Uncertainty Meets Diversity: A Comprehensive Active Learning Framework for Indoor 3D Object Detection – Supplementary Materials

Jiangyi Wang, Na Zhao*
Singapore University of Technology and Design (SUTD)
wangjiangyi0519@gmail.com, na_zhao@sutd.edu.sg

In the supplementary materials, we first provide the theoretical derivation of Eq. 9 in the main paper in Sec. 1, to support the new formulation in Eq. 10. We also provide more details of our method, along with ablation studies to validate the effectiveness, see Sec. 2. Finally, additional quantitative and qualitative results are provided in Sec. 3.

1. Theoretical Derivation

In the main paper, we directly apply the theoretical derivation of Eq. 9. In this section, we prove that given S disjoint partitions $\{\mathcal{D}_{r,s}\}_{s=1}^{S}$ of the r-th selected dataset $\mathcal{D}_r$, *i.e.*, $\cup_{s=1}^{S}\mathcal{D}_{r,s} = \mathcal{D}_r$, the following equation holds, up to a constant c:

$$\begin{aligned} &E(\mathbf{h}(\mathcal{D}_r)) + \beta||\mathbf{h}(\mathcal{D}_r)||_1 \\ &\overset{c}{=} \sum_{s=1}^{S}\{\frac{N_s}{N_{\text{total}}}E(\mathbf{h}(\mathcal{D}_{r,s})) + \beta||\mathbf{h}(\mathcal{D}_{r,s})||_1\} + \mathcal{O}(\frac{N_{\text{inter}}}{N_{\text{total}}}), \end{aligned} \tag{1}$$

where $\mathbf{h}(\mathcal{D}_{r,s})$ represents the histograms of all objects in s-th partition $\mathcal{D}_{r,s}$, N_s and N_{total} are the number of objects in the s-th partition $\mathcal{D}_{r,s}$ and selected labeled samples $\mathcal{D}_r$, N_{inter} is the number of objects that share the same prototypes but belong to different partitions, and $\mathcal{O}(\cdot)$ is the Big O notation from mathematics.

[*Proof*] We first represent the prototype histograms $\mathbf{h}(\mathcal{D}_r)$ in terms of its entry $h_m = \frac{x_m}{N_{total}}$, *i.e.*,

$$\mathbf{h}(\mathcal{D}_r) = [h_1, h_2, \cdots, h_m, \cdots, h_M], \tag{2}$$

where h_m and x_m indicates the frequency and count of m-th prototype in $\mathcal{D}_r$, respectively, and $M = \sum_c M_c$ represents the total number of prototypes. Then, to link with S partitions of $\mathcal{D}_r$, we represent h_m (x_m) as the summation of $h_{m,s}$ ($x_{m,s}$), which is the frequency (count) of m-th prototype in $\mathcal{D}_{r,s}$, over all partitions $s = 1, 2, ..., S$, and identify the dominant set of prototypes for each partition $\mathcal{D}_{r,s}$:

$$I_s = \{m \in \{1, 2, ..., M\} | s = \arg\max_{s'} h_{m,s'}\}. \tag{3}$$

Here, we term these prototypes within partition $\mathcal{D}_{r,s}$ as *dominant* because each of their frequency $h_{m,s}$ reaches the maximum across all S partitions. To simplify the proof, for each prototype index $m \in I_s$, we define $h_{m,-s}$ as the summation of frequencies across all $(S-1)$ other partitions, *i.e.*, $h_{m,-s} = \sum_{s' \neq s} h_{m,s'}$. Notably, for $m \in I_s$, $h_{m,-s}$ is negligible compared to the dominant term $h_{m,s}$.

Lemma 1. $\sum_{s=1}^{S}\sum_{m\notin I_s}\mathcal{O}(h_{m,s}) = \mathcal{O}(\frac{N_{\text{inter}}}{N_{\text{total}}})$.

[*Proof of Lemma 1*] Since $h_{m,s} = \frac{x_{m,s}}{N_{\text{total}}}$, it is equivalent to show that $\sum_{s=1}^{S}\sum_{m\notin I_s}\mathcal{O}(x_{m,s}) = \mathcal{O}(N_{\text{inter}})$. From the definition of N_{inter} that it represents the number of all intersecting prototypes across different partitions, we have:

$$N_{\text{inter}} = \sum_{m=1}^{M}\max_{s:m\notin I_s}(x_{m,s}). \tag{4}$$

From the pigeonhole principle, we ensure that there must exist one pair of (m_0, s_0) such that, $x_{m_0,s_0} \geq 1/M \cdot N_{\text{inter}}$, and $m_0 \notin I_{s_0}$, implying that:

$$\sum_{s=1}^{S}\sum_{m\notin I_s} x_{m,s} \geq \frac{1}{M}\cdot N_{\text{inter}}. \tag{5}$$

Furthermore, we have the inequality from the other side:

$$\begin{aligned} \sum_{s=1}^{S}\sum_{m\notin I_s} x_{m,s} &= \sum_{m=1}^{M}\sum_{s:m\notin I_s} x_{m,s} \\ &\leq \sum_{m=1}^{M}|\{s \in \{1, ..., S\} | m \notin I_s\}| \max_{s:m\notin I_s}(x_{m,s}) \\ &\leq S \cdot N_{\text{inter}}. \end{aligned} \tag{6}$$

We have shown $\sum_{s=1}^{S}\sum_{m\notin I_s} x_{m,s}$ is bounded by some constant multiples of N_{inter}, as established by Eq. 5 (lower bound) and Eq. 6 (upper bound), proving that $\sum_{s=1}^{S}\sum_{m\notin I_s}\mathcal{O}(x_{m,s}) = \mathcal{O}(N_{\text{inter}})$. □

*Corresponding Author: na_zhao@sutd.edu.sg

Lemma 2. $\sum_{s=1}^{S}\sum_{m\in I_s}\mathcal{O}(h_{m,-s}) = \mathcal{O}(\frac{N_{\text{inter}}}{N_{\text{total}}})$.

[*Proof of Lemma 2*] We prove **Lemma 2** by applying **Lemma 1** intermediately, which is shown as:

$$\begin{aligned}\sum_{s=1}^{S}\sum_{m\in I_s}\mathcal{O}(h_{m,-s}) &= \sum_{m=1}^{M}\sum_{s:m\in I_s}\sum_{s'\neq s}\mathcal{O}(h_{m,s'}) \\ &= \sum_{m=1}^{M}\sum_{s:m\notin I_s}\mathcal{O}(h_{m,s}) \\ &= \sum_{s=1}^{S}\sum_{m\notin I_s}\mathcal{O}(h_{m,s}).\end{aligned} \tag{7}$$

In **Lemma 1**, it has $\sum_{s=1}^{S}\sum_{m\notin I_s}\mathcal{O}(h_{m,s}) = \mathcal{O}(\frac{N_{\text{inter}}}{N_{\text{total}}})$. By applying this result to Eq. 7, **Lemma 2** directly holds. □

Proposition 1. $E(\mathbf{h}(\mathcal{D}_r)) = \sum_{s=1}^{S}\frac{N_s}{N_{\text{total}}}E(\mathbf{h}(\mathcal{D}_{r,s})) + \sum_{s=1}^{S}\frac{N_s}{N_{\text{total}}}\log(\frac{N_s}{N_{\text{total}}}) + \mathcal{O}(\frac{N_{\text{inter}}}{N_{\text{total}}})$.

[*Proof of Proposition 1*] Firstly, we use the definition of entropy on $\mathbf{h}(\mathcal{D}_r) = [h_1, h_2, \cdots, h_M]$. Since $\{I_s\}_{s=1}^{S}$ forms a partition of $\{1, 2, .., M\}$, the summation over the M prototypes can be equivalently expressed as the summation over all elements in I_s across different partitions:

$$\begin{aligned}E(\mathbf{h}(\mathcal{D}_r)) &= \sum_{m=1}^{M} h_m \log(h_m) \\ &= \sum_{s=1}^{S}\sum_{m\in I_s} h_m \log(h_m).\end{aligned} \tag{8}$$

Then, using the definition of $h_{m,-s}$, we can express h_m as $h_m = h_{m,s} + h_{m,-s}$, and apply Taylor expansion [2] to Eq. 8, expanding it around $h_{m,s}$:

$$\begin{aligned}E(\mathbf{h}(\mathcal{D}_r)) &= \sum_{s=1}^{S}\sum_{m\in I_s}(h_{m,s}+h_{m,-s})\log(h_{m,s}+h_{m,-s}) \\ &= \sum_{s=1}^{S}\sum_{m\in I_s} h_{m,s}\log(h_{m,s}) + \sum_{s=1}^{S}\sum_{m\in I_s}\mathcal{O}(h_{m,-s}) \\ &= \sum_{s=1}^{S}\sum_{m\in I_s} h_{m,s}\log(h_{m,s}) + \mathcal{O}(\frac{N_{\text{inter}}}{N_{\text{total}}}).\end{aligned} \tag{9}$$

Here, the last equation follows directly from **Lemma 2**. Subsequently, applying the add-and-subtract technique to the summation over all $m \in I_s$, we first introduce the summation over $m \notin I_s$ and then subtract it, resulting in:

$$\begin{aligned}&E(\mathbf{h}(\mathcal{D}_r)) \\ &= \sum_{s=1}^{S}\{\sum_{m\in I_s} + \sum_{m\notin I_s} - \sum_{m\notin I_s}\}h_{m,s}\log(h_{m,s}) + \mathcal{O}(\frac{N_{\text{inter}}}{N_{\text{total}}}) \\ &= \sum_{s=1}^{S}\sum_{m=1}^{M} h_{m,s}\log(h_{m,s}) - \sum_{s=1}^{S}\sum_{m\notin I_s}\mathcal{O}(h_{m,s}) + \mathcal{O}(\frac{N_{\text{inter}}}{N_{\text{total}}}) \\ &= \sum_{s=1}^{S}\sum_{m=1}^{M} h_{m,s}\log(h_{m,s}) + \mathcal{O}(\frac{N_{\text{inter}}}{N_{\text{total}}}).\end{aligned} \tag{10}$$

The last equation is derived from **Lemma 1**. Finally, we rewrite $h_{m,s}$ as $\frac{N_s}{N_{\text{total}}}\frac{x_{m,s}}{N_s}$ and simplify Eq. 10 directly:

$$\begin{aligned}&E(\mathbf{h}(\mathcal{D}_r)) \\ &= \sum_{s=1}^{S}\sum_{m=1}^{M}\frac{N_s}{N_{\text{total}}}\frac{x_{m,s}}{N_s}\log(\frac{N_s}{N_{\text{total}}}\frac{x_{m,s}}{N_s}) + \mathcal{O}(\frac{N_{\text{inter}}}{N_{\text{total}}}) \\ &= \sum_{s=1}^{S}\frac{N_s}{N_{\text{total}}}\sum_{m=1}^{M}\frac{x_{m,s}}{N_s}\log(\frac{x_{m,s}}{N_s}) \\ &\quad + \sum_{s=1}^{S}\frac{N_s}{N_{\text{total}}}\log(\frac{N_s}{N_{\text{total}}})\sum_{m=1}^{M}\frac{x_{m,s}}{N_s} + \mathcal{O}(\frac{N_{\text{inter}}}{N_{\text{total}}}) \\ &= \sum_{s=1}^{S}\frac{N_s}{N_{\text{total}}}E(\mathbf{h}(\mathcal{D}_{r,s})) + \sum_{s=1}^{S}\frac{N_s}{N_{\text{total}}}\log(\frac{N_s}{N_{\text{total}}}) + \mathcal{O}(\frac{N_{\text{inter}}}{N_{\text{total}}}).\end{aligned} \tag{11}$$

□

Proposition 2. $||\mathbf{h}(\mathcal{D}_r)||_1 = \sum_{s=1}^{S}||\mathbf{h}(\mathcal{D}_{r,s})||_1 + \mathcal{O}(\frac{N_{\text{inter}}}{N_{\text{total}}})$.

[*Proof of Proposition 2*] Similar with the proof of **Proposition 1**, we can simplify the l_1-norm as follows:

$$\begin{aligned}||\mathbf{h}(\mathcal{D}_r)||_1 &= \sum_{m=1}^{M}(h_{m,s}+h_{m,-s}) \\ &= \sum_{s=1}^{S}\sum_{m\in I_s} h_{m,s} + \sum_{s=1}^{S}\sum_{m\in I_s}\mathcal{O}(h_{m,-s}) \\ &= \sum_{s=1}^{S}\sum_{m=1}^{M} h_{m,s} + \mathcal{O}(\frac{N_{\text{inter}}}{N_{\text{total}}}) + \sum_{s=1}^{S}\sum_{m\notin I_s}\mathcal{O}(h_{m,s}) \\ &= \sum_{s=1}^{S}||\mathbf{h}(\mathcal{D}_{r,s})||_1 + \mathcal{O}(\frac{N_{\text{inter}}}{N_{\text{total}}}).\end{aligned} \tag{12}$$

□

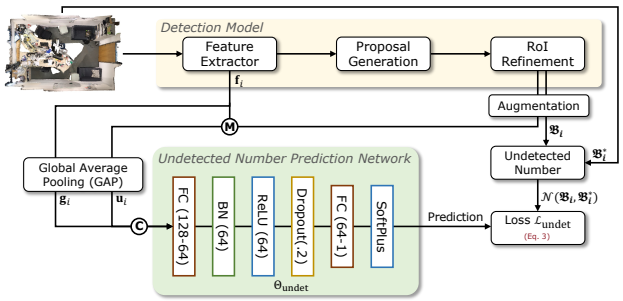

Figure 1. Architecture of undetected object number prediction network Θ_{undet}. The network consists of three-layer MLPs (128-64-1), with 'SoftPlus' activation function. Furthermore, we present a novel data augmentation strategy to handle the data scarcity.

Combining both **Proposition 1** and **Proposition 2**, we conclude the proof that the following equation holds, up to a constant c:

$$E(\mathbf{h}(\mathcal{D}_r)) + \beta||\mathbf{h}(\mathcal{D}_r)||_1 \overset{c}{=} \sum_{s=1}^{S}\{\frac{N_s}{N_{\text{total}}}E(\mathbf{h}(\mathcal{D}_{r,s})) + \beta||\mathbf{h}(\mathcal{D}_{r,s})||_1\} + \mathcal{O}(\frac{N_{\text{inter}}}{N_{\text{total}}}). \tag{13}$$

Here, the constant c equals to $\sum_{s=1}^{S}\frac{N_s}{N_{\text{total}}}\log(\frac{N_s}{N_{\text{total}}})$. □

Remark. 3D indoor scenes naturally form distinct clusters based on the histograms of object categories, each of which is composed of multiple prototypes. To ensure the resulting clusters are approximately disjoint $w.r.t.$ prototype assignment, we apply k-means++ to the histograms of prototypes. As a result, we have $\frac{N_{\text{inter}}}{N_{\text{total}}} \to 0$ as $N_{\text{total}} \to \infty$, which guarantees $\mathcal{O}(\frac{N_{\text{inter}}}{N_{\text{total}}}) \to 0$. Empirically, for a fixed budget expansion ratio $\delta = 6$, we observe that $N_{\text{total}} >$ 1e4 and $\frac{N_{\text{inter}}}{N_{\text{total}}} <$ 2e-2, which is negligible. Thus, we can decompose the original optimization problem, as defined in Eq. 8 in the main paper, into S sub-problems in Eq. 10 within the disjoint clusters.

2. Details of Methodology

We provide the details of the methodology due to the space limit of the main paper, which includes the architecture of the undetected object number prediction network, and the algorithm of Class-aware Adaptive Prototype (CAP) bank.

Architecture of undetected object number prediction network. As shown in Fig. 1, the undetected object number prediction network, Θ_{undet}, is composed of a three-layer MLP (128-64-1) followed by a 'SoftPlus' activation function to guarantee a positive output. Notably, increasing the number of MLP layers or the number of parameters per layer causes the training loss to fluctuate empirically, making it difficult for the network to converge. To enhance the generalizability of Θ_{undet}, we exploit batch normalization [4] and dropout [7] in the network design. We opt Adam [5] optimizer with default parameters to train the network for 80 epochs with a batch size of 16 and an initial learning rate of 0.01. The learning rate decays on [40, 60] epochs with a decay rate of 0.2. Furthermore, due to the scarcity of training data in the active learning setting, we design a novel data augmentation techniques specific to this undetected object number prediction task. As illustrated in Fig. 1, we randomly mask the predicted boxes from the 'RoI Refinement' module to generate different undetected features $\mathbf{u}_i$ and undetected object number $\mathcal{N}(\mathfrak{B}_i, \mathfrak{B}_i^*)$ pairs during the training phase, which encourages the network to better capture the undetectability within point clouds.

Algorithm 1 Class-aware Adaptive Prototype Bank

Input: the unlabeled dataset $\mathcal{D}_U$,
the previous $(r-1)$-th labeled dataset $\mathcal{D}_L^{r-1}$,
the batch size B,
the similarity threshold τ_{sim} (0.3 by default).
Output: the Class-aware Adaptive Prototype (CAP) bank $\mathbf{M} = \{\{\boldsymbol{\mu}_{k,c}\}_{k=1}^{M_c}\}_{c=1}^{C}$.

Initialize the bank $\mathbf{M}$ from all objects in $\mathcal{D}_L^{r-1}$;
for each batch $\{(\mathcal{P}_b, \mathfrak{B}_b^*)\}_{b=1}^{B}$ in $\mathcal{D}_L^{r-1}$ **do**
  for each GT bounding box $(\mathbf{b}_j^*, y_j^*)$ in $\mathfrak{B}_b^*$ **do**
    Extract feature embedding $\mathbf{o}_j$ of bounding box $\mathbf{b}_j^*$, and update the initial prototype $\boldsymbol{\mu}_{1,y_k}$.
  end for
end for
Update the bank $\mathbf{M}$ from all objects in $\mathcal{D}_U$:
for each *predicted* batch $\{(\mathcal{P}_b, \mathfrak{B}_b)\}_{b=1}^{B}$ in $\mathcal{D}_U$ **do**
  for each *predicted* bounding box $(\mathbf{b}_j, y_j)$ in $\mathfrak{B}_b$ **do**
    Extract feature embedding $\mathbf{o}_j$ of predicted bounding box $\mathbf{b}_j$.
    Calculate the IoU_j between predicted bounding box $\mathbf{b}_j$ and its perturbed version.
    for k in $1 : M_{y_j}$ **do**
      Calculate $\text{Sim}(\mathbf{o}_j, \boldsymbol{\mu}_{k,y_j})$ by Eq. 5.
    end for
    Calculate the assignment $\mathcal{A}(\mathbf{o}_j)$ by Eq. 6, and update $\{\boldsymbol{\mu}_{k,y_j}\}_{k=1}^{M_{y_j}}$ by Eq. 7.
  end for
end for

CAP bank	SUN RGB-D		ScanNetV2	
	mAP@.25	mAP@.50	mAP@.25	mAP@.50
l_2-norm	55.6 ↓ 0.5	34.6 ↓ 0.8	60.9 ↓ 0.9	44.3 ↓ 1.6
w/o IoU scores	55.2 ↓ 0.9	34.0 ↓ 1.4	60.1 ↓ 1.7	43.4 ↓ 2.5
Ours	**56.1**	**35.4**	**61.8**	**45.9**

Table 1. Ablation studies of *two* key modifications in the Class-aware Adaptive Prototype (CAP) bank on SUN RGB-D and ScanNetV2 datasets.

Methods	SUN RGB-D					ScanNetV2				
	Sofa	Desk	Chair	Table	Bookshelf	Sofa	Desk	Chair	Table	Bookshelf
RAND	53.57	17.34	71.41	40.27	14.64	85.71	50.12	90.70	54.84	44.55
PPAL	58.82 ↑5.25	19.80 ↑2.46	71.48 ↑0.07	38.27 ↓2.00	20.73 ↑6.09	84.44 ↓1.27	58.54 ↑8.42	91.03 ↑0.33	58.34 ↑3.50	41.12 ↓3.43
KECOR	57.23 ↑3.66	18.91 ↑1.57	70.43 ↓0.98	42.62 ↑2.35	16.01 ↑1.37	85.61 ↓0.10	62.76 ↑12.65	90.25 ↓0.45	57.65 ↑2.81	42.86 ↓1.69
OURS	**62.49** ↑8.92	**29.15** ↑11.81	**77.15** ↑5.74	**48.31** ↑8.04	**24.38** ↑9.74	**85.87** ↑0.16	**65.87** ↑15.75	**91.18** ↑0.48	**58.48** ↑3.64	**53.02** ↑8.47

Table 2. Per-class AP@0.25 (%) scores for specific classes on SUN RGB-D and ScanNetV2 datasets with 10% queried point clouds.

Algorithm of Class-aware Adaptive Prototype Bank. We provide the complete algorithm of Class-aware Adaptive Prototype (CAP) bank, as depicted in Algo. 1. Compared to infinite mixture prototypes (IMP) [1] that was proposed for few-shot learning, we conduct *two* key modifications to suite the indoor 3D detection task, which is highlighted in blue in Algo. 1. Firstly, prior works [8, 9] on active learning for object detection suggest that cosine similarity better captures the relationships between feature embeddings compared to l_2-norm. Therefore, we compute the cosine similarity between the feature embedding $\mathbf{o}_j$ and each of the existing prototypes within the predicted class. Secondly, as many predicted bounding boxes are inaccurate in the early active learning stages, we leverage perturbed IoU score to update the prototypes, mitigating the effects of these invalid bounding boxes.

As shown in Tab. 1, we ablate these *two* key modifications to validate the effectiveness of our CAP bank design. Empirically, cosine similarity outperforms l_2-norm by 0.5% and 0.9% mAP@.25 on SUN RGB-D [6] and ScanNetV2 [3], respectively. Moreover, the perturbed IoU scores are essential for the accurate updates of prototypes, as their absence results in a 1.7% drop in mAP@0.25 on the ScanNetV2 dataset.

3. Additional Experimental Results

We provide the additional experimental results, which include per-class average precision (AP) evaluation and visualizations on SUN RGB-D and ScanNetV2 datasets.

Per-class AP evaluation. Tab. 2 reports per-class AP scores for classes with high intra-class variances. Previous strategies (*e.g.*, **KECOR**) fail to consistently enhance the detection performance of all classes compared to random baseline (see 'Chair' class in Tab. 2), as they do not account for the varying intra-class variances across object categories. In contrast, our approach demonstrates substantial performance improvements across all classes, validating the effectiveness of our design that jointly optimizes intra-class variances and scene-type diversity.

Visualization. Fig. 2 and Fig. 3 present additional qualitative results on SUN RGB-D and ScanNetV2 datasets, respectively. As depicted in Fig. 2, heavy occlusion (*e.g.*, the chair near the window in the first row) and partial visibility (*e.g.*, the red sofa in the second row) complicate accurate detection on the SUN RGB-D dataset, especially when only 10% of the labeled data is used for training. Without referring to the associated images, some of the point clouds are even challenging for humans to correctly recognize, such as the table in the second row and the chair in the bottom-left corner in the fifth row, both of which are partially visible. Nevertheless, it is notable that our proposed method successfully detects most of the objects in these challenging scenarios. Furthermore, our model is able to recognize some unannotated objects, such as the rightmost cabinet near the bed in the fourth row and the table in the fifth row.

In Fig. 3, we further show four visualization examples from the ScanNetV2 dataset, covering various scenarios such as conference room, office, and kitchen. The random sampling method struggles to recognize objects without strong geometric cues (*e.g.*, doors, windows) due to the limited data. In contrast, our proposed method successfully detects most of these challenging objects, such as the leftmost windows in the first row and fourth rows, the door in the second row, and the refrigerators in the third row. These superior detection results indicate that our proposed uncertainty and diversity criteria effectively capture the most informative and representative samples within the indoor 3D datasets, thereby guiding the model to achieve better localization of 3D bounding boxes.

References

[1] Kelsey Allen, Evan Shelhamer, Hanul Shin, and Joshua Tenenbaum. Infinite mixture prototypes for few-shot learning. In *International conference on machine learning*, pages 232–241. PMLR, 2019. 4

[2] Tom M Apostol. *Calculus, Volume 1*. John Wiley & Sons, 1991. 2

[3] Angela Dai, Angel X. Chang, Manolis Savva, Maciej Halber, Thomas Funkhouser, and Matthias Nießner. Scannet: Richly-annotated 3d reconstructions of indoor scenes. In *Proc. Computer Vision and Pattern Recognition (CVPR), IEEE*, 2017. 4

[4] Sergey Ioffe. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *arXiv preprint arXiv:1502.03167*, 2015. 3

[5] Diederik P Kingma. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 3

[6] Shuran Song, Samuel P Lichtenberg, and Jianxiong Xiao. Sun rgb-d: A rgb-d scene understanding benchmark suite. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 567–576, 2015. 4

Figure 2. Additional qualitative comparisons between random sampling and our proposed active learning method on SUN RGB-D dataset.

[7] Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: a simple way to prevent neural networks from overfitting. *The journal of machine learning research*, 15(1):1929–1958, 2014. 3

[8] Jiaxi Wu, Jiaxin Chen, and Di Huang. Entropy-based active learning for object detection with progressive diversity constraint. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9397–9406, 2022. 4

[9] Chenhongyi Yang, Lichao Huang, and Elliot J Crowley. Plug and play active learning for object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 17784–17793, 2024. 4

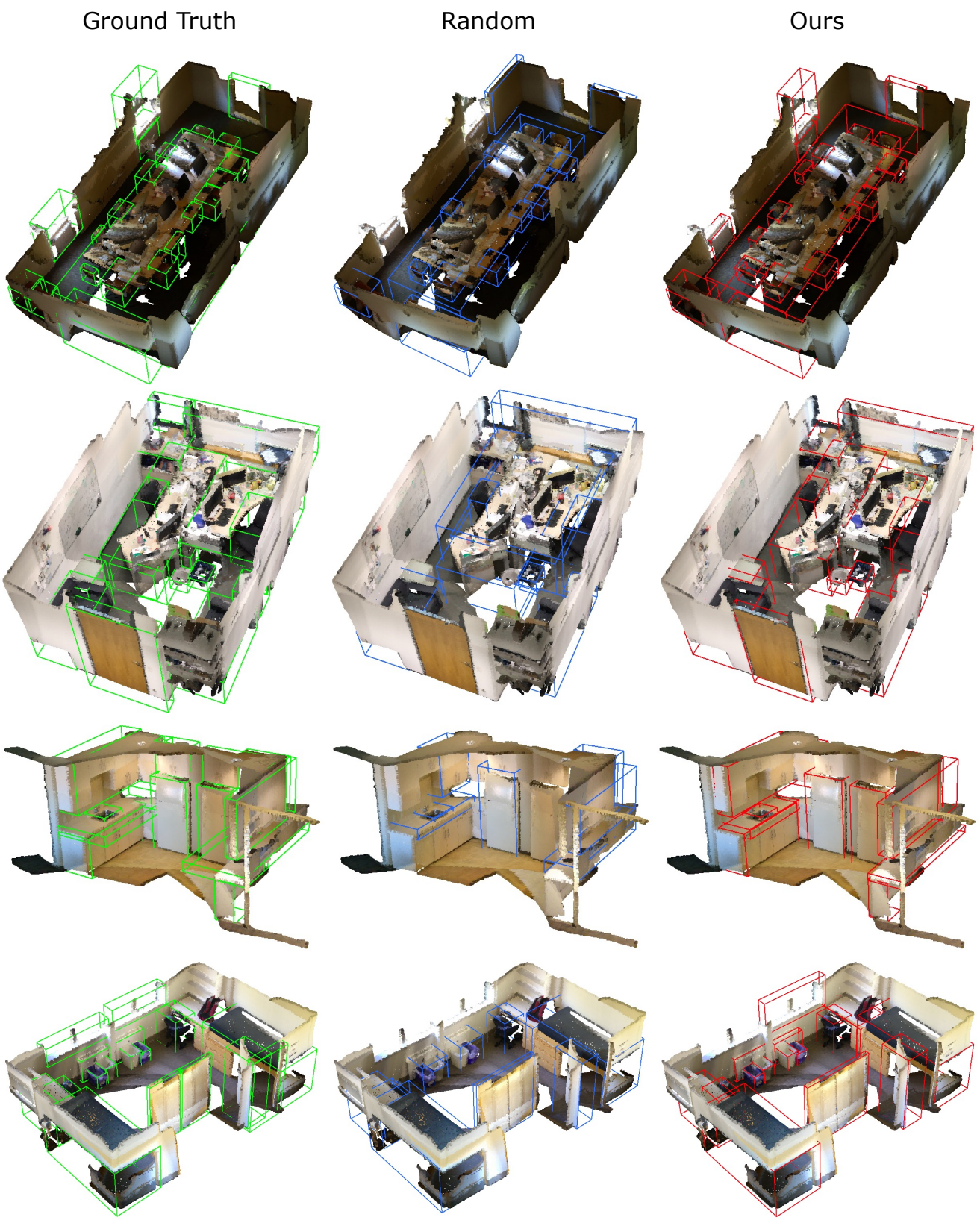

Figure 3. Additional qualitative comparisons between random sampling and our proposed active learning method on ScanNetV2 dataset.