel representations. For instance, Zhou and Tuzel [24] proposed VoxelNet which first converted the point clouds into voxels and extracted voxel features with volumetric CNNs. Shen and Stamos [25] detected objects in 2D and then detected 3D objects within the voxelized 3D frustums, which the 2D detection procedure has defined. These voxel-based methods are limited by their large computational cost in 3D CNNs. GSDN [26] tackled the performance issues by using an encoder-decoder architecture built from sparse 3D convolutional blocks. The major weakness of these grid/voxel-based methods is their accuracy. Because when point clouds are projected to grids or voxels, these methods [23], [24], [27], [28], [29], [30] will inevitably lose much accuracy during the projection procedure. Grid-based and voxel-based methods suffer from the entanglement of geometric and semantic information since they treat the geometry coordinate as the model input, which limits the feature representation. FCAF3D [31] is a state-of-the-art voxel-based indoor 3D object detection method that utilizes sparse 3D convolutions and is anchor-free. FCAF3D performs classification and regression on the 3D feature map position of each existing point on multiple scales. However, if SCGNet is applied to FCAF3D, all the points will be treated as nodes in the semantic-context graph, resulting in a huge sparse graph that slows down the process and hinders contextual information localization. Hence, SCGNet is not suitable for the indoor voxel-based method.

Authorized licensed use limited to: INSTITUTE OF COMPUTING TECHNOLOGY CAS. Downloaded on October 19,2026 at 08:42:45 UTC from IEEE Xplore. Restrictions apply.

B. Point-Based 3D Detection

With the help of PointNet/PointNet++ [5], [6], some point-based methods direct computed 3D proposal features from the raw point cloud input. For instance, Shi et al. [32] directly grouped points in 3D boxes to extract object features. VoteNet [3] introduced a new voting method to aggregate points as a set of object candidates according to their voted centers. Due to objects being irregularly and sparsely distributed in the whole scene, the performance of VoteNet is limited by the quality of proposal embedding from PointNet++. Some follow-up works, such as MLCVNet [7] and HGNet [10], leveraged the semantic information to improve the generation of object proposals. H3DNet [9] introduced a set of overcomplete geometric primitives to provide high-fidelity object proposals. BRNet [33] backtracked the vote centers to refine the object proposals. These methods adopted semantic or geometric information directly to refine the proposals but entangled geometric and semantic information during the proposal representation learning procedure. Moreover, ProposalContrast [34] learned robust 3D representations by contrasting region proposals and eventually benefiting the localization of objects. ProficientTeachers [35] advocated a box-wise contrastive learning strategy to explore representation on expressive 3D boxes. We focus on mining the semantic information to enhance the proposal features. Different from RGB images, it is hard to model a scene for point cloud inputs. In this study, we show how to unify the semantic clues and proposal context relations for better scene understanding to boost the performance of object classification and localization.

C. GNN for 3D Object Detection

Research on graph neural networks [36], [37], [38] seeks to construct and update a graph with nodes and edges to model dependencies among nodes flexibly. Spectral graph convolution [39] is performed in graph convolutional networks to propagate information among nodes iteratively. Graph attention network [40] employed an attention mechanism in specifying relative weights among different neighbor nodes and shows astonishing performance over previous works. Following the graph attention network, Wen et al. [41] performed graph attention for region-level relation enhancement and regional-global relation enhancement in images and the corresponding words in sentences. 3D point cloud can be naturally and efficiently represented by a graph inasmuch as the graph can represent an original surface that is discretized. Hu [42] discussed the applications of graph spectral methods in low-level point cloud processing. Furthermore, Zhang [43] proposed a hypergraph-based new model based on the tensor-based methods which can estimate spectrum components and frequency coefficients of point clouds in the hypergraph. Chen [44] proposed a 3D point cloud resampling method that is rooted in graph signal processing, they use a graph to represent the surface of an original object to capture the point cloud's local dependencies. Deng [45] designed hypergraph spectral filters for point cloud resampling to preserve distinct and sharp point cloud features. In the 3D perceptual domain, some approaches also treat the point cloud as a graph. Landrieu and Simonovsky [46] partitioned a point cloud into simple geometrical shapes and treated them as graph nodes for semantic segmentation. Point-GNN [47] leveraged the graph representation to extract shape features for 3D object detection. These two methods treat points as graph nodes to aggregate point features and cause the problem that they neglect some global relations between different regions. To solve the problem, Yin et al. [48] built spatial relations among different regions by treating the non-empty pillar grids as graph nodes to enhance the representation ability of the pillar features. Object DGCNN [49] learned the point cloud features in BEV and then leveraged DGCNN framework to model object relations. Our method aims to leverage the proposed semantic-context graph at the object level to aggregate the object semantics in one scene.

III. Methods

A. Preliminaries and Overview

Given a set of points, the goal of the point-based 3D detector is to produce the 3D bounding boxes with corresponding semantic labels for all objects in the scene. Fig. 2 (top) shows the typical feature flow for VoteNet [3] detector, which could be roughly divided into three steps: (I) feature extraction (PointNet++ [6]), (II) proposal initialization (Hough Voting [50] and vote aggregating), and (III) proposal prediction (Classification, Regression, and NMS [51]). For *step-I*, PointNet++ [6] is adopted to extract features for input point clouds ($T \times 3$). A subset of the input points with their coordinates and enriched point-wise features ($M \times (3 + F)$) are output for further processing. Note that M is the point number and 3, F are the dimensions of point coordinates and features respectively. For *step-II*, the enriched point features are further aggregated into proposal features by Hough Voting [50] considering local spatial constraints. After *step-II*, N object proposals with their voting coordinates and features will be output. For *step-III*, 3D NMS [51] along with classification and regression heads are utilized to conduct the final predictions. The detector's performance is limited by the quality of the proposal features. Although both semantic and local spatial information is encoded in point/proposal-wise features, the high-level relation of the whole input scene is not involved in the typical three-step pipeline.

We propose a plug-and-play (PnP) proposal feature enhancement module between *step-II* and *step-III* by considering the proposal's semantics and context within the same scene. Without enough color/texture information, it is hard to understand the scene information for point cloud inputs. The proposal features from regional points through PointNet++ [6] contain complicated information, which is hard to model the semantic information for the proposals. To this end, we propose to re-code the proposals for generating category-aware embeddings by learning global category representations (Sec. III-B) and then build a proposal graph using context relation as edges (Sec. III-C). Algorithm 1 describes details of the training procedure of SCGNet. As shown in Fig. 2 (bottom), the proposed SCGNet is easy to combine

Authorized licensed use limited to: INSTITUTE OF COMPUTING TECHNOLOGY CAS. Downloaded on October 19,2026 at 08:42:45 UTC from IEEE Xplore. Restrictions apply.

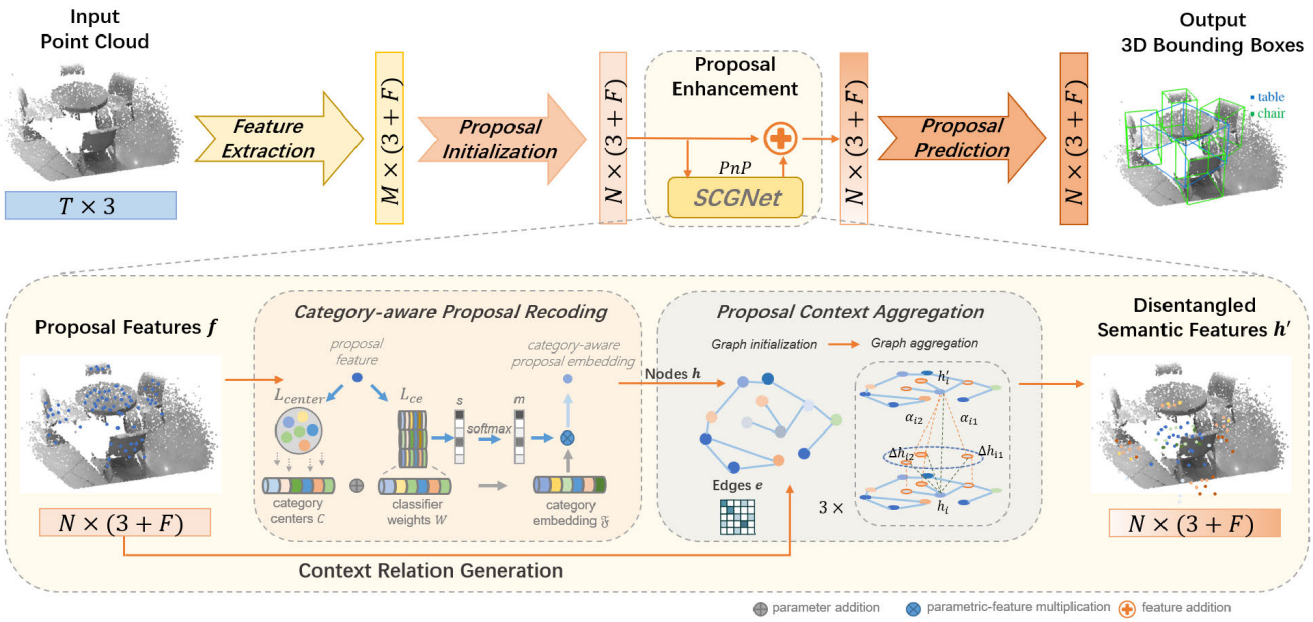


Fig. 2. An overview of our SCGNet. Our model takes N proposals with its features as inputs and outputs enhanced proposal features. The category-aware proposal recoding module first encodes semantic embedding for each proposal. SCGNet then constructs the most relevant relations between proposals as an undirected sparse graph through the context feature similarity screening strategy. In the semantic-context graph module, the semantic embeddings together with the proposal context, are unified to generate a new representation for each proposal. The new representation is then added to the original proposal features to enhance the localization and classification. Note that our model is plug and play (**PnP**) and can boost the performance of most point-based indoor 3D object detection.

Algorithm 1 Training Procedure for SCGNet

Input: proposal features $\boldsymbol{f} \in \mathbb{R}^{N\times F}$, graph layer number L_g and totally T epochs;
Output: enhanced proposal features $\boldsymbol{f} \in \mathbb{R}^{N\times F}$;

Initial $\boldsymbol{W}$, $\boldsymbol{C}$;
for $t = 1 : T$ **do**
  Acquire $\boldsymbol{W}$, $\boldsymbol{C}$;
  $\mathfrak{F} \leftarrow \boldsymbol{W} \oplus \boldsymbol{C}$;
  $\boldsymbol{s} \leftarrow CLF(\boldsymbol{f})$;
  Compute $\boldsymbol{m}$ with $\boldsymbol{s}$ by Eq. 4;
  $\boldsymbol{h} \leftarrow CAPR(\mathfrak{F}, \boldsymbol{m})$ by Eq. 5;
  $\boldsymbol{z} \leftarrow \varphi(\boldsymbol{f})$;
  Generate $\boldsymbol{r}$ with $\boldsymbol{z}$ according to Eq. 7;
  Build the graph $\boldsymbol{G} : \boldsymbol{G} =< \mathcal{N}, \mathcal{E} >$ by $\boldsymbol{h}$, $\boldsymbol{r}$ and let the graph into sparse form according to Eq. 8;
  for $l = 1 : L_g$ **do**
    $\beta \leftarrow a(W_1, \boldsymbol{h})$;
    Calculate α through β according to Eq. 12;
    $\Delta\boldsymbol{h} \leftarrow f_g(\boldsymbol{h})$;
    Calculate $\boldsymbol{h}'$ through $\boldsymbol{h}$, $\Delta\boldsymbol{h}$ and α according to Eq. 14;
    Update $\boldsymbol{h} \leftarrow \boldsymbol{h}'$
  end
  Compute L_{ce} by Eq. 1;
  Compute L_{center} by Eq. 2;
  Update $\boldsymbol{W}$ and $\boldsymbol{C}$ in CAPR;
  Update full model parameters;
  Output $\boldsymbol{f} \leftarrow \boldsymbol{f} + \boldsymbol{h}'$;
end

with most existing point-based methods for the shape of the inputs and outputs is not changed.

B. Category-Aware Proposal Recoding

Previous methods [52] directly aggregated proposal features to establish scene context. However, since the semantic and spatial information is encoded in each proposal, it is hard to model the semantic relation for different proposals in one scene. To mitigate this problem, we propose a category-aware proposal recoding module (CAPR) to learn features with more semantic clues. Due to the point clouds do not contain ample color or texture information to learn semantic embedding directly, we propose to learn category embedding across all proposals and then re-code features for each proposal by the learned category embedding.

With the fact that the weights of the classifier actually contain abundant semantic information since the feature activation trained from all the input point clouds are recorded in the classifier weights. Consequently, we add a new classifier and use the weights from the classification head (CLF) as the semantic embedding for each category. Formally, $\boldsymbol{W} = [w_i]_{i=1}^{L}$, $w_i \in \mathbb{R}^F$ is the weights parameter of the last MLP layer in the classification head, where F, L are the feature dimension and the number of the categories respectively. w_i contains condensed semantic information of class i. For the classification head, given N proposals and L class, we use the standard cross-entropy loss:

$$\mathcal{L}_{\text{ce}} = -\frac{1}{N}\sum_{i=1}^{N}\sum_{j=1}^{L} y_{ij}\log(s_{ij}), \tag{1}$$

Authorized licensed use limited to: INSTITUTE OF COMPUTING TECHNOLOGY CAS. Downloaded on October 19,2026 at 08:42:45 UTC from IEEE Xplore. Restrictions apply.

where y_{ij}, s_{ij} represent the truth label and the prediction probability of the i^{th} proposal for the j^{th} class.

Meanwhile, the feature center of each proposal cluster with the same label is also a representation of the category. Thus we maintain a set of semantic centers for each category with the help of the center loss:

$$\mathcal{L}_{\text{center}} = \frac{1}{2}\sum_{i=1}^{N} \left\| \boldsymbol{f}_i - \boldsymbol{c}_{y_i} \right\|_2^2, \tag{2}$$

where c_{y_i} represents the feature center of the y_i^{th} class, f_i is the feature of the i^{th} proposal. Let $\boldsymbol{C} = [c_i]_{i=1}^{L}$ which denotes L category center features. The final category embedding $\mathfrak{F} \in \mathbb{R}^{L\times F}$ is obtained via

$$\mathfrak{F} = \boldsymbol{W} \oplus \boldsymbol{C} = \left[\mathfrak{f}_1, \ldots, \mathfrak{f}_j, \ldots, \mathfrak{f}_L\right], \tag{3}$$

where $\oplus$ is the add operation, $\mathfrak{f}_j \in \mathbb{R}^F$ is the category embedding $\mathfrak{F}$ of category j. Note that $\boldsymbol{W}$ and $\boldsymbol{C}$ will be continuously updated during training so that category embedding will be more accurate over time.

With the learned category embedding, we re-code the proposal features by converting from category-aware embedding to proposal-specific features. Since the outputs of the classification results in $\boldsymbol{s}$ represent the significance of each category to the object, we can leverage these outputs to describe the relational degree of each category-aware embedding and then re-code the semantic feature globally. Formally, we compute the mapping weight $m_{ij} \in \boldsymbol{m}$:

$$m_{ij} = \frac{\exp\left(s_{ij}\right)}{\sum_j \exp\left(s_{ij}\right)}, \tag{4}$$

where m_{ij} represent the mapping weight from i^{th} proposal to the j^{th} class. Thus we can compute the semantic feature $\boldsymbol{h} = [\boldsymbol{h}_i]_{i=1}^{N}, h_i \in \mathbb{R}^F$ by summing the weighted output of all category embedding:

$$h_i = \textstyle\sum_{j=0}^{L} m_{ij}\mathfrak{f}_j. \tag{5}$$

Here, $i = 1 \sim N$, where N is the number of the proposal in the scene after *step-II* and F is the dimension of h_i.

C. Proposal Context Generation and Aggregation

Category-aware proposal recoding relieves the dilemma of incomplete and ambiguous inputs. However, it processes each proposal separately, lacking context information. Holding on to the philosophy that object recognition ability can be maximally enhanced only when the scene is understood with both clear semantic clues and context information, we build a semantic-context graph $\boldsymbol{G}$: $\boldsymbol{G} =< \mathcal{N}, \mathcal{E} >$ with these two information in the proposal context aggregation (PCAg) module as shown in Fig. 2. To utilize semantic clues, nodes $\mathcal{N}$ of $\boldsymbol{G}$ are initialized with the semantic feature $\boldsymbol{h}$. Then, each edge $e_{ij} \in \mathcal{E}$ encodes the relationship between two proposals. Since the correlation between objects depends on both the semantic information and the location of the object, it is not appropriate to directly use the semantic features' similarity or spatial Euclidean distance to determine the correlation. As demonstrated in Fig. 3 (b), when directly using spatial distance to model the proposal relationship, the correlation is noisy and redundant. If only using semantic features' similarity to obtain the correlation, the relation will ignore the geometric information, resulting in semantic-similar correlation as shown in Fig. 3(c). In Fig. 3(d), when using context feature similarity, the correlation are distributed in various areas of the scene and avoided the local redundancy, which can represent the scene layout better. We adopt the proposal feature $\boldsymbol{f} \in \mathbb{R}^{N\times F}$ before semantic recoding to calculate the correlation, which is based on the context feature similarity.

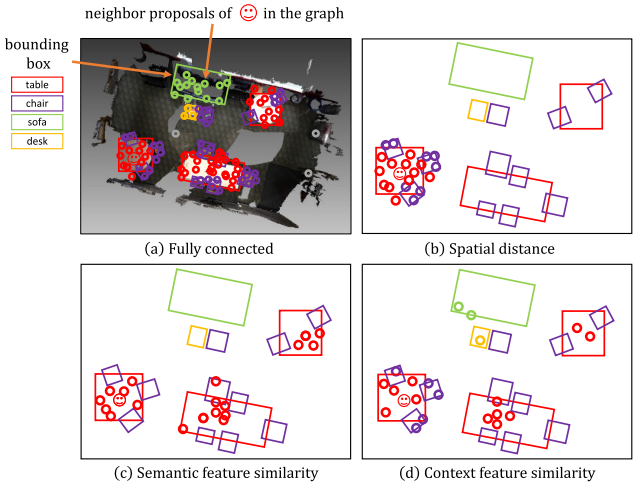


Fig. 3. The figure illustrates the proposal relation generated by different generation strategies. The smile denotes the coordinate position of the target proposal, while the circles represent the coordinate positions of the K most relevant neighbor proposals. The rectangles depict the bounding boxes of the ground truth object. All proposals will undergo prediction for their location regression and category, after which redundant proposals will be eliminated via non-maximum suppression (NMS).

Thus, given i^{th} proposal where h_i, f_i are the i^{th} semantic features $\boldsymbol{h}$ and proposal features $\boldsymbol{f}$ respectively, the proposal features $\boldsymbol{f}$ is considered as the input for learning the context relations. We first transform $\boldsymbol{f} \in \mathbb{R}^{N\times F}$ to a more compact embedding space:

$$z_i = \varphi(f_i), \tag{6}$$

where $z_i \in \mathbb{R}^D$, D is the dimension of the embedding space, and $\varphi(.)$ is a non-linear function. Let $\boldsymbol{z} = \{z_i\}_{i=1}^{N}$ and the relation matrix $\boldsymbol{r} \in \mathbb{R}^{N\times N}$ can be calculated by:

$$r_{ij} = \frac{z_i z_j^T}{\|z_i\| * \|z_j\|}, \tag{7}$$

where $*$ means scalar multiplication. Note that in most point-based 3D object detection frameworks, proposals are generated by randomly selecting vote points. Most generated proposals are invalid or incomplete, as shown in Fig. 3(a). It is ineffective and even introduces much noisy information if considering all the proposal relations in the graph. Therefore, instead of applying a fully connected graph, the key design is building a sparse graph for locating the most representative and informative relations. For each proposal p_i, we screen out the top K relative neighbor nodes to build the graph according to $\boldsymbol{r}$.

Authorized licensed use limited to: INSTITUTE OF COMPUTING TECHNOLOGY CAS. Downloaded on October 19,2026 at 08:42:45 UTC from IEEE Xplore. Restrictions apply.

Formally, this relation search strategy can be written as:

$$Relation(p_i) = Top_K(r_{ij}, K), j = 1, \ldots, N. \quad (8)$$

Then each edge $e_{ij} \in \mathcal{E}$ of our semantic-context graph will be constructed by:

$$\mathcal{N}_i = Relation(p_i), \quad (9)$$

$$e_{ij} = \begin{cases} 1 & \text{if } j \in \mathcal{N}_i \\ 0 & \text{if } j \notin \mathcal{N}_i \end{cases}, \quad (10)$$

where i, j represent node i, j respectively and $\mathcal{N}_i$ represents the neighbor set of node i. This strategy ensures our semantic-context graph can focus on the most relevant relations for object recognition and saves unnecessary consumption.

With these effective context relations as graph edges together with semantic embedding nodes, the PCAg module is employed to learn new proposal representations for further enhancing the classification and localization ability. However, it goes without saying that an object can get various relation information from its surroundings and different relations show the different significance for the object. Therefore, we automatically learn the weight of each neighbor node when aggregating node embedding in the form:

$$\beta_{ij} = a\left(\left[W_1 h_i || W_1 h_j\right]\right), \quad (11)$$

$$\alpha_{ij} = \frac{\exp\left(\beta_{ij}\right)}{\sum_{k \in \mathcal{N}_i} \exp\left(\beta_{ij}\right)}, \quad (12)$$

where β_{ij} is the attention coefficient that indicates the importance of i^{th} node's feature to j^{th} node's. W_1 is a parameterized weight matrix, a is a shared attentional mechanism function that is implemented by a single-layer feed-forward neural network to compute the attention coefficients, α_{ij} is the weight of neighbor node j in $\mathcal{N}_i$, and $[\cdot\|\cdot]$ means concatenation operation. To further capture the relation between node i and node j, we use $f_g(\cdot)$, which is a non-linear function to capture the relative information:

$$\Delta h_{ij} = f_g\left(h_i - h_j\right), \quad (13)$$

where h_i, h_j are semantic features of node i and node j, Δh_{ij} is the relative feature between node i and node j. Thus, with the help of the attention mechanism, we update graph nodes in the form:

$$h_i' = \sigma\left(\sum_{j \in \mathcal{N}_i} \alpha_{ij} W_2([h_j || \Delta h_{ij}])\right), \quad (14)$$

where W_2 is a parameterized weight matrix, σ is an activation function, $j \in \mathcal{N}_i$ represents that node j is in the node i's neighbor set. Lastly, we leverage the multi-head mechanism to boost the representation ability further. In detail, each head executes the transformation of Eq. 14, and their features are concatenated, then the final MLP layer is applied for resizing the feature dimension to be original.

After these steps, the new representation h_i' for each proposal is added to the original proposal feature f_i to improve both classification and localization ability. Fig. 4 shows the architecture details of our graph aggregation procedure in the PCAg module.

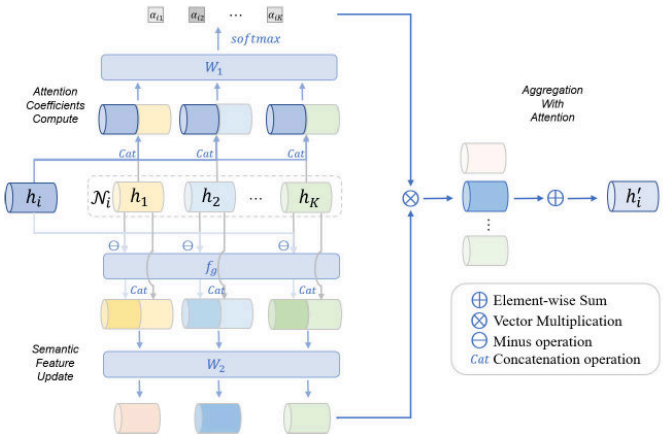


Fig. 4. Architecture details of our graph aggregation procedure in PCAg module. Note that we show the example of the graph aggregation procedure in a one-layer single-head semantic-context graph while our PCAg module contains a three layer multi-head semantic-context graph.

IV. Experiments

A. Datasets and Evaluation Protocol

To demonstrate the effectiveness of our SCGNet, we perform the experiments on two widely used indoor benchmark datasets: SUN RGB-D and ScanNet V2 for the 3D object detection task:

- **ScanNet V2** [53] contains 1,513 reconstructed indoor scenes with per-point instances and 3D bounding boxes of 18 categories. We use 1,201 scenes for training, and 312 scenes are used for testing. A standard evaluation protocol [3] is followed, which uses mean Average Precision (mAP) under different IoU thresholds without considering the orientation of bounding boxes.
- **SUN RGB-D** [54] is a single-veiw RGB-D dataset which is used for 3D scene understanding. There are 10,335 indoor RGB-D images in the dataset, 5,285 images are used for training, and 5,050 images are used for testing. All the images are annotated with the oriented 3D bounding boxes and the corresponding categories. All the point clouds are converted from these RGB-D images based on proposed camera parameters [3]. There are 37 object categories in the SUN RGB-D dataset. We follow the standard evaluation protocol as VoteNet [3] and report the details on the ten most common categories.

Input data preparation. The training data setting mostly follows VoteNet [3] that we sub-sample 40,000 points for ScanNet V2 and 20,000 points from a point cloud on SUN RGB-D. After sampling, a height feature for each point, indicating its distance to the floor, is included in the XYZ coordinates features. For data augmentation, point clouds on the x-axis and y-axis are randomly flipped on the ScanNet V2 dataset. Some other augmentation strategies, such as rotating and scaling, are all applied in the training process. On the SUN RGB-D dataset, we randomly flip the point clouds along the x-axis following VoteNet [3]. Rotating and scaling are also used for the point clouds on the SUN RGB-D dataset.

Authorized licensed use limited to: INSTITUTE OF COMPUTING TECHNOLOGY CAS. Downloaded on October 19,2026 at 08:42:45 UTC from IEEE Xplore. Restrictions apply.

TABLE I
PERFORMANCE COMPARISON WITH STATE-OF-THE-ART 3D OBJECT DETECTION NETWORKS ON SCANNET V2 VALIDATION SET. POINTFORMER+ MEANS THE VOTENET EQUIPPED WITH POINTFORMER AND † DENOTES THAT THE MODEL IS IMPLEMENTED ON MMDETECTION3D. WE TREAT THE VOTENET†, H3DNET† AND GROUPFREE3D† AS BASE MODELS AND PLUG OUR SCGNET IN THESE MODELS TO ENHANCE THE PERFORMANCE

Method	mAP@0.25	mAP@0.50
HGNet [10]	61.3	34.4
GSPN [57]	62.8	34.8
Pointformer+ [58]	64.1	-
3D-MPA [59]	64.2	49.2
MLCVNet [7]	64.7	42.1
VoteNet† [3]	63.8	44.2
VoteNet† + SCGNet	66.2 (2.4 ↑)	46.1 (1.9 ↑)
H3DNet† [9]	66.1	47.7
H3DNet† + SCGNet	67.0 (0.9 ↑)	49.3 (1.6 ↑)
GroupFree3D† (w2x,L12,O512) [8]	68.2	52.6
GroupFree3D† (w2x,L12,O512) + SCGNet	69.1 (0.9 ↑)	53.1 (0.5 ↑)

B. Setups and Implementation Details

To ensure the reproducibility of our method, we implement our SCGNet on the code base of MMdetection3D [55] and all experiments are running on the same environment (one Nvidia Tesla P40 GPU). In general, SCGNet takes the N output proposals as input. In the category-aware proposal recoding module, there is no need to pretrain the parameters for category classification since they are initialized in the same way as the parameters for the 3D detectors. The semantic embedding of each category are obtained from the category classification head and a set of additional semantic centers are maintained as the network parameters. For the semantic-context graph initialization, the non-linear function φ is a two-layer MLP with a Rectified Linear Unit (ReLU) activation function, and the number of the neighbors K is set as 16. To aggregate the semantic-context graph, two graph convolutional layers are used, including a layer with eight attention heads followed by an exponential linear unit (ELU) and a layer with a single attention head. Dropout [56] with $p = 0.4$ is applied to the two layers' inputs and the shared attentional mechanism function a.

We conduct our studies based on four widely-adopted 3D detection methods: VoteNet [3], H3DNet [9], GroupFree3DNet [8], and ImVoteNet [60]. The configurations for different backbones are listed in brief.

- **VoteNet+SCGNet** is end-to-end optimized using AdamW [61] optimizer with a batch size of eight. The initial learning rate is set to 0.008, and the whole network is trained for 220 epochs on both datasets. The learning rate is decreased by 10× after 120, 160, and 180 epochs on both datasets. The total number of the sampled object candidates $N = 256$ and the proposal feature dimension $F = 128$. After voting and sampling procedure, the proposal features are fed into our SCGNet for feature enhancement. All the settings follow the VoteNet.
- **H3DNet+SCGNet** is end-to-end optimized using the AdamW [61] optimizer with the batch size of three. The initial learning rate is 0.008 and is decreased by 10× after 120, 160, and 180 epochs, with a total of 220 epochs on ScanNet V2, following the settings in H3DNet. 256 sampled object candidates are set and their feature dimension is set to 128. Note that H3DNet uses three geometric primitives to provide more accurate signals, and we plug our SCGNet into each primitive after the voting and sampling procedure to boost the performance.
- **GroupFree3D+SCGNet** is trained on the ScanNet V2 dataset with the AdamW [61] optimizer. The initial learning rate is set to 0.006, and the batch size is eight. For the learning rate decay strategy, the learning rate is decreased by 10× after 280 and 340 epochs, with a total of 400 epochs following the settings in GroupFree3D. We set 12 decoder stage as backbone together with 512 sampled object candidates. In view of the added sampled object candidates, the neighbor number K of our SCGNet is set to 32. Different from the settings of VoteNet and H3DNet, the proposal feature dimension $F = 288$ while $F = 128$ in the VoteNet and H3DNet. Thus we simultaneously change the node feature dimension from the SCGNet as 288. We add SCGNet after transformer module in GroupFree3D.
- **ImVoteNet+SCGNet** is trained on SUN RGB-D dataset. ImVoteNet is a two-stage network, and we first trained the first stage network and then stored the parameters as the pre-trained network for the next second stage training. During the second stage, the AdamW [61] optimizer is used and the batch size is set to 16. The initial learning rate is 0.008 and is decreased by 10× after 120 and 160 epochs with a total of 220 epochs. Note that all the basic settings follow the ImVoteNet. The number of the sampled object candidates N is also set to 256 with 128 feature dimension. We add the SCGNet only in the second stage training after the voting and sampling procedure.

C. Quantitative Evaluation

1) Results on ScanNet V2: We evaluate our method on ScanNet V2 dataset by comparing with several SOTA 3D object detection frameworks including HGNet [10], GSPN [57], Pointformer+ [58], 3D-MPA [59], MLCVNet [7], VoteNet [3], H3DNet [9], and GroupFree3d [8].

As shown in Table I, by applying SCGNet to VoteNet†, we achieve a **2.4** improvement on mAP@0.25 and **1.9** on mAP@0.50. When applying SCGNet to H3DNet† and GroupFree3D†, **0.9/0.9** improvements on mAP@0.25 and **1.6/0.5** improvements on mAP@0.50 are obtained. H3DNet leverages a set of geometric primitives to provide more accurate signals for 3D object detection. The enhancement of H3DNet shows that our proposed SCGNet method is complementary to the geometric primitives and can be used together to obtain better performance. GroupFree3D detector is based on the transformer structure to generate object features, which can also be enhanced via SCGNet. Our SCGNet

Authorized licensed use limited to: INSTITUTE OF COMPUTING TECHNOLOGY CAS. Downloaded on October 19,2026 at 08:42:45 UTC from IEEE Xplore. Restrictions apply.

TABLE II

3D OBJECT DETECTION RESULTS ON SUN RGB-D VAL DATASET WITH MAP@0.25.NOTATIONS:† DENOTES THAT THE MODEL IS IMPLEMENTED ON MMDETECTION3D. VOTENET†+OURS, IMVOTENET†+OURS INDICATE APPLYING OUR SCGNET TO THE 3D DETECTOR RESPECTIVELY. THE BEST RESULTS ARE IN **BOLD** WHILE THE SECOND BEST RESULTS ARE MARKED WITH UNDERLINE

Method	bathtub	bed	bookshelf	chair	desk	dresser	nightstand	sofa	table	toilet	mAP
VoteNet [3]	74.4	83.0	28.8	75.3	22.0	29.2	62.2	64.0	47.3	90.1	57.7
BRNet [33]	76.2	86.9	32.5	77.4	29.6	35.9	65.9	66.4	51.8	91.3	61.1
GroupFree3D [8]	80.0	87.8	32.5	79.4	**32.6**	36.0	66.7	70.0	**53.8**	91.1	63.0
imVotenet [60]	75.9	87.6	41.3	76.7	28.7	41.4	**69.9**	70.7	51.5	90.5	63.4
VoteNet† [3]	75.5	85.6	31.9	77.4	24.8	27.9	58.6	67.4	51.1	90.5	59.1
VoteNet† + SCGNet	**81.0**	87.0	34.1	78.8	27.2	30.2	64.7	70.0	50.7	89.1	61.3 (2.2 ↑)
imVoteNet† [60]	77.3	**88.2**	41.4	79.5	30.8	39.9	68.0	72.4	51.5	**91.5**	64.0
imVoteNet† + SCGNet	79.5	87.8	**46.5**	**80.0**	32.0	**43.8**	67.9	**75.2**	52.9	90.7	**65.6** (1.6 ↑)

TABLE III

3D OBJECT DETECTION RESULTS ON SUN RGB-D VAL DATASET WITH MAP@0.50.NOTATIONS:† DENOTES THAT THE MODEL IS IMPLEMENTED ON MMDETECTION3D. VOTENET†+OURS INDICATES APPLYING OUR SCGNET TO THE VOTENET†, IMVOTENET†+OURS INDICATES APPLYING OUR SCGNET TO THE IMVOTENET†. NOTE THAT THE BETTER RESULTS IN THE SAME BASELINE ARE MARKED IN **BOLD**

Method	bathtub	bed	bookshelf	chair	desk	dresser	nightstand	sofa	table	toilet	mAP
VoteNet†[3]	45.4	53.4	6.8	56.5	5.9	12.0	38.6	49.1	**21.3**	68.5	35.8
VoteNet† + SCGNet	**55.5**	**57.9**	**10.0**	**58.4**	**6.6**	**17.4**	**45.5**	**53.2**	19.2	**69.7**	**39.3** (3.5 ↑)
ImVoteNet†[60]	**46.7**	55.7	10.0	58.1	6.5	18.0	39.5	54.3	19.9	**69.3**	37.8
ImVoteNet† + SCGNet	44.5	**56.2**	**11.1**	**59.7**	**7.3**	**26.0**	**46.7**	**58.0**	**20.9**	64.0	**39.5** (1.7 ↑)

can help object proposals to absorb clear and efficient context, eliminating the redundant information brought by the full-connected interaction among all the proposals. All these results of more improved performance on a fairly challenging mAP@0.50 show that SCGNet has the ability to make the detector detect objects more precisely, attributing to its ability to eliminate ambiguity with the generated scene understanding information.

2) Results on SUN RGB-D: We compare our method on the SUN RGB-D dataset with VoteNet [3], BRNet [33], GroupFree3D [8], and ImVoteNet [60]. The detection results are shown in Table II. We achieve **2.2**, **1.6** on mAP@0.25 improvement on VoteNet† and ImVoteNet†. Compared to the ImVoteNet†, SCGNet improves its performance on seven of ten categories in mAP@0.25, the categories such as "bathtub" (**2.2**), "dresser" (**3.9**) and "bookshelf" (**5.1**) get significant improvements. We also show per-category results under mAP@0.50. Equipped with our SCGNet, the VoteNet implemented with MMdetection3D achieves **3.5** improvements in total, and nine of ten categories get different degrees of improvements as shown in Table III. For the ImVoteNet, our SCGNet increases its performance to **39.5** on mAP@0.50. This phenomenon shows that our method indeed helps the model absorb rich scene information for better classification and localization.

D. Qualitative Evaluation

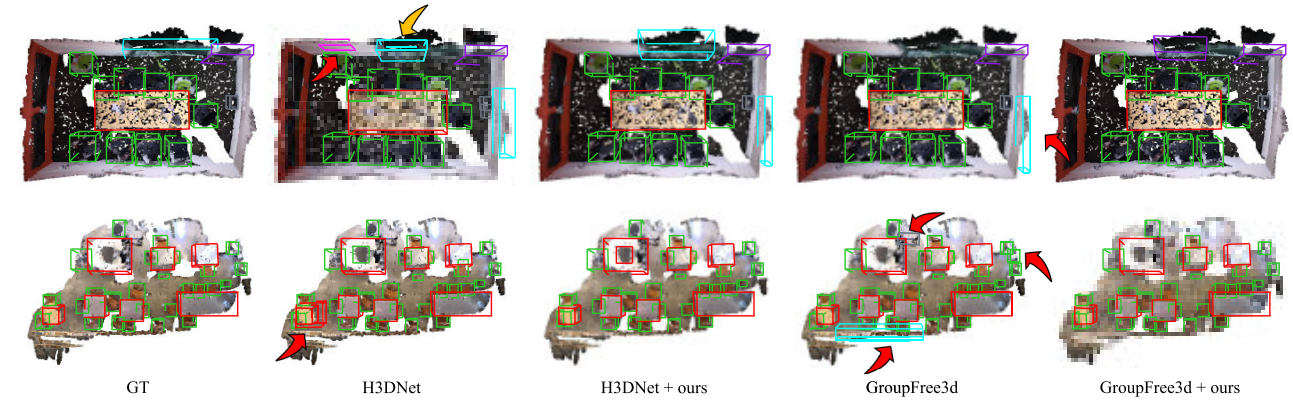
Fig. 5(a) shows a visualization comparison of results in our method and baseline methods (VoteNet [3], H3DNet [9] and GroupFree3D [8]) on the ScanNet V2 validation set. It is observed that due to the large number and variety of objects of ScanNet V2, the existing methods tend to detect redundant bounding boxes (red arrow). SCGNet can effectively correct this error with the help of semantic-context aggregation.

The qualitative comparison results on SUN RGB-D are shown in Fig. 5(b). Although SUN RGB-D has fewer objects to detect, the incomplete point clouds make the prediction worse, including redundant bounding boxes error (red arrow), shape or orientation error (yellow arrow), and category error (green arrow). When integrated with SCGNet, the category-aware proposal recoding and the graph aggregating fix the low-quality feature, relieving the problem of redundant bounding boxes error and category errors. Moreover, the semantic-context graph takes context relation into consideration and hence does a favor for correct shape and orientation prediction. This proves that SCGNet provides an obvious correction effect for location and direction errors.

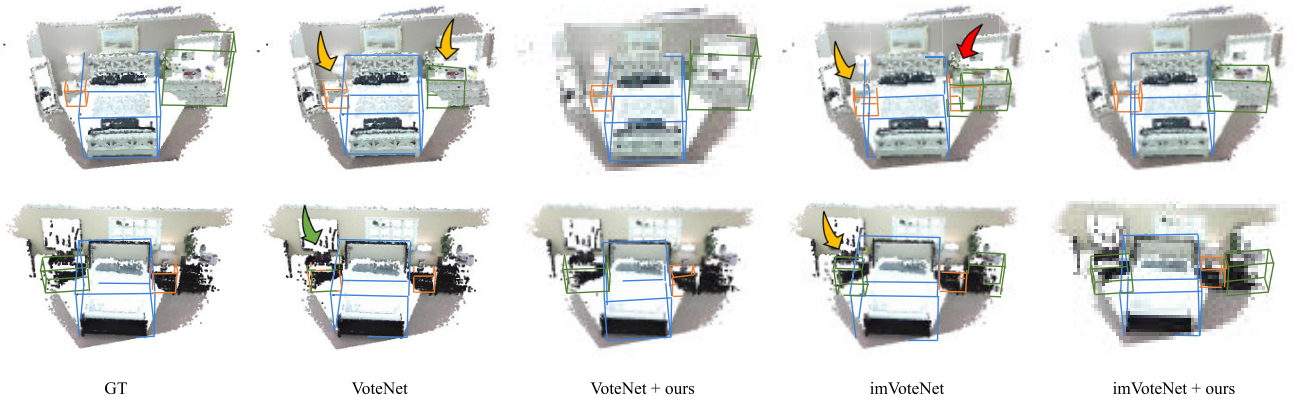
E. Statistic Analysis

In Table II and Table III, we observed that both VoteNet and imVoteNet achieved better mAP results when equipped with our SCGNet. However, not all objects were consistently detected better, as certain objects such as toilets, bathtubs, and tables did not always yield better results. To investigate this phenomenon and further explore the applicability of our model, we conducted a statistical analysis of scene data from the SUN RGB-D dataset. Specifically, we counted the types and numbers of objects that appeared simultaneously around each object in each scene, and then used quartile scaling to scale the count values from 0 to 100 to avoid extreme values. We visualized the statistical data as a heatmap, as shown in Fig. 6. Based on the heatmap, we observed that chairs appeared most frequently around tables and had an overwhelming advantage compared to other objects. In contrast, toilets and

Authorized licensed use limited to: INSTITUTE OF COMPUTING TECHNOLOGY CAS. Downloaded on October 19,2026 at 08:42:45 UTC from IEEE Xplore. Restrictions apply.



(a) ScanNet v2



(b) SUN RGB-D

Fig. 5. Visualization of detection results on ScanNet v2 and SUN RGB-D. Note that H3DNet+ours, GroupFree3D+ours, VoteNet+ours and ImVoteNet+ours mean applying our method to H3DNet, GroupFree3D, VoteNet and ImVoteNet.

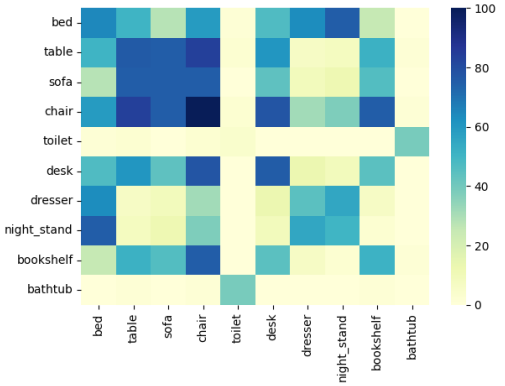


Fig. 6. The heatmap on SUN RGB-D dataset. Note that we counted the types and numbers of objects that appeared simultaneously around each object in each scene and then use quartile scaling to scale the count values from 0 to 100 for avoiding extreme values.

bathtubs appeared together in scenes, and there were hardly any other objects present. Combining these observations with the detection results on SUN RGB-D, we concluded that when the layout in a scene is too monotonous (such as a scene containing only a toilet and a bathtub, or a scene full of chairs), the performance of our SCGNet may be slightly worse.

F. Ablation Study

We conduct ablation studies to investigate the contribution of each part of SCGNet. All the experiments run on the ScanNet V2 dataset and take VoteNet as the backbone method.

TABLE IV

ABLATION STUDIES OF DIFFERENT GRAPH CONSTRUCTION STRATEGIES. THE $\mathcal{N}_{con}$ AND $\mathcal{N}_{sem}$ IN THE NODE COLUMN INDICATE TAKING ORIGIN CONTEXT FEATURES OR SEMANTIC FEATURES AS GRAPH NODES. IN THE EDGE COLUMN, THE EDGES ARE GENERATED WITH THREE DIFFERENT STRATEGIES, GEOMETRIC DISTANCE ($\mathcal{E}_{geo}$), SEMANTIC FEATURE SIMILARITY ($\mathcal{E}_{sem}$), AND CONTEXT FEATURE SIMILARITY ($\mathcal{E}_{con}$)

	Node $\mathcal{N}$	Edge $\mathcal{E}$	mAP@0.25	mAP@0.5
W/o $\boldsymbol{G}$	-	-	63.8	44.2
$\boldsymbol{G}$-I	$\mathcal{N}_{con}$	$\mathcal{E}_{geo}$	64.2	45.2
$\boldsymbol{G}$-II	$\mathcal{N}_{con}$	$\mathcal{E}_{sem}$	65.4	44.9
$\boldsymbol{G}$-III	$\mathcal{N}_{con}$	$\mathcal{E}_{con}$	64.9	45.7
$\boldsymbol{G}$-IV	$\mathcal{N}_{sem}$	$\mathcal{E}_{geo}$	64.5	44.3
$\boldsymbol{G}$-V	$\mathcal{N}_{sem}$	$\mathcal{E}_{sem}$	65.4	44.1
$\boldsymbol{G}$-VI (SCGNet)	$\mathcal{N}_{sem}$	$\mathcal{E}_{con}$	**66.2**	**46.1**

1) Configurations for $< \mathcal{N}, \mathcal{E} >$ of G: We test different combinations of node features and edge generation strategies to verify the effectiveness of our proposed semantic-context graph. As shown in Table IV, the node feature can be the context feature $\mathcal{N}_{con}$ or the semantic feature $\mathcal{N}_{sem}$, and the edge can be generated with three different strategies: geometric distance $\mathcal{E}_{geo}$, semantic feature similarity $\mathcal{E}_{sem}$, and context feature similarity $\mathcal{E}_{con}$. The comparison between experiments

Authorized licensed use limited to: INSTITUTE OF COMPUTING TECHNOLOGY CAS. Downloaded on October 19,2026 at 08:42:45 UTC from IEEE Xplore. Restrictions apply.

TABLE V

THE COMBINATIONS OF GEOMETRIC, SEMANTICS, AND CONTEXTUAL INFORMATION WHEN CONSTRUCTING THE EDGES OF THE UNDIRECTED GRAPH. $\mathcal{E}_{geo+sem+con}$ MEANS COMBINING GEOMETRIC, SEMANTIC, CONTEXT FEATURES TO BUILD THE EDGES; $\mathcal{E}_{sem+con}$ MEANS COMBINING SEMANTIC, CONTEXT FEATURES TO BUILD THE EDGES; $\mathcal{E}_{sem+geo}$ MEANS COMBINING SEMANTIC AND CONTEXT FEATURES TO BUILD THE EDGES;

Edge $\mathcal{E}$	$\mathcal{E}_{geo+sem+con}$	$\mathcal{E}_{sem+con}$	$\mathcal{E}_{sem+geo}$	$\mathcal{E}_{con}$
mAP@0.25	65.9	66.0	63.9	66.2

TABLE VI

ABLATION STUDY BASED ON DIFFERENT COMPONENTS OF OUR MODEL IN SCANNET V2. OUR FINAL MODEL SCGNET IS REPORTED IN THE LAST LINE. WE CONDUCT THIS ABLATION ANALYSIS ON VOTENET. † DENOTES THAT THE MODEL IS IMPLEMENTED ON MMDETECTION3D

	PCAg	CAPR	mAP@0.25
VoteNet†[3]			63.8
	✓		64.6
		✓	64.9
VoteNet†+SCGNet	✓	✓	66.2

TABLE VII

SCANNET V2 TEST MAP@25 AND MAP@50 FOR VARIOUS PROPOSAL NEIGHBOR NUMBERS

K	8	16	32	64	256
mAP@0.25	66.0	66.2	**66.3**	65.5	64.3
mAP@0.50	44.5	**46.1**	45.9	43.7	43.1

TABLE VIII

THE ABLATION ANALYSIS ON L_{center}. † DENOTES THAT THE MODEL IS IMPLEMENTED ON MMDETECTION3D

Method	mAP@0.25	mAP@0.50
VoteNet†	63.8	44.2
VoteNet†+L_{center}	63.2	43.4
VoteNet†+SCGNet	66.2	46.1

G-V and ***G***-VI shows that, when merely using semantic relations and ignoring the context relations in the scene, the detection ability will have a decline (experiment ***G***-V performs weaker than both ***G***-VI and w/o ***G*** on mAP@0.5). As the situation of applying geometric relations in edges, ***G***-I and ***G***-IV suffer performance degradation for the reason that geometric relations can only capture local relations which can not represent the whole scene. When replacing the semantic feature of SCGNet (***G***-VI) with context feature (***G***-III), the low-quality context feature caused by low-quality inputs finally results in a drop on detection accuracy. Moreover, the role of semantic feature and context feature is also important. SCGNet (***G***-VI) uses semantic feature as node feature for a better representation and generates edges with context relations, which can generate the correlation by taking both semantic information and geometric information into consideration. When reversing their role (***G***-II), the low-quality node feature and geometry-ignoring edges lead to a weaker performance.

2) The Verification of CAPR and PCAg: We evaluate the effectiveness of the proposed CAPR module and PCAg module in our SCGNet indivisually, by applying the sub-modules to VoteNet one-by-one on the ScanNet V2 dataset. When the CAPR module was applied to VoteNet, the semantic feature $\boldsymbol{h}$, which is output from the PCAg module, was added to the origin proposal feature $\boldsymbol{f}$ to enhance the feature representation ability. On the other hand, when the PCAg module was applied to VoteNet, it treated the origin features as nodes and built a graph with context feature similarity. As shown in Table VI, our experiments showed that applying the CAPR module leads to an improvement in mAP@0.25 of 1.6, while the PCAg module improved the evaluation score to 64.9. As expected, when equipped with CAPR and PCAg together, the mAP@0.25 was boosted to 66.2. These results demonstrated that the semantic features captured by the designed sub-modules indeed brought satisfactory improvements to the detection performance.

3) Multiple Information Combinations: We further test multiple information combinations when constructing the edges of the undirected graph to verify the impact of multiple information to the edge relations. In detail, features from different sources (semantic features, context features and geometric features) are concatenated together, and an MLP layer is constructed to fuse the features. Following Eqs. 6 – 10, we replace the $\boldsymbol{f} \in \mathbb{R}^{N \times F}$ with the fused features for calculating the graph edge. As shown in Table V, the results conclude that simply using contextual information to generate edges ($\mathcal{E}_{con}$) gets the best performance (**66.2** mAP@0.25), while the combinations without contextual information ($\mathcal{E}_{sem+geo}$) get **63.9** mAP@0.25. When the combinations contain contextual information, the model can maintain high performance. The above results indicate that the combinations information is superfluous when constructing the graph and single context feature can provide enough information to extract context relations.

4) Number of Neighbors K: The number of graph neighbors in each node is an important hyper-parameter that has a strong impact on graph clustering ability. Table VII shows the results of different K. When we set $K = 256$, the sparse unilateral connected digraph becomes the complete digraph, and the detection performance decreases a lot. Although when $K = 64$, the graph has become sparser than $K = 256$, there still exist lots of redundant relations that will do harm to the graph aggregation. If $K = 8$, the performance is decreased compared to $K = 16$ means that the model can not capture enough information from too few relations.

5) The Effectiveness of CAPR: We further verify the effectiveness of the category-aware proposal recoding (CAPR) module. In the CAPR module, the feature center of each cluster with the same label can be viewed as a feature representation of each category. And we obtain these category features through center loss L_{center}. As shown in Table VIII, it is obvious that VoteNet with center loss will slightly damage the detection performance. Our CAPR module can not only erase the performance degradation from center loss but boost the perception ability. Then, a simple classifier is designed to

Authorized licensed use limited to: INSTITUTE OF COMPUTING TECHNOLOGY CAS. Downloaded on October 19,2026 at 08:42:45 UTC from IEEE Xplore. Restrictions apply.

TABLE IX
THE ABLATION STUDIES OF DIFFERENT MLP DEPTHS IN CAPR CLASSIFIER

	MLP Depth	MLP ACC	mAP@0.25
①	3	90.21	66.2
②	6	90.45	66.1
③	9	91.13	66.1

TABLE X
THE COMPARISON BETWEEN SPARSE GRAPH NEURAL NETWORK IN PCA AND TRANSFORMER.† DENOTES THAT THE MODEL IS IMPLEMENTED ON MMDETECTION3D

VoteNet†	+SCGNet	+SCGNet$_{Tr220}$	+SCGNet$_{Tr480}$
63.8	66.2	53.4	63.0

learn semantic clues in each category in the CAPR module. To analyze the influence of the CAPR module, we conducted another ablation study by adding more linear projection layers to the CAPR classifier. The classifier in the CAPR module is implemented with MLP layers. As shown in Table IX, the accuracy of CAPR classifier improves on more layers. When the MLP depth goes from 3 to 9, the accuracy of the classifier improves from **90.21** to **91.93**. However, the final detection performance does not change a lot. Such results show that the mechanism of using high-level semantic representation is more important than adopting a good classifier.

6) The Effectiveness of PCAg: When designing the proposal context aggregation (PCAg) module, the most intuitive idea is leveraging transformer to model the context relations. And in principle, when we change the sparse graph neural network in PCAg module to be fully connected, the graph neural network can be simply replaced with a transformer block. To further verify the effectiveness of PCAg module, we set this ablation study which replaces the graph neural networks in PCAg module with two transformer blocks (SCGNet$_{Tr220}$) following Pointformer [58] and train it with a total of 220 epochs. The result in Table X shows that a significant drop ($-$**12.8** mAP@0.25) is observed by using transformer blocks. When training SCGNet$_{Tr480}$ with 480 epochs, it converges on **63** ($-$**3.2**) in mAP@25. This phenomenon shows that the PCAg module can filter noisy and inefficient relations while transformer blocks capture information from all other proposals.

7) The Effectiveness of the Disentanglement: We investigate the effectiveness of the disentangled semantic feature $\boldsymbol{h'}$. After SCGNet, the disentangled semantic feature $\boldsymbol{h'}$ is added to the origin feature $\boldsymbol{f}$ as the input of the detection head. To validate the effectiveness, we compare the performance of the detection head when fed with different input features. Specifically, in Table XI, we evaluate the 3D detection performance using $\boldsymbol{h'}$, $\boldsymbol{f}$ and $\boldsymbol{h'}$+$\boldsymbol{g}(\boldsymbol{concat})$ as inputs, where $\boldsymbol{h'}$, $\boldsymbol{f}$ and $\boldsymbol{h'}$+$\boldsymbol{g}(\boldsymbol{concat})$ represent feature extracted by different ways. The "+" operator denotes the addition of two features. Our experiments ① and ② demonstrate that the origin feature $\boldsymbol{f}$ contains crucial information which is necessary for subsequent 3D detection, despite being entangled with complex information. We then compare the disentangled semantic feature $\boldsymbol{h'}$ with its entangled counterpart $\boldsymbol{h'}$+$\boldsymbol{g}(\boldsymbol{concat})$ in experiments ③, ④ and observe that $\boldsymbol{h'}$ has better representation ability than $\boldsymbol{h'}$+$\boldsymbol{g}(\boldsymbol{concat})$. Overall, our findings suggest that the addition operation involves a trade-off between the disentangled semantic feature and the origin coupled feature, which contains necessary information for 3D detection. Moreover, the disentangled semantic feature demonstrates a significant advantage over entangled one.

TABLE XI
THE ABLATION STUDY TO EXPLORE WHY $\boldsymbol{h'}$ IS ADDED TO $\boldsymbol{f}$. WE CONDUCT OUR EXPERIMENT OF OUR SCGNET IN VOTENET† ON THE SCANNET V2 DATASET. NOTE THAT $\boldsymbol{f}$, $\boldsymbol{h'}$ AND $\boldsymbol{h'}$+$\boldsymbol{g}(\boldsymbol{concat})$ REPRESENT TREAT $\boldsymbol{f}$, $\boldsymbol{h'}$ AND $\boldsymbol{h'}$+$\boldsymbol{g}(\boldsymbol{concat})$ AS THE INPUT FEATURE OF THE LAST DETECTION HEAD RESPECTIVELY, $\boldsymbol{h'}$ + $\boldsymbol{f}$ REPRESENTS THAT TREAT THE SUMMATION OF $\boldsymbol{f}$ AND $\boldsymbol{h'}$ AS THE INPUT FEATURE OF THE LAST DETECTION HEAD.WE CONCATENATE THE SEMANTIC-CONTEXT FEATURE AND GEOMETRIC COORDINATE AND THEN USE A MLP FOR RESTORING ITS FEATURE DIMENSION TO GENERATE $\boldsymbol{h'}$+$\boldsymbol{g}(\boldsymbol{concat})$. "–" INDICATES THAT THE MODEL CAN NOT BE TRAINED SUCCESSFULLY

	input	mAP@0.25	mAP@0.50
①	$\boldsymbol{h'}$+$\boldsymbol{g}(\boldsymbol{concat})$	–	–
②	$\boldsymbol{h'}$	–	–
③	($\boldsymbol{h'}$+$\boldsymbol{g}(\boldsymbol{concat})$)+$\boldsymbol{f}$	64.9	45.3
④	$\boldsymbol{h'}$+$\boldsymbol{f}$	66.2	46.1
⑤	$\boldsymbol{f}$	63.8	44.2

TABLE XII
THE EFFICIENCY COMPARISON FOR DIFFERENT METHODS. † DENOTES THAT THE MODEL IS IMPLEMENTED ON MMDETECTION3D. GROUPFREE3D REPORTED HERE IS CONFIGURED WITH SIX DECODER STAGES AND 256 SAMPLED OBJECT CANDIDATES

Method	VoteNet †	+ SCGNet	GroupFree3D†	+ SCGNet
Params	0.95M	1.35M	14.47M	16.48M
FLOPS	5.78	5.88	8.40	8.92
Time	0.5090s	0.5737s	0.7936s	0.8526s
mAP@0.25	63.8	66.2	66.3	67.0
mAP@0.50	44.2	46.1	52.6	53.1

8) Model Efficiency: We compare the model size, speed, and computational complexity to evaluate the efficiency of our SCGNet in VoteNet and GroupFree3D. All the experiments are running on the same work environment (one NVIDIA Tesla P40 GPU), and all the models are implemented in the MMdetection3D framework. As shown in Table XII, equipped with our SCGNet, VoteNet, and GroupFree3D increase **0.4**M and **2.01**M parameters, respectively. We gain a **3.76**% AP relative improvement while only a **1.73**% relative increment on FLOPS compared with VoteNet. The results show that our SCGNet is an efficient model and can simultaneously boost the point-based 3D object detection performance.

V. CONCLUSION

In this paper, we propose a simple plug-and-play network, called semantic-context graph network (SCGNet) for better 3D detection performance. It aims to alleviate the confusion sourcing from entanglement and make full use of the information from the point cloud. To this end, we propose the graph-based feature refinement method, where the node features are only from semantic information, and the relation among nodes is obtained by taking both semantic and

Authorized licensed use limited to: INSTITUTE OF COMPUTING TECHNOLOGY CAS. Downloaded on October 19,2026 at 08:42:45 UTC from IEEE Xplore. Restrictions apply.

geometric information into account. In addition, our proposed category-aware proposal recoding module learns the features with more semantic clues to enhance the discriminative ability of proposals. Exhaustive experiments are carried out on the ScanNet v2 and SUN RGB-D datasets illustrating that our SCGNet can bring improvements in most point-based pipelines and achieves state-of-the-art results. Currently, our approach is only suitable for indoor scenes and has limitations on the outdoor scenes for the outdoor scenes have more open and complex environments and we will improve this in the future.

REFERENCES

[1] X. Hu, C.-W. Fu, L. Zhu, T. Wang, and P.-A. Heng, "SAC-Net: Spatial attenuation context for salient object detection," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 31, no. 3, pp. 1079–1090, Mar. 2021.

[2] R. Girshick, "Fast R-CNN," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 1440–1448.

[3] C. R. Qi, O. Litany, K. He, and L. Guibas, "Deep Hough voting for 3D object detection in point clouds," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 9277–9286.

[4] X. Pan et al., "Self-supervised feature augmentation for large image object detection," *IEEE Trans. Image Process.*, vol. 29, pp. 6745–6758, 2020.

[5] R. Q. Charles, H. Su, M. Kaichun, and L. J. Guibas, "PointNet: Deep learning on point sets for 3D classification and segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 652–660.

[6] C. R. Qi, L. Yi, H. Su, and L. J. Guibas, "PointNet++: Deep hierarchical feature learning on point sets in a metric space," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 30, 2017, pp. 1–10.

[7] Q. Xie et al., "MLCVNet: Multi-level context VoteNet for 3D object detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 10447–10456.

[8] Z. Liu, Z. Zhang, Y. Cao, H. Hu, and X. Tong, "Group-free 3D object detection via transformers," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2021, pp. 2949–2958.

[9] Z. Zhang, B. Sun, H. Yang, and Q. Huang, "H3DNet: 3D object detection using hybrid geometric primitives," in *Proc. Eur. Conf. Comput. Vis.* Springer, 2020, pp. 311–329.

[10] J. Chen, B. Lei, Q. Song, H. Ying, D. Z. Chen, and J. Wu, "A hierarchical graph network for 3D object detection on point clouds," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 392–401.

[11] A. Vaswani et al., "Attention is all you need," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 30, 2017, pp. 1–11.

[12] H. Hu, J. Gu, Z. Zhang, J. Dai, and Y. Wei, "Relation networks for object detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 3588–3597.

[13] N. Carion, F. Massa, G. Synnaeve, N. Usunier, A. Kirillov, and S. Zagoruyko, "End-to-end object detection with transformers," in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, 2020, pp. 213–229.

[14] Z. Yuan, X. Song, L. Bai, Z. Wang, and W. Ouyang, "Temporal-channel transformer for 3D LiDAR-based video object detection for autonomous driving," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 32, no. 4, pp. 2068–2078, Apr. 2022.

[15] L. Zhao, J. Guo, D. Xu, and L. Sheng, "Transformer3D-Det: Improving 3D object detection by vote refinement," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 31, no. 12, pp. 4735–4746, Dec. 2021.

[16] Y. Xu et al., "Transformers in computational visual media: A survey," *Comput. Vis. Media*, vol. 8, no. 1, pp. 33–62, Mar. 2022.

[17] C. Zhang, J. Yu, Y. Song, and W. Cai, "Exploiting edge-oriented reasoning for 3D point-based scene graph analysis," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 9705–9715.

[18] J. Wald, H. Dhamo, N. Navab, and F. Tombari, "Learning 3D semantic scene graphs from 3D indoor reconstructions," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 3961–3970.

[19] M. Gori, G. Monfardini, and F. Scarselli, "A new model for learning in graph domains," in *Proc. IEEE Int. Joint Conf. Neural Netw.*, vol. 2, Jul. 2005, pp. 729–734.

[20] B. Yang, W. Luo, and R. Urtasun, "PIXOR: Real-time 3D object detection from point clouds," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 7652–7660.

[21] A. Gao et al., "ESGN: Efficient stereo geometry network for fast 3D object detection," *IEEE Trans. Circuits Syst. Video Technol.*, early access, Aug. 29, 2022, doi: 10.1109/TCSVT.2022.3202810.

[22] Q. Meng, W. Wang, T. Zhou, J. Shen, Y. Jia, and L. Van Gool, "Towards a weakly supervised framework for 3D point cloud object detection and annotation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 44, no. 8, pp. 4454–4468, Mar. 2021.

[23] X. Chen, H. Ma, J. Wan, B. Li, and T. Xia, "Multi-view 3D object detection network for autonomous driving," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 1907–1915.

[24] Y. Zhou and O. Tuzel, "VoxelNet: End-to-end learning for point cloud based 3D object detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 4490–4499.

[25] X. Shen and I. Stamos, "Frustum VoxNet for 3D object detection from RGB-D or depth images," in *Proc. IEEE Winter Conf. Appl. Comput. Vis. (WACV)*, Mar. 2020, pp. 1698–1706.

[26] J. Gwak, C. B. Choy, and S. Savarese, "Generative sparse detection networks for 3D single-shot object detection," in *Proc. Eur. Conf. Comput. Vis.*, 2020, pp. 297–313.

[27] M. Engelcke, D. Rao, D. Z. Wang, C. H. Tong, and I. Posner, "Vote3Deep: Fast object detection in 3D point clouds using efficient convolutional neural networks," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2017, pp. 1355–1361.

[28] J. Ku, M. Mozifian, J. Lee, A. Harakeh, and S. L. Waslander, "Joint 3D proposal generation and object detection from view aggregation," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Oct. 2018, pp. 1–8.

[29] A. H. Lang, S. Vora, H. Caesar, L. Zhou, J. Yang, and O. Beijbom, "PointPillars: Fast encoders for object detection from point clouds," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 12697–12705.

[30] D. Z. Wang and I. Posner, "Voting for voting in online point cloud object detection," in *Proc. Robot., Sci. Syst.*, Rome, Italy, 2015, vol. 1, no. 3, pp. 10–15.

[31] D. Rukhovich, A. Vorontsova, and A. Konushin, "FCAF3D: Fully convolutional anchor-free 3D object detection," in *Proc. 17th Eur. Conf. Comput. Vis. (ECCV)*. Tel Aviv, Israel: Springer Oct. 2022, pp. 477–493.

[32] S. Shi, X. Wang, and H. Li, "PointRCNN: 3D object proposal generation and detection from point cloud," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 770–779.

[33] B. Cheng, L. Sheng, S. Shi, M. Yang, and D. Xu, "Back-tracing representative points for voting-based 3D object detection in point clouds," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 8963–8972.

[34] J. Yin et al., "*ProposalContrast*: Unsupervised pre-training for LiDAR-based 3D object detection," in *Proc. 17th Eur. Conf. Comput. Vis. (ECCV)*. Tel Aviv, Israel: Springer, Oct. 2022, pp. 17–33.

[35] J. Yin et al., "Semi-supervised 3D object detection with proficient teachers," in *Proc. 17th Eur. Conf. Comput. Vis. (ECCV)*. Tel Aviv, Israel: Springer, Oct. 2022, pp. 727–743.

[36] G. Li, M. Müller, A. Thabet, and B. Ghanem, "DeepGCNs: Can GCNs go as deep as CNNs?" in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 9267–9276.

[37] X. Wang et al., "Heterogeneous graph attention network," in *Proc. World Wide Web Conf.*, May 2019, pp. 2022–2032.

[38] L. Gong and Q. Cheng, "Exploiting edge features for graph neural networks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 9211–9219.

[39] T. N. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," 2016, *arXiv:1609.02907*.

[40] P. Veličković, G. Cucurull, A. Casanova, A. Romero, P. Lio, and Y. Bengio, "Graph attention networks," 2017, *arXiv:1710.10903*.

[41] K. Wen, X. Gu, and Q. Cheng, "Learning dual semantic relations with graph attention for image-text matching," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 31, no. 7, pp. 2866–2879, Jul. 2021.

[42] H. Wei, "Graph spectral point cloud processing," in *Graph Spectral Image Processing*. Piscataway, NJ, USA: IEEE, 2021, p. 181.

[43] S. Zhang, S. Cui, and Z. Ding, "Hypergraph spectral analysis and processing in 3D point cloud," *IEEE Trans. Image Process.*, vol. 30, pp. 1193–1206, 2021.

Authorized licensed use limited to: INSTITUTE OF COMPUTING TECHNOLOGY CAS. Downloaded on October 19,2026 at 08:42:45 UTC from IEEE Xplore. Restrictions apply.

[44] S. Chen, D. Tian, C. Feng, A. Vetro, and J. Kovacevic, "Fast resampling of three-dimensional point clouds via graphs," *IEEE Trans. Signal Process.*, vol. 66, no. 3, pp. 666–681, Feb. 2018.

[45] Q. Deng, S. Zhang, and Z. Ding, "An efficient hypergraph approach to robust point cloud resampling," *IEEE Trans. Image Process.*, vol. 31, pp. 1924–1937, 2022.

[46] L. Landrieu and M. Simonovsky, "Large-scale point cloud semantic segmentation with superpoint graphs," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 4558–4567.

[47] W. Shi and R. Rajkumar, "Point-GNN: Graph neural network for 3D object detection in a point cloud," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 1711–1719.

[48] J. Yin, J. Shen, C. Guan, D. Zhou, and R. Yang, "LiDAR-based online 3D video object detection with graph-based message passing and spatiotemporal transformer attention," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 11495–11504.

[49] Y. Wang and J. M. Solomon, "Object DGCNN: 3D object detection using dynamic graphs," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 34, 2021, pp. 20745–20758.

[50] P. V. Hough, "Machine analysis of bubble chamber pictures," in *Proc. Int. Conf. High Energy Accel. Instrum.*, 1959, 1959, pp. 554–556.

[51] A. Neubeck and L. Van Gool, "Efficient non-maximum suppression," in *Proc. 18th Int. Conf. Pattern Recognit. (ICPR)*, vol. 3. 2006, pp. 850–855.

[52] Y. Zhang, D. Huang, and Y. Wang, "Pc-RGNN: Point cloud completion and graph neural network for 3D object detection," in *Proc. AAAI Conf. Artif. Intell.*, vol. 35, no. 4, 2021, pp. 3430–3437.

[53] A. Dai, A. X. Chang, M. Savva, M. Halber, T. Funkhouser, and M. Nießner, "ScanNet: Richly-annotated 3D reconstructions of indoor scenes," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 5828–5839.

[54] S. Song, S. P. Lichtenberg, and J. Xiao, "SUN RGB-D: A RGB-D scene understanding benchmark suite," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 567–576.

[55] M Contributors. (2020). *MMDetection3D: OpenMMLab next-generation platform for general 3D object detection*. [Online]. Available: https://github.com/open-mmlab/mmdetection3d

[56] G. E. Hinton, N. Srivastava, A. Krizhevsky, I. Sutskever, and R. R. Salakhutdinov, "Improving neural networks by preventing co-adaptation of feature detectors," 2012, *arXiv:1207.0580*.

[57] L. Yi, W. Zhao, H. Wang, M. Sung, and L. J. Guibas, "GSPN: Generative shape proposal network for 3D instance segmentation in point cloud," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 3947–3956.

[58] X. Pan, Z. Xia, S. Song, L. E. Li, and G. Huang, "3D object detection with pointformer," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 7463–7472.

[59] F. Engelmann, M. Bokeloh, A. Fathi, B. Leibe, and M. NieBner, "3D-MPA: Multi-proposal aggregation for 3D semantic instance segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 9031–9040.

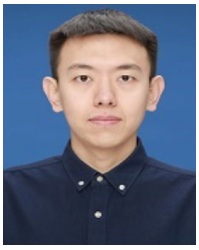
[60] C. R. Qi, X. Chen, O. Litany, and L. J. Guibas, "ImVoteNet: Boosting 3D object detection in point clouds with image votes," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 4404–4413.

[61] I. Loshchilov and F. Hutter, "Decoupled weight decay regularization," 2017, *arXiv:1711.05101*.

Shuwei Dong received the B.Sc. degree in computer science from Jilin University, China, in 2021, where he is currently pursuing the master's degree with the School of Artificial Intelligence. His research interests include 3D computer vision and machine learning.

Xiaoyu Kong received the B.Sc. degree in computer science from Jilin University in 2021. He is currently pursuing the master's degree with the Harbin Institute of Technology. His research interests include computer vision and machine learning.

Xingjia Pan received the B.S. degree in automation and finance from Nankai University, China, in 2015, and the Ph.D. degree in pattern recognition and intelligent system from the National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences, in 2020. He is currently with Momenta, Beijing, China. His research interests include computer vision and machine learning.



Fan Tang received the B.Sc. degree in computer science from North China Electric Power University, China, in 2013, and the Ph.D. degree from the National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences, in 2019. He is currently an Assistant Professor with the Institute of Computing Technology, Chinese Academy of Sciences. His research interests include computer graphics, computer vision, and machine learning.

Wei Li is currently the Leader of the Stereo Vision Team, Map Division, Didiglobal, Beijing, China. As the Leader of Didiglobal, he has worked at Amazon, Microsoft, DiDiLabs, WeRide, Didi Meiyan, and Wenyuan Zhixing, mainly engaged in 3D stereo vision-related works. His research interests include 3D computer vision.

Yi Chang (Senior Member, IEEE) is currently the Dean of the School of Artificial Intelligence, Jilin University (JLU), Changchun, China. He became a Chinese National Distinguished Professor in 2017 and the ACM Distinguished Scientist in 2018. His research interests include information retrieval, data mining, machine learning, natural language processing, and artificial intelligence. He won the Best Paper Award from ACM KDD in 2016 and ACM International WSDM Conference in 2016. He has served as one of the Conference General Chair for the ACM International WSDM Conference in 2018 and the International ACM SIGIR Conference on Research and Development in Information Retrieval in 2020. He is an Associate Editor of IEEE TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING (TKDE).

Weiming Dong (Member, IEEE) received the B.Sc. and M.Sc. degrees in computer science from Tsinghua University, China, in 2001 and 2004, respectively, and the Ph.D. degree in computer science from the University of Lorraine, France, in 2007. He is currently a Professor with the Sino-European Laboratory in Computer Science, Automation and Applied Mathematics (LIAMA) and the National Laboratory of Pattern Recognition (NLPR) at the Institute of Automation, Chinese Academy of Sciences. His research interests include image synthesis and image recognition. He is a member of the ACM.

Authorized licensed use limited to: INSTITUTE OF COMPUTING TECHNOLOGY CAS. Downloaded on October 19,2026 at 08:42:45 UTC from IEEE Xplore. Restrictions apply.