Diff3DETR: Agent-based Diffusion Model for Semi-supervised 3D Object Detection

Jiacheng Deng[1], Jiahao Lu[1], and Tianzhu Zhang[1,2†]

[1] MoE Key Laboratory of Brain-inspired Intelligent Perception and Cognition, University of Science and Technology of China
[2] Deep Space Exploration Lab
{dengjc,lujiahao}@mail.ustc.edu.cn,tzzhang@ustc.edu.cn

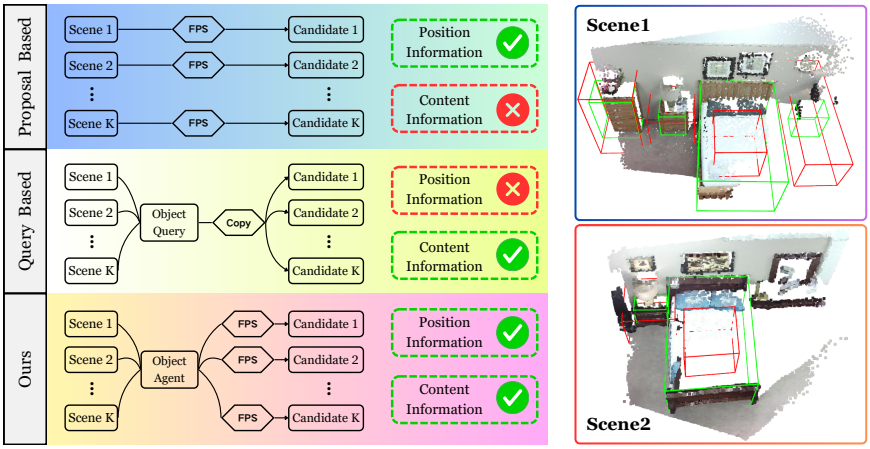
Abstract. 3D object detection is essential for understanding 3D scenes. Contemporary techniques often require extensive annotated training data, yet obtaining point-wise annotations for point clouds is time-consuming and laborious. Recent developments in semi-supervised methods seek to mitigate this problem by employing a teacher-student framework to generate pseudo-labels for unlabeled point clouds. However, these pseudo-labels frequently suffer from insufficient diversity and inferior quality. To overcome these hurdles, we introduce an Agent-based Diffusion Model for Semi-supervised 3D Object Detection (Diff3DETR). Specifically, an agent-based object query generator is designed to produce object queries that effectively adapt to dynamic scenes while striking a balance between sampling locations and content embedding. Additionally, a box-aware denoising module utilizes the DDIM denoising process and the long-range attention in the transformer decoder to refine bounding boxes incrementally. Extensive experiments on ScanNet and SUN RGB-D datasets demonstrate that Diff3DETR outperforms state-of-the-art semi-supervised 3D object detection methods.

Keywords: 3D object detection · Diffusion model · Transformer · Semi-supervised learning

1 Introduction

3D object detection aims to localize and recognize 3D objects in 3D space to facilitate scene understanding, making it crucial for 3D applications such as autonomous driving [1, 9], AR/VR [26], and robotic navigation [41]. The rapid development of deep learning-based methods [12, 18, 19, 23, 36, 42], including PointNet [24,25], Transformer [33], and DETR [3,19], has significantly propelled advancements in 3D object detection. However, most existing approaches heavily rely on labeled point cloud data. Manually annotating vast amounts of 3D point cloud scenes is extremely time-consuming and labor-intensive, which could limit the potential for applying 3D object detection in larger-scale scenarios.

† Corresponding Author



(a) Candidate Generation Mode **(b) Box Difference**

Fig. 1: (a) presents three candidate generation modes: Farthest Point Sampling (FPS), learnable object query, and ours. Our candidate generation mode simultaneously considers the distribution of sampling locations and the learning of content information. (b) displays the geometric differences between initial boxes (in red) and ground truth boxes (in green) in two scenes, highlighting the importance of aggregating features from the correct areas for 3D object detection.

To mitigate the dependence on annotated 3D point cloud data, semi-supervised methods [13, 35, 37, 43] that utilize a small amount of labeled data alongside a large volume of unlabeled data are gaining attention and rapid development. Methods based on semi-supervised learning [40] leverage the untapped information in unlabeled point clouds to compensate for the information loss due to the scarcity of annotated data, thereby effectively enhancing the detection performance. Existing semi-supervised approaches [13, 35, 37, 43] can be broadly categorized into two types: consistency-based methods [43] and pseudo-label-based methods [13, 35, 37]. For consistency-based methods, the core idea is to encourage consistency in the predictions for data augmented in different ways. For instance, SESS [43] enforces consensus on object location, semantic category, and size through three consistency losses between model outputs for differently augmented data. However, due to significant noise in model predictions, consistency constraints might lead to suboptimal results.

On the other hand, pseudo-label-based methods aim to select high-quality pseudo-labels from the model predictions on unlabeled data and then combine these with labeled data for model training, achieving more accurate detection results. 3DIoUMatch [37] and NESIE [35] equip detectors with global and multi-side localization quality estimation modules to assist in pseudo-label filtering and suppression strategies. Diffusion-SS3D [13] leverages a diffusion model [14, 27, 30] to randomly initialize noise sizes and labels to enhance the diversity and quantity

of pseudo-labels and denoise the noisy boxes to improve the quality of pseudo-labels. The diffusion-based method [13] has achieved the best results, marking the diffusion model as a significant trend for the future development of semi-supervised 3D object detection.

Compared to 2D images, the expansive space and sparse distribution of point clouds in 3D scenes result in approximately 90% of spatial areas lacking point cloud coverage. The existing diffusion-based method [13] employs Farthest Point Sampling (FPS) on the point cloud to obtain a fixed number of object candidates, thus avoiding sampling from empty areas which could lower the recall rate. However, compared to object queries in DETR [3], these candidates struggle to learn sufficient content embedding across scenes. Moreover, the existing diffusion-based method [13] aggregates features within the initial boxes. Thus, noisy initial boxes can adversely affect feature aggregation, leading to sub-optimal outcomes.

Based on the discussion above, we identify two critical aspects that need consideration and improvement for building more accurate diffusion-based semi-supervised 3D object detection models: 1) *How to effectively model object candidates?* As shown in Figure 1(a), existing 3D object detection methods [18, 19, 23, 28, 39] primarily obtain candidates through two approaches: Farthest Point Sampling (FPS) and learnable object query [3]. FPS sampling is more likely to distribute candidates across areas where objects are located, reducing the rate of empty sampling. However, it cannot learn content embedding across scenes to aggregate object features effectively. On the other hand, learnable object queries can update and learn across scenes in the dataset but have a higher rate of sampling empty spaces, which lowers the object recall rate. Therefore, balancing the sampling position and the content learning is critical to modeling object candidates effectively. 2) *How to aggregate the correct features to assist in the incremental refinement of noisy initial boxes?* Existing diffusion-based methods [4, 13] generate initial bounding boxes using random Gaussian noise. However, as illustrated in Figure 1(b), these noisy initial boxes significantly differ from the ground truth object boxes. Consequently, the aggregated features often contain substantial errors, making it challenging to iteratively predict accurate object boxes. Therefore, it is crucial to appropriately expand the receptive field around the initial boxes to more accurately locate the correct target area and aggregate features, aiding in the denoising and correction of noisy boxes.

To achieve the above goals, we propose an agent-based diffusion model in a unified DETR architecture for semi-supervised 3D object detection, namely **Diff3DETR**, which consists of an agent-based object query generator and a box-aware denoising module. Overall, our method employs the mean teacher framework, integrating the diffusion model and DETR architecture within a unified model. The randomness of the diffusion process generates a greater quantity of pseudo-labels. Simultaneously, the agent-based object query generator creates object queries that balance sampling locations and content embedding. In the box-aware denoising module, features are aggregated within a long-range perceptive field to iteratively optimize noisy boxes, achieving accurate object predictions. More specifically, the agent-based object query generator initially

establishes learnable object agents to obtain satisfactory content embeddings, where object agents could dynamically adapt to specific scenes through interaction with scene features. The object queries are derived through linear interpolating FPS-sampled locations with object agents. In the box-aware denoising module, the DDIM [30] denoising process and transformer decoder are ingeniously intertwined. Moreover, to ensure each object query focuses on the designated object area during the query process, we bind object queries with the noisy box locations, enhancing the positional dependency of object queries.

To sum up, the contributions of this work can be summarized as follows: (1) We introduce an agent-based diffusion model within a unified DETR framework, which includes an agent-based object query generator and a box-aware denoising module. To the best of our knowledge, this is the first diffusion-based DETR framework in the semi-supervised 3D object detection field. (2) We develop an agent-based object query generator to generate object queries that better adapt to dynamic scenes while balancing sampling locations and content embedding. Additionally, we design a box-aware denoising module that leverages the incremental refinement capabilities of the DDIM denoising process and the long-range attention of the transformer decoder to denoising initial boxes for accurate 3D object detections. (3) Extensive experimental results on the ScanNet and SUN RGB-D datasets demonstrate that Diff3DETR achieves superior performance and outperforms existing state-of-the-art methods.

2 Related Work

In this section, we provide a concise overview of methodologies related to diffusion models for perception tasks and semi-supervised 3D object detection.

Diffusion Models for Perception Tasks. Diffusion models have demonstrated remarkable success in image generation [2, 14, 15, 22, 29]. Consequently, researchers have begun exploring the integration of diffusion models into perception tasks. Pix2Seq-D [5] utilizes the Bit Diffusion model [6] to conduct panoptic segmentation [16] tasks across images and videos. [20] utilizes diffusion models, showcasing their efficacy in extracting discriminative features for classification tasks. DiffusionDet [4] frames object detection as a noise-to-box task, wherein high-quality object bounding boxes are generated by progressively denoising randomly generated proposals. In line with this work, Diffusion-SS3D [13] harnesses the power of diffusion models in semi-supervised 3D object detection settings, aiming to offer a novel approach to generate more dependable pseudo-labels. In this work, we extend the approach of Diffusion-SS3D by proposing the first diffusion-based DETR framework, which includes an agent-based object query generator and a box-aware denoising module.

Semi-supervised 3D Object Detection. While fully supervised methods [11, 18, 19, 23, 34, 39] have demonstrated superior performance, they are often constrained by the labor-intensive process of bounding box annotations, presenting significant practical limitations. Consequently, several semi-supervised 3D object detection techniques [10, 13, 17, 37, 38, 43] have emerged to address

this issue. SESS [43] represents the pioneering effort in semi-supervised 3D object detection. By enforcing consistency between the outputs of the mean teacher [32], SESS effectively learns from unlabeled data. 3DIoUMatch [37] introduces confidence-based filtering and IoU prediction strategies to select high-quality pseudo-labels generated by the teacher model. However, many existing methods heavily rely on the teacher model for pseudo-label generation, limiting their ability to identify bounding boxes beyond the scope of teacher model predictions. To mitigate this challenge, models based on the Diffusion model have been proposed. Diffusion-SS3D [37] leverages the diffusion model for semi-supervised 3D object detection, treating the task as a denoising process to enhance the quality of pseudo-labels. In this work, we adhere to the diffusion-based methodology and employ the agent-based object query generator to produce object queries better suited for complex and dynamic environments. Additionally, we devise the box-aware denoising module to enhance refinement capabilities.

3 Methodology

In this section, we introduce the details of our Diff3DETR. Section 3.1 covers the preliminaries. Section 3.2 describes the Diff3DETR framework, focusing on how the teacher detector generates high-quality pseudo-labels to assist the student detector's training and the computational specifics of both models. Section 3.3 elaborates on the structural details of the Diff3DETR detector.

3.1 Preliminaries

Semi-supervised 3D object detection. Given the point cloud of a scene as input, the goal of 3D object detection is to classify and localize amodal 3D bounding boxes for objects within the point cloud. The point cloud data is $\mathbf{x} \in \mathbb{R}^{n\times 3}$, where n denotes the number of points. Within the semi-supervised framework, we are provided with N training samples. The samples involve a set of N_l labeled scenes $\{\mathbf{x}_i^l, \mathbf{y}_i^l\}_{i=1}^{N_l}$ and a larger set of N_u unlabeled scenes $\{\mathbf{x}_i^u\}_{i=1}^{N_u}$. The ground-truth annotations $\mathbf{y}_i^l$ encompass K objects $\{\mathbf{b}_k^l, l_k\}_{k=1}^{K}$ within $\mathbf{x}_i^l$, with $\mathbf{b}$ and l signifying a collection of bounding box parameters and semantic class labels with a total of N_{cls} classes. Specifically, the bounding box $\mathbf{b}$ is formulated as $\mathbf{b} = \{\mathbf{b}_c, \mathbf{b}_s, \mathbf{b}_o\}$, where $\mathbf{b}_c = \{c_x, c_y, c_z\}$ delineates the centroid coordinates, $\mathbf{b}_s = \{s_l, s_w, s_h\}$ represents the object dimensions, and $\mathbf{b}_o$ is the object orientation along the upright-axis.

Diffusion model. Diffusion models, a class inspired by non-equilibrium thermodynamics [8], present a novel approach in generating data by progressively introducing noise into the data samples. This process is mathematically modeled as a Markov chain [21] consisting of T diffusion steps, with the forward diffusion process described by:

$$q(\mathbf{z}_T|\mathbf{z}_0) = \mathcal{N}(\mathbf{z}_T|\sqrt{\bar{\alpha}_T}\mathbf{z}_0, (1-\bar{\alpha}_T)\mathbf{I}), \tag{1}$$

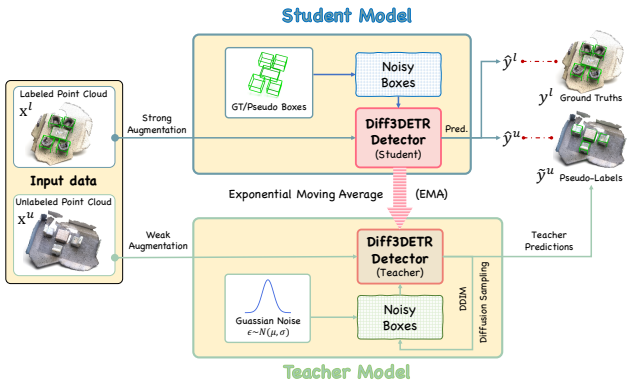


Fig. 2: The framework of Diff3DETR. Diff3DETR adopts the framework of the mean teacher [32], consisting of a student model and a teacher model. The student and teacher models start from GT/pseudo boxes and Gaussian noise, respectively, gradually adding noise to generate noisy boxes and ultimately predicting the accurate object boxes through the Diff3DETR detector. The student model updates its parameters under the supervision of ground truths and pseudo-labels, while the teacher model updates its parameters using an Exponential Moving Average (EMA) strategy.

where $\mathbf{I}$ is identity matrix. In this equation, the original data sample $\mathbf{z}_0$ is transformed into a noisy latent representation $\mathbf{z}_T$ through the application of additive noise. The noise scale $\alpha_s := 1 - \beta_s$ and $\bar{\alpha}_T := \prod_{s=1}^{T} \alpha_s$ are computed as the product of individual noise scales from the first diffusion step to step T, and β_s is a predetermined variance schedule. A neural network $f_\theta(\mathbf{z}_T)$ is then trained to reverse this diffusion process by predicting the noiseless data $\mathbf{z}_0$ from the noisy data $\mathbf{z}_T$, optimizing an L_2 loss objective:

$$\mathcal{L}_{train} = \frac{1}{2} \left\| f_\theta(\mathbf{z}_T) - \mathbf{z}_0 \right\|^2 \tag{2}$$

During the inference phase, the model iteratively reconstructs the original data sample from its noisy counterpart by iteratively applying an update rule. The sequence of transformations $\mathbf{z}_T \rightarrow \mathbf{z}_{T-\Delta} \rightarrow \ldots \rightarrow \mathbf{z}_0$ refines the data sample at each step until the original data is retrieved. A detailed formulation of diffusion models is provided in the supplementary materials.

3.2 The Teacher-Student Framework of Diff3DETR

The proposed Diff3DETR is a novel diffusion-based framework for semi-supervised 3D object detection. As illustrated in Figure 2, Diff3DETR first processes the input point cloud through data augmentation of varying intensities, subsequently feeding them into two branches: the student model and the teacher model. The teacher model introduces Gaussian noise to the 3D bounding boxes and reverses

the diffusion process as in DDIM [30] to generate reliable pseudo bounding boxes. For the student model, the student model takes both labeled and unlabeled point clouds as input and adds Gaussian noise to the ground truth boxes and pseudo boxes to obtain noisy boxes. The student Diff3DETR detector then takes these noisy boxes and the scene point cloud as input, directly predicting the final object boxes without undergoing a multi-round diffusion sampling process. The prediction results $\hat{\mathbf{y}}^l$ and $\hat{\mathbf{y}}^u$ are supervised by ground truths $\mathbf{y}^l$ and pseudo-labels $\tilde{\mathbf{y}}^u$, respectively. Furthermore, the parameters of the teacher Diff3DETR detector are updated using an Exponential Moving Average (EMA) strategy based on the parameters of the student Diff3DETR detector.

To more clearly describe the computational processes of the teacher model and the student model, we detail the computation specifics for both in Algorithm 1 and Algorithm 2, respectively. Compared to the teacher model, the student model mainly differs in the following aspects: 1) the generation of the noisy box's size and label is achieved by incrementally adding Gaussian noise to the GT/pseudo boxes; 2) instead of using a multi-step DDIM diffusion sampling process for denoising the noisy boxes, it directly predicts accurate boxes and computes the loss with ground truths and pseudo-labels. The overall loss function $\mathcal{L}$ for the student model is defined as follows:

$$\mathcal{L} = \mathcal{L}_l(x^l, y^l) + \lambda \mathcal{L}_u(x^u, \tilde{y}^u), \tag{3}$$

where $\mathcal{L}_l$ and $\mathcal{L}_u$ are the detection loss functions for labeled samples and unlabeled samples, respectively. Both loss functions are used in 3DIoUMatch [37] for bounding box regression and classification, combined through a loss weight hyper-parameter λ. Model inference is completed through the student model by adding DDIM diffusion sampling for multi-step iterative denoising, similar to the inference step in Algorithm 1.

3.3 The Detector of Diff3DETR

The overall architecture of the Diff3DETR detector is depicted in Figure 3. Initially, input point clouds are downsampled and feature-extracted using a PointNet++ encoder [24]. These downsampled point clouds are further processed with Farthest Point Sampling (FPS) to identify noisy centers, which serve as preliminary center points for object boxes. The coordinates of these noisy centers are then interpolated in an agent-based object query generator to form object queries for detecting scene targets. Noisy boxes are comprised of noisy centers, along with noisy sizes and noisy labels, which are directly initialized using Gaussian noise. Object queries and noisy boxes undergo prediction of the actual boxes and gradual iterative denoising of noisy boxes within a box-aware denoising module through a decoder layer and the DDIM [30] layer, respectively.

Agent-based object query generator. The agent-based object query generator uniformly samples the normalized scene space at a resolution of $L \times W \times H$ grid points and assigns a learnable variable to each grid point. The object agents are formed by adding each grid point's learnable variable to its grid point position coordinates encoded by a two-layer MLP. The initial object agents are

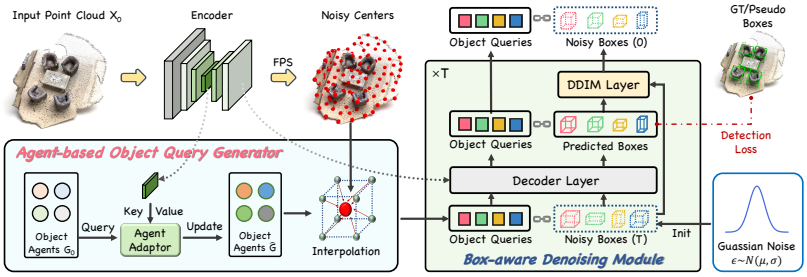


Fig. 3: The overall architecture of the Diff3DETR detector. Input point clouds undergo downsampling and feature extraction, with FPS selecting noisy centers. The agent-based object query generator sets learnable agents interacting with scene features and generates object queries through trilinear interpolation with these centers. Concurrently, noisy boxes initialized with Gaussian noise and object queries are processed in the box-aware denoising module. This module updates queries and predicts boxes, aided by the DDIM layer for iterative denoising.

denoted as $G_0 \in \mathbb{R}^{N_G \times C}$, where $N_G = L \times W \times H$ denotes the number of object agents and C represents the number of feature channels.

To facilitate dynamic adaptability of object agents across various scenes, we design an agent adaptor based on the attention mechanism [33], which can be formulated as follows:

$$Q = \text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V, \tag{4}$$

where Q, K, and V represent the queries, keys, and values, respectively, and d_k denotes the dimension of the keys. In this context, queries represent G_0, and keys and values correspond to the high-level point cloud semantic features from the encoder's intermediate layers. The updated object agents, denoted as $\hat{G}$, are then interpolated through the noisy centers' coordinates $\mathbf{b}_c^{noise}$ to obtain the object queries $\hat{O} \in \mathbb{R}^{N_Q \times C}$, where N_Q represents the number of object queries and C is the number of feature channels.

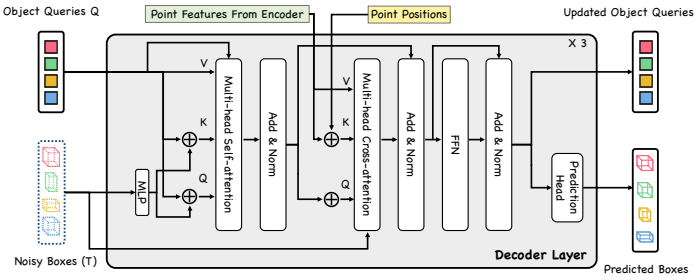


Fig. 4: The architecture of the decoder layer.

Algorithm 1 Teacher Model

```
Input: point cloud, steps, T
Output: pseudo labels
*Extract points and features
pts, feats = Teacher.encoder(pc)
*Update object agents with the input scene
object_agents = update(object_agents, feats)

*Generate object centers
centers = FPS(pts)
*Object query interpolation
queries = interpolate(object_agents, centers)

*Noisy sizes and class labels
sizes_t = normal(mean=0, std=1)
labels_t = normal(mean=0, std=1)

*Uniform sample DDIM times
times = reversed(linespace(-1, T, steps))
time_pairs = list(zip(times[:-1], times[1:]))

for t_cur, t_next in zip(time_pairs):
    *Generate noisy boxes
    boxes_t = random_match((queries, centers),
              (sizes_t, labels_t))
    *Predict pseudo labels
    pls = Teacher.decoder(queries, feats,
              boxes_t, t_cur)
    *Estimate boxes_t at t_next
    boxes_t = ddim(boxes_t, pls, t_cur, t_next)
    *Box renewal
    boxes_t = box_renewal(boxes_t)

*Filtering
pls = filter(pls)
Return pls
```

Algorithm 2 Student Model

```
Input: point cloud, gts, pls
Output: predictions
*Extract points and features
pts, feats = Student.encoder(pc)
*Update object agents with the input scene
object_agents = update(object_agents, feats)
*Pad bounding boxes
sizes = prepare_size(gts, pls)
labels = prepare_label(gts, pls)
*Signal scaling
sizes = (sizes*2 -1)* size_scale
labels = (labels*2 -1)* label_scale
*Generate object centers
centers = FPS(pts)
*Object query interpolation
queries = interpolate(object_agents, centers)
*Corrupt GT bounding boxes
t = randint(0, T)
eps = normal(mean=0, std=1)
sizes_crpt = sqrt(alpha_cumprod(t))*sizes
        +sqrt(1-alpha_cumprod(t))*eps
labels_crpt = sqrt(alpha_cumprod(t))*labels
        +sqrt(1-alpha_cumprod(t))*eps
*Generate noisy boxes
boxes_crpt = match((queries, centers),
             (sizes_crpt, labels_crpt))
*Predict object candidates
preds = Student.decoder(queries, feats,
             boxes_crpt, t)
*Update student by (pseudo) ground truths
loss = detector_loss(preds, gts, pls)
Student = grad_update(Student, loss)
*Update teacher via exponential moving average
Teacher = ema_update(Teacher, Student)
Return preds
```

Box-aware denoising module. Noisy boxes comprise three components: noisy centers, noisy sizes, and noisy semantic labels. As shown in Figure 4, object queries and noisy boxes are fed into the decoder layer, where object queries are updated through self-attention and cross-attention mechanisms to predict accurate boxes. Given object query set $\hat{O} = \{q_i\}$, noisy boxes $\mathbf{b^{noise}} = \{\mathbf{b}_c^{noise}, \mathbf{b}_s^{noise}, \mathbf{b}_o^{noise}\}$ and point cloud features $P = \{p_k\}$, the output feature of the multi-head self-attention of each query element is the aggregation of the values that weighted by the attention weights, formulated as:

$$\text{Self-Att}(q_i, \{q_k\}) = \sum_{h=1}^{H} W_h \left(\sum_{k=1}^{K} A_{h,i,k} \cdot V_h q_k \right), \tag{5}$$

$$A_{h,i,k} = \frac{\exp\left((Q_h(q_i + \mathbf{MLP}(b_i^{noise})))^\top (U_h(q_k + \mathbf{MLP}(b_k^{noise}))) \right)}{\sum_{k=1}^{K} \exp\left((Q_h(q_i + \mathbf{MLP}(b_i^{noise})))^\top (U_h(q_k + \mathbf{MLP}(b_k^{noise}))) \right)}, \tag{6}$$

where h indexes over attention heads, A_h is the attention weight, Q_h, V_h, U_h, W_h indicate the query projection weight, value projection weight, key projection weight, and output projection weight, respectively. The output feature of the multi-head cross-attention of each object query are formulated as:

$$\text{Cross-Att}(q_i, \{p_k\}) = \sum_{h=1}^{H} W_h \left(\sum_{k=1}^{K} \bar{A}_{h,i,k} \cdot V_h p_k \right), \tag{7}$$

$$\bar{A}_{h,i,k} = \frac{\exp\left((Q_h q_i)^\top (U_h p_k) + \mathbf{R}_{i,k}\right)}{\sum_{k=1}^{K} \exp\left((Q_h q_i)^\top (U_h p_k) + \mathbf{R}_{i,k}\right)}, \tag{8}$$

where $\mathbf{R}_{i,k}$ represents the 3D Vertex Relative Position Encoding (3DV-RPE) of the ith object query's corresponding noisy box with respect to the kth point in the point cloud. Inspired by the V-DETR [28] approach, 3DV-RPE encodes a point by its relative position to the target object and is crucial for augmenting the transformers to capture the spatial context of the tokens. For further details, please refer to V-DETR [28]. The denoising process for noisy boxes occurs in the DDIM layer as detailed in Algorithm 1, predicting the noisy boxes for next step t_{next} based on the predicted boxes and noisy boxes at current step t_{cur}.

4 Experiments

4.1 Datasets.

In our study, we conduct evaluations on two primary datasets: ScanNet [7] and SUN RGB-D [31], employing evaluation protocols from existing semi-supervised 3D object detection literature. ScanNet is an established dataset for 3D indoor scene benchmarking and is composed of 1,201 training and 312 validation scenes, reconstructed from 2.5 million high-resolution RGB-D images. Our study places an emphasis on the 18 semantic classes as aligned with prior studies. The SUN RGB-D dataset, another significant 3D benchmark, consists of 5,285 training scenes and 5,050 validation scenes. We assess the models across 10 object classes.

4.2 Evaluation metrics.

For the model evaluation, we split the datasets into partitions with various proportions of labeled and unlabeled data to support semi-supervised learning (SSL). Specifically, we allocate 5%, 10%, 20%, and 100% of labeled data for the ScanNet evaluation, and 1%, 5%, 10%, and 20% for SUN RGB-D. Performance metrics are calculated using the mean Average Precision (mAP). mAP@0.25 at an Intersection over Union (IoU) threshold of 0.25 and mAP@0.5 are reported. These metrics provide insights into precision at a granular level for object detection tasks. The evaluation is conducted across three random data splits to ensure robustness, and report averaged performance and the standard deviation.

4.3 Implementation details

For the detector, we establish a grid of object agents with $(L, W, H) = (16, 16, 4)$ and set the number of noisy centers and object queries to 128. For the diffusion process, we set the maximum timesteps to 1000. Like Diffusion-SS3D [13], we set the mean of the sampling sizes to 0.25. We set the random sampling mean for noisy labels to the inverse of the respective class number. Our teacher model employs a dual-step DDIM sampling technique ($T = 2$) to generate pseudo-labels and produce final evaluation results.

Table 1: Results on ScanNet val dataset under different ratios of labeled data. The best is denoted by **boldface**, while the second best is underlined.

Model	5%		10%		20%		100%	
	mAP@0.25	mAP@0.5	mAP@0.25	mAP@0.5	mAP@0.25	mAP@0.5	mAP@0.25	mAP@0.5
VoteNet [23]	27.9±0.5	10.8±0.6	36.9±1.6	18.2±1.0	46.9±1.9	27.5±1.2	57.8	36.0
SESS [43]	32.0±0.7	14.4±0.7	39.5±1.8	19.8±1.3	49.6±1.1	29.0±1.0	61.3	39.0
3DIoUMatch [37]	40.0±0.9	22.5±0.5	47.2±0.4	28.3±1.5	52.8±1.2	35.2±1.1	62.9	42.1
NESIE [35]	40.5±1.1	23.8±0.8	48.8±0.9	31.1±1.1	54.5±0.8	37.3±0.5	63.8	44.1
Diffusion-SS3D [13]	43.5±0.2	27.9±0.3	50.3±1.4	33.1±1.5	55.6±1.7	36.9±1.4	64.1	43.2
Ours	**45.1±0.5**	**29.2±0.5**	**51.6±1.2**	**34.2±0.9**	**57.0±1.5**	**38.2±0.7**	**65.7**	**44.9**

Table 2: Results on SUN RGB-D val dataset under different ratios of labeled data. The best is denoted by **boldface**, while the second best is underlined.

Model	1%		5%		10%		20%	
	mAP@0.25	mAP@0.5	mAP@0.25	mAP@0.5	mAP@0.25	mAP@0.5	mAP@0.25	mAP@0.5
VoteNet [23]	18.3±1.2	4.4±0.4	29.9±1.5	10.5±0.5	38.9±0.8	17.2±1.3	45.7±0.6	22.5 ±0.8
SESS [43]	20.1±0.2	5.8±0.3	34.2±2.0	13.1±1.0	42.1±1.1	20.9±0.3	47.1±0.7	24.5±1.2
3DIoUMatch [37]	21.9±1.4	8.0±1.5	39.0±1.9	21.1±1.7	45.5±1.5	28.8±0.7	49.7±0.4	30.9±0.2
NESIE [35]	/	/	41.1±1.2	21.8±1.8	47.4±0.8	29.2±1.2	**53.4±0.9**	31.2±1.3
Diffusion-SS3D [13]	30.9±1.0	14.7±1.2	43.9±0.6	24.9±0.3	49.1±0.5	30.4±0.7	51.4±0.8	32.4±0.6
Ours	**32.5±0.8**	**16.3±1.5**	**45.7±1.2**	**26.2±1.1**	**50.2±0.8**	**31.7±0.3**	53.2±1.5	**34.0±1.1**

4.4 Comparison with State-of-the-art Methods

In this section, we compare our Diff3DETR with state-of-the-art approaches on ScanNet dataset and SUN RGB-D dataset.

Results on ScanNet dataset. Table 1 displays a comparison of our approach against the current state-of-the-art methods on the ScanNet validation dataset, achieving leading results across different ratios of labeled data. Through meticulous design in the generation of object queries and more accurate incremental refinement by the box-aware denoising module for noisy boxes,

our method surpasses the best existing method by 1.6% mAP@0.25 and 1.3% mAP@0.5 with only 5% labeled data. Additionally, Figure 5(a) presents qualitative results of our method from the ScanNet dataset.

Results on SUN RGB-D dataset. Table 2 demonstrates the comparative performance of our method against the current state-of-the-art methods on the SUN RGB-D validation dataset, where it achieves leading results across different ratios of labeled data. Specifically, our method outperforms the best existing method by 1.8% mAP@0.25 and 1.3% mAP@0.5 with only 5% labeled data. Additionally, Figure 5(b) presents qualitative results from scenes within the SUN RGB-D dataset, showcasing the capabilities of our method.

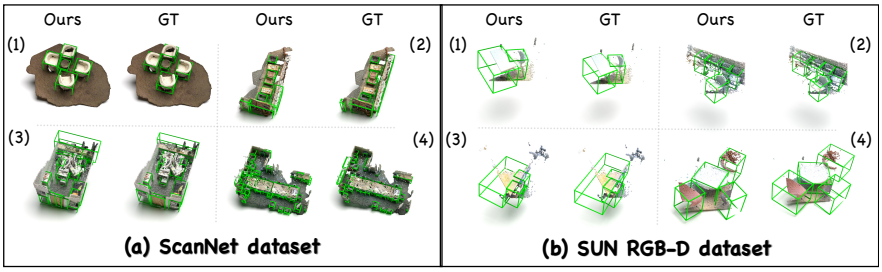


Fig. 5: The qualitative results on ScanNet and SUN RGB-D datasets.

4.5 Ablation Study

Table 3: Evaluation of the model with different designs on ScanNet val dataset. "AA" denotes Agent Adaptor, "AOQG" stands for Agent-based Object Query Generator, "BDM" refers to Box-aware Denoising Module, "3DV-PRE" signifies 3D Vertex Relative Position Encoding, and "DDIM" stands for the denoising process.

	AOQG	AA	BDM	3DV-PRE	DDIM	ScanNet (5%)	
						mAP@0.25	mAP@0.5
[A]	✗	✗	✗	✗	✗	42.2±0.5	27.0±0.3
[B]	✓	✗	✗	✗	✗	43.1±0.7	27.6±0.5
[C]	✓	✓	✗	✗	✗	43.5±0.9	27.8±0.6
[D]	✓	✓	✓	✗	✗	44.3±0.4	28.6±0.3
[E]	✓	✓	✓	✓	✗	44.6±0.2	28.8±0.6
[F]	✓	✓	✓	✓	✓	**45.1±0.5**	**29.2±0.5**

Evaluation of the model with different designs. In Table 3, we present a series of ablation studies to validate the effectiveness of our designs. [A] rep-

resents the baseline IoU-aware VoteNet model [37] without employing DDIM iterative denoising [30]. [B] illustrates that the agent-based object query generator produces improved object queries which aid the model in achieving a 0.9% increase in mAP@0.25 and a 0.6% rise in mAP@0.5. The contrast between [C] and [B] confirms the significance of the agent adaptor for dynamically adapting to the scene. [D] further incorporates the proposed box-aware denoising module, which assists the network in aggregating features from the correct regions and incrementally refining predicted boxes, resulting in an effective increase of 0.8% mAP@0.25 and 0.8% mAP@0.5. [E] validates that 3D vertex relative position encoding is conducive to focusing the object queries on point cloud regions surrounding noisy boxes. [F] represents the complete Diff3DETR model which achieves the best performance among all variants.

Effectiveness of the agent-based object query generator. Object agents are initialized as learnable vectors uniformly distributed in normalized 3D space, and their density across the dimensions of length, width, and height significantly influences the quality of subsequent object query generation. Figure 6 illustrates the visualization of grid point distributions under different $[L, W, H]$ resolution settings for length, width, and height. Results from Table 4 on the ScanNet dataset indicate that the $[16, 16, 4]$ distribution achieves the best performance. This outcome is attributed to the fact that objects within scenes have a notably lower density distribution in vertical height compared to on the horizontal plane, hence allocating more object agents across the horizontal plane aids in modeling a wider variety of spatial semantic information.

Effectiveness of the box-aware denoising module. The box-aware denoising module ingeniously integrates the DETR decoder architecture with the DDIM structure, where the number of blocks in both the decoder layer and DDIM significantly affects the model's ability to decode the scene's object boxes. Table 5 presents ablation studies on the number of blocks in the decoder layer and DDIM, showing that, overall, more decoder blocks and DDIM iterations per cycle yield better detection results. However, increasing the number of blocks in

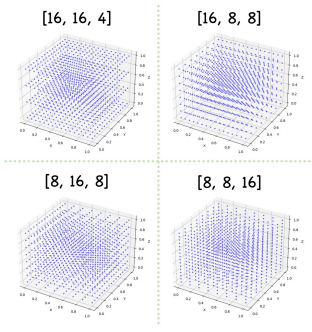


Fig. 6: Distribution visualization of object agents.

Ratio	5%	10%	20%	100%
ScanNet mAP@0.25				
[8, 8, 16]	44.5±0.4	50.9±0.9	56.2±1.3	65.2
[8, 16, 8]	44.9±0.3	51.2±0.6	56.5±0.7	65.4
[16, 8, 8]	44.8±0.3	51.3±1.1	56.3±0.9	65.4
[16, 16, 4]	**45.1±0.5**	**51.6±1.2**	**57.0±1.5**	**65.7**
ScanNet mAP@0.5				
[8, 8, 16]	28.5±0.8	33.6±0.6	37.4±0.2	44.1
[8, 16, 8]	28.8±0.6	33.8±0.9	37.9±0.5	44.5
[16, 8, 8]	28.9±0.3	33.8±0.3	38.0±0.4	44.7
[16, 16, 4]	**29.2±0.5**	**34.2±0.9**	**38.2±0.7**	**44.9**

Table 4: Results under different distributions of object agents.

both the decoder layer and DDIM considerably impacts the model's training and inference speed. Therefore, we seek a trade-off between accuracy and computational cost, selecting #DL=3 and #DDIM=2 as the final model configuration.

Table 5: Ablation studies on the number of blocks in the decoder layer and the DDIM. "#DL" and "#DDIM" respectively denote the number of blocks in the decoder layer and the DDIM.

#DL	#DDIM	ScanNet mAP@0.25				ScanNet mAP@0.5			
		5%	10%	20%	100%	5%	10%	20%	100%
1	1	42.7±0.2	49.3±1.1	54.9±0.8	63.0	26.5±0.4	31.4±0.8	35.1±0.5	42.1
1	2	43.6±0.6	50.2±1.3	55.6±1.2	64.3	27.7±0.8	32.7±0.8	36.9±0.2	43.8
1	4	43.9±0.7	50.5±0.9	55.9±1.1	64.5	27.7±0.2	32.5±0.7	36.3±0.7	43.7
3	1	44.2±0.3	50.8±0.8	56.2±1.1	64.5	27.9±0.7	33.0±0.7	36.5±0.8	43.5
3	2	45.1±0.5	51.6±1.2	57.0±1.5	65.7	29.2±0.5	34.2±0.9	38.2±0.7	44.9
3	4	45.4±0.4	52.0±1.3	57.3±1.7	66.1	28.8±0.5	34.3±0.7	37.9±0.7	45.3
9	1	45.3±0.9	52.0±0.9	57.0±1.4	65.5	29.0±1.1	33.8±0.8	37.1±1.3	44.9
9	2	**46.3±0.7**	52.8±1.0	58.2±1.1	66.3	**30.1±0.6**	35.1±0.6	39.3±0.5	46.2
9	4	46.2±0.5	**53.4±0.9**	**58.7±1.2**	**66.8**	29.7±0.9	**35.5±0.4**	**39.4±1.0**	**46.7**

4.6 Limitations

The diffusion model diffuses object boxes from ground-truth boxes to a random distribution, and the model learns to reverse this noising process. Applying the diffusion model to semi-supervised 3D object detection tasks offers several inherent merits of the diffusion model. First, the diffusion to a random distribution can generate more diverse pseudo-labels. Second, the denoising process coincides with the decoder process of the detection framework, promoting mutual enhancement. However, the slow denoising process of the diffusion model requires more computational resources for training and inference, hindering the algorithm's potential application in real-time devices and large-scale scenes. A more detailed analysis and discussion regarding model overhead are conducted in the supplementary materials.

5 Conclusion

In this paper, we introduce a novel agent-based diffusion model within a unified DETR framework for semi-supervised 3D object detection. Our proposed Diff3DETR comprises an agent-based object query generator and a box-aware denoising module. The agent-based object query generator is designed to produce object queries that effectively adapt to dynamic scenes while striking a balance between sampling locations and content embedding. Meanwhile, the box-aware denoising module utilizes the DDIM denoising process and the long-range attention in the transformer decoder to incrementally refine bounding boxes, thereby achieving better results. Extensive experiments on the ScanNet and SUN RGB-D benchmarks underline the superiority of our Diff3DETR.

Acknowledgements

This work was partially supported by the National Defense Science and Technology Foundation Strengthening Program Funding (Grant 2023-JCJQ-JJ-0219), the National Nature Science Foundation of China (NSFC 62121002), and Youth Innovation Promotion Association CAS.

References

1. Arnold, E., Al-Jarrah, O.Y., Dianati, M., Fallah, S., Oxtoby, D., Mouzakitis, A.: A survey on 3d object detection methods for autonomous driving applications. IEEE Transactions on Intelligent Transportation Systems **20**(10), 3782–3795 (2019)
2. Cao, H., Tan, C., Gao, Z., Xu, Y., Chen, G., Heng, P.A., Li, S.Z.: A survey on generative diffusion models. IEEE Transactions on Knowledge and Data Engineering (2024)
3. Carion, N., Massa, F., Synnaeve, G., Usunier, N., Kirillov, A., Zagoruyko, S.: End-to-end object detection with transformers. In: European conference on computer vision. pp. 213–229. Springer (2020)
4. Chen, S., Sun, P., Song, Y., Luo, P.: Diffusiondet: Diffusion model for object detection. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 19830–19843 (2023)
5. Chen, T., Li, L., Saxena, S., Hinton, G., Fleet, D.J.: A generalist framework for panoptic segmentation of images and videos. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 909–919 (2023)
6. Chen, T., Zhang, R., Hinton, G.: Analog bits: Generating discrete data using diffusion models with self-conditioning. arXiv preprint arXiv:2208.04202 (2022)
7. 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. pp. 5828–5839 (2017)
8. De Groot, S.R., Mazur, P.: Non-equilibrium thermodynamics. Courier Corporation (2013)
9. GEIGER, A., LENZP, U.R.: Arewereadyfor autonomousdriving (2012)
10. Griffiths, D., Boehm, J., Ritschel, T.: Finding your (3d) center: 3d object detection using a learned loss. In: Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XVIII 16. pp. 70–85. Springer (2020)
11. Gwak, J., Choy, C., Savarese, S.: Generative sparse detection networks for 3d single-shot object detection. In: Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part IV 16. pp. 297–313. Springer (2020)
12. He, J., Deng, J., Zhang, T., Zhang, Z., Zhang, Y.: Hierarchical shape-consistent transformer for unsupervised point cloud shape correspondence. IEEE Transactions on Image Processing (2023)
13. Ho, C.J., Tai, C.H., Lin, Y.Y., Yang, M.H., Tsai, Y.H.: Diffusion-ss3d: Diffusion model for semi-supervised 3d object detection. Advances in Neural Information Processing Systems **36** (2024)
14. Ho, J., Jain, A., Abbeel, P.: Denoising diffusion probabilistic models. Advances in neural information processing systems **33**, 6840–6851 (2020)

15. Kingma, D., Salimans, T., Poole, B., Ho, J.: Variational diffusion models. Advances in neural information processing systems **34**, 21696–21707 (2021)
16. Kirillov, A., He, K., Girshick, R., Rother, C., Dollár, P.: Panoptic segmentation. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 9404–9413 (2019)
17. Liu, C., Gao, C., Liu, F., Li, P., Meng, D., Gao, X.: Hierarchical supervision and shuffle data augmentation for 3d semi-supervised object detection. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 23819–23828 (2023)
18. 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. pp. 2949–2958 (2021)
19. Misra, I., Girdhar, R., Joulin, A.: An end-to-end transformer model for 3d object detection. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 2906–2917 (2021)
20. Mukhopadhyay, S., Gwilliam, M., Agarwal, V., Padmanabhan, N., Swaminathan, A., Hegde, S., Zhou, T., Shrivastava, A.: Diffusion models beat gans on image classification. arXiv preprint arXiv:2307.08702 (2023)
21. Norris, J.R.: Markov chains. No. 2, Cambridge university press (1998)
22. Peebles, W., Xie, S.: Scalable diffusion models with transformers. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 4195–4205 (2023)
23. Qi, C.R., Litany, O., He, K., Guibas, L.J.: Deep hough voting for 3d object detection in point clouds. In: proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 9277–9286 (2019)
24. Qi, C.R., Su, H., Mo, K., 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. pp. 652–660 (2017)
25. Qi, C.R., Yi, L., Su, H., Guibas, L.J.: Pointnet++: Deep hierarchical feature learning on point sets in a metric space. Advances in neural information processing systems **30** (2017)
26. Rokhsaritalemi, S., Sadeghi-Niaraki, A., Choi, S.M.: A review on mixed reality: Current trends, challenges and prospects. Applied Sciences **10**(2), 636 (2020)
27. Rombach, R., Blattmann, A., Lorenz, D., Esser, P., Ommer, B.: High-resolution image synthesis with latent diffusion models. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 10684–10695 (2022)
28. Shen, Y., Geng, Z., Yuan, Y., Lin, Y., Liu, Z., Wang, C., Hu, H., Zheng, N., Guo, B.: V-detr: Detr with vertex relative position encoding for 3d object detection. arXiv preprint arXiv:2308.04409 (2023)
29. Sohl-Dickstein, J., Weiss, E., Maheswaranathan, N., Ganguli, S.: Deep unsupervised learning using nonequilibrium thermodynamics. In: International conference on machine learning. pp. 2256–2265. PMLR (2015)
30. Song, J., Meng, C., Ermon, S.: Denoising diffusion implicit models. arXiv preprint arXiv:2010.02502 (2020)
31. Song, S., Lichtenberg, S.P., Xiao, J.: Sun rgb-d: A rgb-d scene understanding benchmark suite. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 567–576 (2015)
32. Tarvainen, A., Valpola, H.: Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results. Advances in neural information processing systems **30** (2017)

33. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł., Polosukhin, I.: Attention is all you need. In: Advances in neural information processing systems. pp. 5998–6008 (2017)
34. Wang, C., Deng, J., He, J., Zhang, T., Zhang, Z., Zhang, Y.: Long-short range adaptive transformer with dynamic sampling for 3d object detection. IEEE Transactions on Circuits and Systems for Video Technology (2023)
35. Wang, C., Yang, W., Zhang, T.: Not every side is equal: Localization uncertainty estimation for semi-supervised 3d object detection. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 3814–3824 (2023)
36. Wang, H., Dong, S., Shi, S., Li, A., Li, J., Li, Z., Wang, L., et al.: Cagroup3d: Class-aware grouping for 3d object detection on point clouds. Advances in Neural Information Processing Systems **35**, 29975–29988 (2022)
37. Wang, H., Cong, Y., Litany, O., Gao, Y., Guibas, L.J.: 3dioumatch: Leveraging iou prediction for semi-supervised 3d object detection. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 14615–14624 (2021)
38. Wu, X., Peng, L., Xie, L., Hou, Y., Lin, B., Huang, X., Liu, H., Cai, D., Ouyang, W.: Semi-supervised 3d object detection with patchteacher and pillarmix. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 38, pp. 6153–6161 (2024)
39. Xie, Q., Lai, Y.K., Wu, J., Wang, Z., Zhang, Y., 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. pp. 10447–10456 (2020)
40. Yang, X., Song, Z., King, I., Xu, Z.: A survey on deep semi-supervised learning. IEEE Transactions on Knowledge and Data Engineering (2022)
41. Ye, C., Qian, X.: 3-d object recognition of a robotic navigation aid for the visually impaired. IEEE Transactions on Neural Systems and Rehabilitation Engineering **26**(2), 441–450 (2017)
42. Zhang, Z., Sun, B., Yang, H., Huang, Q.: H3dnet: 3d object detection using hybrid geometric primitives. In: Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XII 16. pp. 311–329. Springer (2020)
43. Zhao, N., Chua, T.S., Lee, G.H.: Sess: Self-ensembling semi-supervised 3d object detection. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 11079–11087 (2020)