Exploit class-agnostic 3D segmentation masks for Open-Vocabulary queries

Kirill Ivanov
ADL4CV, TUM
kirill.ivanov@tum.de

Abstract

We propose a framework for retrieving class-agnostic 3D regions that match any open-vocabulary text query, without task-specific fine-tuning. The method leverages rich 2D vision–language priors: a target scene is rendered from multiple viewpoints, relevant per-view features are selected according to the query, and these cues are fused back into the volume. A lightweight, class-agnostic 3D segmentation network then aggregates the fused information to produce a coherent, semantically meaningful mask of the queried region.

1. Introduction

Accurately localizing what is where in a 3D scene has a broad range of applications, including robot manipulation and navigation, immersive AR/VR interaction, and digital-content creation. Conventional 3D segmentation pipelines inherit two limitations from their 2D counterparts. First, they operate on a closed vocabulary: the set of categories that can be predicted is fixed at training time. Second, extending that vocabulary typically demands a new round of dense annotation and full network retraining, an impractical requirement for large-scale or rapidly evolving environments.

Open-vocabulary formulations seek to remove both constraints by pairing geometric representations with language-conditioned visual priors. In principle, a free-form text prompt ("red sofa", "all obstacles on the floor") should suffice to highlight previously unseen objects. In practice, current solutions remain hindered by three factors: (i) imprecise alignment between textual cues and 3D geometry, (ii) reliance on multi-view rendering pipelines that inflate computational cost, and (iii) a residual dependence on task-specific fine-tuning for optimal performance.

This work presents a pipeline that retrieves class-agnostic 3D regions corresponding to any open-vocabulary query without additional training. The design follows the decoupling strategy popularized in recent 2D models such as MaskCLIPpp [21]: language grounding is handled entirely in the image domain, whereas spatial coherence is enforced by a separate, class-agnostic 3D backbone trained with a contrastive objective. The resulting system (i) separates semantic grounding from geometric segmentation, (ii) reuses existing foundation models with no task-specific fine-tuning, (iii) exposes a small set of easily interpretable hyper-parameters, and (iv) produces coherent, semantically meaningful 3D masks across a variety of indoor scenes.

2. Related Work

Research on 3D scene segmentation can be organized along two orthogonal axes: (i) supervision regime, ranging from closed vocabularies to open-vocabulary prompts, and (ii) representation, i.e. whether semantic reasoning is carried out purely in 3D or relies on 2D intermediates. The following review adopts this structure.

2.1. Closed-set 3D semantic segmentation

Early fully 3D approaches operate directly on point clouds or voxels and predict a fixed label set. PointNet [13], PointNet++ [14], KPConv [17], MinkowskiNet [3], Cylinder3D [24], and PointTransformer [23] are representative. These models achieve high accuracy on benchmarks such as ScanNet [4] and SemanticKITTI [2] but require dense manual annotations, which are expensive to obtain at scale.

2.2. 2D-guided 3D segmentation

A parallel line of work leverages mature image networks. Multi-view pipelines such as SemanticFusion [11], and Virtual MVF [19] render the scene to color images, predict per-pixel labels with a 2D CNN, and project the predictions back into 3D. This strategy greatly reduces annotation cost because 2D masks are cheaper to collect than 3D labels, yet it still inherits the closed vocabulary of the underlying image model.

2.3. Class-agnostic 3D segmentation

Inspired by the Segment Anything Model (SAM) [5] in 2D, several works have explored category-free grouping in 3D. OmniSeg3D [20] predicts a dense field of object and part proposals in point clouds, while SA-3D [6] lifts SAM

masks into a volumetric representation. These methods offer promptable segmentation without retraining but do not attach semantic labels; an additional alignment step is required for recognition.

2.4. Open-vocabulary 3D segmentation

The scarcity of 3D ground truth has motivated open-vocabulary techniques that couple geometry with pre-trained vision–language models such as CLIP [15]. LSeg [7] treats 2D semantic segmentation as an embedding alignment problem: pixel features are mapped into the CLIP text space, enabling zero-shot masks for arbitrary phrases. Building on this idea, OpenScene [12] back-projects multi-view LSeg features onto the point cloud to obtain per-point embeddings that can be queried with natural language. CLIP-Fog [10] and CLIP-FO3D [1] follow a similar strategy but incorporate sphere-based convolutions to fuse multi-view cues. Segment Anything-3D (SA-3D) [6] combines SAM proposals with CLIP supervision to assign open-set labels to 3D masks. Very recent systems such as OpenMask3D [22] and Point-SAM [16] explicitly fuse SAM-generated masks with CLIP-aligned point features, demonstrating that the synergy of large vision–language models and geometric priors scales to dense point clouds.

Open-vocabulary ideas are also beginning to reshape 3D object detection, with models like OV-VoteNet [9] and OpenMask3D-Det [18] predicting free-form categories without additional training. Quantitative studies report that these label-free systems approach, and in some cases match, the accuracy of fully supervised baselines while adding new concepts at negligible annotation cost.

2.5. Summary

Closed-set 3D networks excel when exhaustive labels are available, whereas 2D-guided methods trade geometric purity for cheaper supervision. Class-agnostic models provide promptable masks but lack semantics. Open-vocabulary approaches bridge this gap by importing language grounding from 2D foundation models, thereby supporting free-form queries across segmentation, detection, and other 3D perception tasks. The present work follows the latter direction, combining image-level vision–language cues with a lightweight, class-agnostic 3D backbone to obtain flexible open-vocabulary segmentation in reconstructed scenes.

3. Method

Overview The proposed algorithm combines a frozen vision–language image model with a class-agnostic 3D representation. A text prompt is first resolved to candidate image regions in a subset of input views. These 2D observations are then projected into the reconstructed scene and consolidated into a small set of prototypical 3D descriptors that define the queried region (Figure 1).

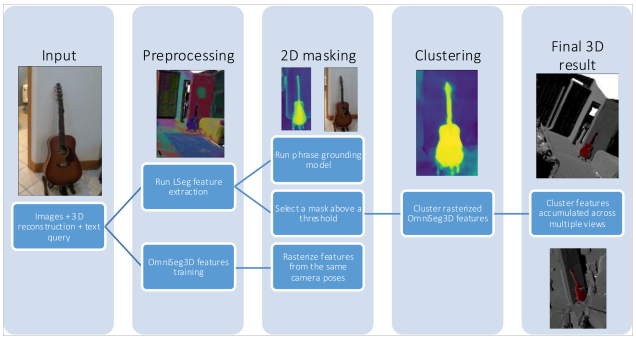


Figure 1. Overview of the proposed open-vocabulary 3D segmentation pipeline. (1) A class-agnostic feature field is first learned on the reconstructed scene and rasterised into every camera view. (2) A frozen vision–language network (LSeg or Grounding DINO) converts a text prompt into binary masks on the subset of RGB images that contain the queried object. (3) Rasterized 3D features inside each mask are clustered with HDBSCAN; cluster centroids are retained. (4) Centroids from all views are merged in a second HDBSCAN pass to obtain a compact set of feature prototypes. (5) Cosine similarity to these prototypes produces the final 3D mask or highlights the region in newly rendered views.

Inputs Even though many datasets such as ScanNet [4] come a sequence of RGB-D images, we assume a pre-built 3D reconstruction is given together with the RGB images and calibrated camera poses that were used to create it. The only user input is an open-vocabulary text query such as "red sofa." Note that in a case of a ScanNet dataset we are also provided with a cleaned mesh.

Outputs The algorithm returns a set of prototype vectors in the learned 3D feature space. Points, voxels, or rendered pixels whose descriptors lie within a specified cosine distance of any prototype are labeled as belonging to the target object or region.

3.1. Approach

3D feature extraction. Scene-consistent, class-agnostic features are first learned on the reconstructed geometry. The implementation employs OmniSeg3D in its Gaussian-splatting variant, although the method is compatible with other 3D representations. Once trained, the features are rasterized into every input view, producing one feature map per camera pose. The features are designed to capture local shape continuity rather than semantic categories.

2D query masks. Given an open-vocabulary text prompt, a frozen vision–language network is applied to the RGB views that plausibly contain the target object; the remaining views can be skipped. The network returns a binary image region for each processed frame. Two alternatives have

been evaluated: LSeg, which thresholds CLIP-based similarity scores, and Grounding DINO, which supplies rectangular detections that are treated as masks. The sole requirement is that the model output a binary region aligned with the textual query.

Per-view clustering. For every selected view the rasterized 3D features located inside the binary mask are extracted and clustered with HDBSCAN. Only the cluster mean is retained, reducing noise and data volume. HDBSCAN is used because it introduces a single, interpretable parameter, the minimum cluster size, which implicitly reflects the expected object scale.

Global consolidation. The cluster means from all processed views are merged in a second HDBSCAN pass to obtain the final prototype set. These prototypes can be used either to select Gaussian splats in the reconstruction or to highlight corresponding regions in newly rendered views by evaluating cosine similarity in the OmniSeg3D feature space.

3.2. Hyperparameters

Several hyper-parameters are inherited from the image-level models and from the two clustering stages. The image models (LSeg or Grounding DINO) expose a confidence threshold that is applied to their similarity scores or detection logits. Because a narrowly specified text prompt ("red leather sofa") generally yields fewer false positives than a broad one ("furniture"), the threshold is set as a function of prompt specificity; higher values are used for concrete queries, lower values for generic queries.

The clustering stage introduces two additional parameters. In the per-view step the minimum cluster size controls the smallest group of projected 3D descriptors considered valid. Larger values suppress spurious selections, which are frequent with similarity-based masks such as those produced by LSeg. After all views have been processed, the second clustering pass operates on the set of per-view centroids. Here the minimum cluster size represents the number of distinct views in which the object must be detected before it is accepted.

To recover the final mask, either as Gaussian splats in the reconstruction or as highlighted pixels in a rendered image, a similarity search is performed in the OmniSeg3D feature space. The cosine-similarity threshold for this search is fixed at 0.95 throughout all experiments.

4. Evaluation

4.1. Qualitative results

Dataset Experiments are conducted on scenes 0 and 2 of the ScanNet benchmark [4].

Class	LSeg	Ours(LSeg)	GDINO	Ours(GDINO)
bed	**0.805400**	0.685096	0.150349	0.223188
refrigerator	**0.740700**	0.648852	0.172766	0.235231
coffee table	0.214200	0.300615	0.131786	**0.585103**
curtains	**0.668400**	0.568797	0.179124	0.408395
couch	**0.744300**	0.496864	0.221643	0.246086
floor	**0.721600**	0.455712	0.227481	0.100537
tv	0.320500	**0.438510**	0.189654	0.407430
backpack	0.343400	0.388006	0.206748	**0.437556**
ceiling	**0.456412**	0.388316	0.149557	0.067867
couch	**0.413625**	0.204179	0.260813	0.349405
doors	0.365846	0.475618	0.295334	**0.511147**
fireplace	0.158921	**0.417614**	0.049485	0.026661

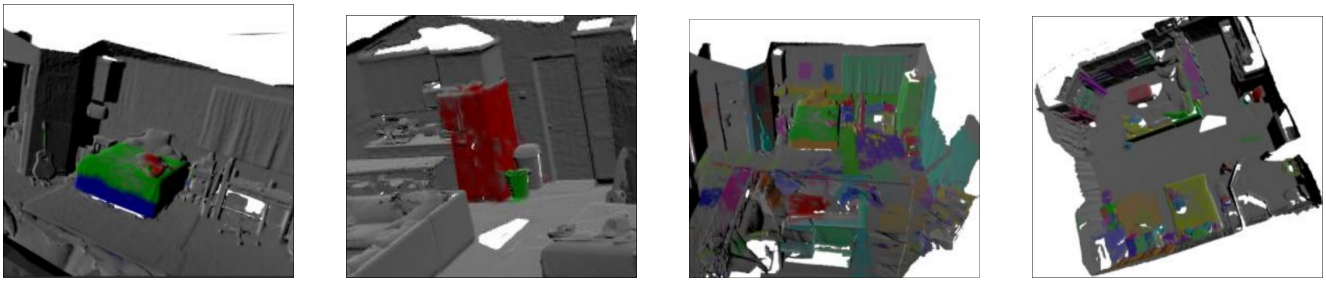
Table 1. Per-class Intersection-over-Union (IoU) on ScanNet scene0. The table compares two image-level baselines (LSeg and Grounding DINO) with the same models combined with the proposed 3D clustering ("Ours"). Scores are computed in the original camera views; only a representative subset of classes is shown. Boldface marks the highest IoU for each class.

Model selection Class-agnostic 3D descriptors are obtained with the Gaussian-splatting variant of OmniSeg3D, using 16-dimensional feature vectors. Two alternative image backbones provide the language-conditioned masks: LSeg [7] and Grounding DINO [8].

Metrics For every class present in a scene, intersection-over-union (IoU) is computed by aggregating predictions across all camera views. Four numbers are reported per class: the IoU achieved by the image model alone (either LSeg or Grounding DINO) and the IoU obtained when the same image model is coupled with the 3D clustering stage. The confidence threshold for each image model is tuned to its optimal value before computing the scores.

4.2. Quantitative results

A joint 2D+3D pipeline should, in principle, profit from view-to-view consistency: an object visible only in part of one image ought to be recoverable through its full 3D extent. Results on ScanNet scene0 (Table 1) only partly validate this assumption. When LSeg is used in isolation it already attains the highest Intersection-over-Union (IoU) for several large or visually distinctive objects, for example bed (0.81) and refrigerator (0.74), suggesting that its similarity maps are sufficiently precise whenever the target covers a substantial image area. Adding the 3D clustering to LSeg yields mixed outcomes: the extra step improves IoU on objects whose image masks are noisy yet spatially coherent across views (tv, fireplace), but it can degrade scores on broad planar regions such as ceiling and floor, where the clustering may fragment a previously continuous mask.



(a) Bed parts. (b) Refrigerator. (c) Fabric. (d) Furniture.

Figure 2. Qualitative segmentation results for four representative query types. (a) Bed—the method retrieves all disjoint parts of the object, including mattress and frame. (b) Refrigerator—a single, compact instance is isolated despite nearby clutter. (c) Fabric—a material-level query highlights soft surfaces such as cushions and curtains. (d) Furniture—a broad, category-level query selects multiple semantically related items within the scene.

Grounding DINO exhibits a different pattern. Its raw outputs, limited to rectangular detections, are comparatively coarse; however, the 3D fusion stage refines these boxes into tighter masks and markedly improves IoU, most noticeably for coffee table ($0.13 \rightarrow 0.59$) and curtains ($0.18 \rightarrow 0.41$). Overall, the evidence indicates that the 2D + 3D fusion is most beneficial when the initial image-level predictions are coarse or heavily affected by view-specific noise, whereas a precise per-pixel baseline such as LSeg may already capture large, well-defined objects without additional refinement.

Figure 2 illustrates the range of query types that can be addressed under the open-vocabulary setting. In Fig. 2a the query "bed" is resolved into several spatially disjoint parts (*e.g.*, mattress and frame); these components are preserved in the OmniSeg3D feature space because they originate from separate Segment Anything masks. Fig. 2d demonstrates that a query may also refer to a semantically related group of objects rather than a single instance. In Fig. 2b, objects located in close proximity to the target, here a trash can neighboring a refrigerator, are occasionally included owing to noisy image-level predictions, which lowers the IoU metric. Nevertheless, such extraneous points form distinct cluster centers and can, in principle, be excluded by a downstream post-processing module.

4.3. Ablation studies

Table 1 indicates that LSeg achieves competitive performance overall but under-segments small objects, whereas Grounding DINO detects most instances yet introduces more false positives and shows reduced precision for abstract queries (e.g., "fabric"). The ablation study in Table 2 further suggests that the proposed clustering pipeline is comparatively insensitive to variations in the threshold parameter.

Class	LSeg threshold	LSeg IoU	Ours IoU
wall	0.7	**0.4279**	0.4335
wall	0.75	0.1309	**0.3436**
cabinets	0.7	**0.2951**	0.2209
cabinets	0.75	0.1685	**0.2958**

Table 2. Sensitivity of segmentation accuracy to the LSeg confidence threshold (ScanNet scene0). Intersection-over-Union (IoU) is reported for LSeg alone and for the proposed 2D + 3D pipeline ("Ours") at two threshold settings (0.70 and 0.75). Raising the threshold sharply lowers LSeg's IoU, whereas the fused method remains comparatively stable, indicating reduced dependence on this parameter.

5. Conclusion

The pipeline does not match the overall scores reported for OpenScene. Nevertheless, it provides a flexible design that separates language grounding from 3D segmentation and lets users swap in different 2D or 3D backbones while adjusting only a few clear hyper-parameters. Although its masks are less precise, they remain stable across scenes and query types, making the method a practical front-end for tasks where robustness and adaptability matter more than pixel-level accuracy.

References

[1] Ahmed Abdelreheem, Helisa Dhamo, and Nassir et al. Navab. Clip-fo3d: A zero-shot baseline for open-vocabulary 3d scene understanding. *arXiv preprint arXiv:2310.09279*, 2023. 3

[2] Jens Behley, Michael Garbade, and Andres et al. Milioto. Semantickitti: A dataset for semantic scene understanding of lidar sequences. In *Proc. IEEE/CVF International Conf. on Computer Vision (ICCV)*, pages 9297–9307, 2019. 2

[3] Christopher Choy, JunYoung Gwak, and Silvio Savarese. 4d spatio-temporal convnets: Minkowski convolutional neural networks. In *Proceedings of the IEEE Conference on Com-*

puter Vision and Pattern Recognition, pages 3075–3084, 2019. 2

[4] 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. 2, 3, 4

[5] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C. Berg, Wan-Yen Lo, Piotr Dollár, and Ross Girshick. Segment anything. *arXiv:2304.02643*, 2023. 2

[6] Xiang Lai, Yiming Wang, Andreas Geiger, and Angela Dai. Segment anything in 3d. *arXiv preprint arXiv:2306.00984*, 2023. 2, 3

[7] Boyi Li, Kilian Q Weinberger, Serge Belongie, Vladlen Koltun, and Rene Ranftl. Language-driven semantic segmentation. In *International Conference on Learning Representations*, 2022. 3, 4

[8] Shilong Liu, Zhaoyang Zeng, Tianhe Ren, Feng Li, Hao Zhang, Jie Yang, Chunyuan Li, Jianwei Yang, Hang Su, Jun Zhu, and Lei Zhang. Grounding dino: Marrying dino with grounded pre-training for open-set object detection, 2023. 4

[9] Sheng Lu, Jingwei Zhang, and Angela Dai. Ov-votenet: Open-vocabulary 3d object detection. *arXiv preprint arXiv:2309.01421*, 2023. 3

[10] Pan Luo, Siyuan Huang, Xinjie Wang, and Christopher B. Choy. Clip-fog: Clip features on grids for open-vocabulary 3d scene understanding. *arXiv preprint arXiv:2310.08543*, 2023. 3

[11] John McCormac, Ankur Handa, Stefan Leutenegger, and Andrew Davison. Semanticfusion: Dense 3d semantic mapping with convolutional neural networks. In *IEEE Int. Conf. on Robotics and Automation (ICRA)*, pages 4628–4635, 2017. 2

[12] Songyou Peng, Kyle Genova, Chiyu "Max" Jiang, Andrea Tagliasacchi, Marc Pollefeys, and Thomas Funkhouser. Openscene: 3d scene understanding with open vocabularies. In *CVPR*, 2023. 3

[13] Charles R Qi, Hao Su, Kaichun Mo, and Leonidas J Guibas. Pointnet: Deep learning on point sets for 3d classification and segmentation. *arXiv preprint arXiv:1612.00593*, 2016. 2

[14] Charles R Qi, Li Yi, Hao Su, and Leonidas J Guibas. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. *arXiv preprint arXiv:1706.02413*, 2017. 2

[15] Alec Radford, Jong Wook Kim, and Tao et al. Xu. Learning transferable visual models from natural language supervision. In *Proc. International Conf. on Machine Learning (ICML)*, pages 8748–8763, 2021. 3

[16] Xiaoyu Tang, Muxin Liu, and Pan et al. Zhou. Pointsam: Point cloud segmentation anything. *arXiv preprint arXiv:2312.00398*, 2023. 3

[17] Hugues Thomas, Charles R. Qi, Jean-Emmanuel Deschaud, Beatriz Marcotegui, François Goulette, and Leonidas J. Guibas. Kpconv: Flexible and deformable convolution for point clouds. *Proceedings of the IEEE International Conference on Computer Vision*, 2019. 2

[18] Ziyi Wang, Xiangyu Chen, and Vladlen Koltun. Openmask3d-det: A simple recipe for open-vocabulary 3d object detection. *arXiv preprint arXiv:2404.01245*, 2024. 3

[19] Ilija M. Yalniz, Arda Genc, Marc Pollefeys, and Federico Tombari. Virtual multi-view fusion for 3d semantic segmentation. In *Proc. IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)*, pages 13904–13914, 2022. 2

[20] Haiyang Ying, Yixuan Yin, Jinzhi Zhang, Fan Wang, Tao Yu, Ruqi Huang, and Lu Fang. Omniseg3d: Omniversal 3d segmentation via hierarchical contrastive learning. *arXiv preprint arXiv:2311.11666*, 2023. 2

[21] Quan-Sheng Zeng, Yunheng Li, Daquan Zhou, Guanbin Li, Qibin Hou, and Ming-Ming Cheng. High-quality mask tuning matters for open-vocabulary segmentation, 2025. 2

[22] Yuhang Zhang, Weiyang Li, and Tai et al. Wang. Openmask3d: Open-vocabulary 3d instance segmentation with masked foundation models. *arXiv preprint arXiv:2403.01234*, 2024. 3

[23] Hengshuang Zhao, Li Jiang, Jiaya Jia, Philip HS Torr, and Vladlen Koltun. Point transformer. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 16259–16268, 2021. 2

[24] Xinge Zhu, Hui Zhou, Tai Wang, Fangzhou Hong, Yuexin Ma, Wei Li, Hongsheng Li, and Dahua Lin. Cylindrical and asymmetrical 3d convolution networks for lidar segmentation. *arXiv preprint arXiv:2011.10033*, 2020. 2

Exploit class-agnostic 3D segmentation masks for Open-Vocabulary queries

Supplementary Material

Class	LSeg	Ours(LSeg)	GDINO	Ours(GDINO)
bed	0.805400	0.685096	0.150349	0.223188
refrigerator	0.740700	0.648852	0.172766	0.235231
coffee table	0.214200	0.300615	0.131786	0.585103
curtains	0.668400	0.568797	0.179124	0.408395
couch	0.744300	0.496864	0.221643	0.246086
floor	0.721600	0.455712	0.227481	0.100537
tv	0.320500	0.438510	0.189654	0.407430
backpack	0.343400	0.388006	0.206748	0.437556
desk	0.386400	0.406701	0.017831	0.000659
shelf	0.329200	0.403034	0.021425	0.038184
toaster oven	0.151700	0.000335	0.193160	0.402917
table	0.277200	0.311513	0.124005	0.056605
trash can	0.189900	0.281279	0.104117	0.122400
doors	0.275700	0.231765	0.111503	0.272946
cabinets	0.295100	0.249453	0.147502	0.247581
wall	0.528900	0.245484	0.115402	0.225909
stools	0.384000	0.235229	0.068714	0.015685
pillows	0.289400	0.203767	0.021570	0.018629
sink	0.220400	0.202048	0.025559	0.008621
ceiling	0.711900	0.197428	0.030369	0.021243
toilet	0.300300	0.186918	0.215597	0.061634
bicycle	0.291800	0.155447	0.276031	0.046529
doorframe	0.119700	0.083004	0.021281	0.017084
microwave	0.260300	0.056254	0.050755	0.000000
window	0.243800	0.000000	0.044370	0.004157
clock	0.391600	0.000000	0.092011	0.000000

Table 3. IoU scores computed over initial camera poses on a ScanNet scene 0.

Class	GDINO	LSeg
cabinets	0.30	0.70
ceiling	0.50	0.75
couch	0.70	0.70
doorframe	0.60	0.72
doors	0.60	0.65
floor	0.40	0.72
microwave	0.80	0.75
pillows	0.30	0.75
refrigerator	0.80	0.70
shelf	0.40	0.75
shoe	0.30	0.65
stick	0.40	0.65
table	0.40	0.75
trash can	0.60	0.70
tv	0.80	0.75
wall	0.30	0.65
window	0.50	0.80

Table 4. Optimal threshold levels on different classes on a ScanNet scene 0.

category	threshold level	iou omni	iou dino
backpack	0.800000	0.437556	0.206748
bed	0.400000	0.223188	0.150349
bicycle	0.800000	0.046529	0.276031
cabinets	0.300000	0.247581	0.147502
ceiling	0.400000	0.021243	0.030369
clock	0.800000	0.000000	0.092011
coffee table	0.800000	0.585103	0.131786
couch	0.400000	0.246086	0.221643
curtains	0.500000	0.408395	0.179124
desk	0.500000	0.000659	0.017831
dish rack	0.600000	0.001477	0.016504
doorframe	0.300000	0.017084	0.021281
doors	0.600000	0.272946	0.111503
floor	0.300000	0.100537	0.227481
guitar	0.800000	0.456141	0.233798
guitar case	0.400000	0.000000	0.001617
kitchen counter	0.400000	0.046925	0.055392
laundry basket	0.800000	0.165061	0.100356
microwave	0.800000	0.000000	0.050755
mirror	0.600000	0.001710	0.012498
nightstand	0.700000	0.003168	0.026231
open kitchen cabinet	0.300000	0.010623	0.029947
pillows	0.300000	0.018629	0.021570
refrigerator	0.500000	0.235231	0.172766
scale	0.800000	0.000000	0.154340
shelf	0.300000	0.038184	0.021425
shoe	0.300000	0.001973	0.001819
shower	0.500000	0.025339	0.119279
sink	0.600000	0.008621	0.025559
stick	0.300000	0.001021	0.002210
stools	0.300000	0.015685	0.068714
table	0.500000	0.056605	0.124005
tissue box	0.400000	0.010267	0.027188
toaster	0.600000	0.000345	0.025507
toaster oven	0.800000	0.402917	0.193160
toilet	0.800000	0.061634	0.215597
trash can	0.500000	0.122400	0.104117
tv	0.800000	0.407430	0.189654
wall	0.300000	0.225909	0.115402
window	0.500000	0.004157	0.044370

Table 5. Results with DINO on the 2nd ScanNet scene.

category	threshold	omni iou	lseg iou
bag	0.800000	0.002526	0
bottles	0.900000	0.000000	0
cabinets	0.800000	0.135456	0
ceiling	0.900000	0.388316	0
closet	0.850000	0.017407	0
coffee maker	0.800000	0.000886	0
controller	0.850000	0.003507	0
couch	0.900000	0.204179	0
doorframe	0.850000	0.047500	0
doors	0.800000	0.475618	0
fan	0.800000	0.062497	0
fireplace	0.800000	0.417614	0

Table 6. Results with Ours LSeg version on the 2nd ScanNet scene.

category	threshold	lseg iou
wall	0.700000	0.113635
wall	0.750000	0.143723
wall	0.800000	0.221234
wall	0.850000	0.288572
wall	0.900000	0.048668
stack of chairs	0.700000	0.197109
stack of chairs	0.750000	0.305202
stack of chairs	0.800000	0.321871
stack of chairs	0.850000	0.000013
stack of chairs	0.900000	0.000000
floor	0.700000	0.219698
floor	0.750000	0.252404
floor	0.800000	0.289907
floor	0.850000	0.254676
floor	0.900000	0.114480
doors	0.700000	0.121596
doors	0.750000	0.210236
doors	0.800000	0.365846
doors	0.850000	0.163282
doors	0.900000	0.000000
couch	0.700000	0.086430
couch	0.750000	0.136211
couch	0.800000	0.226888
couch	0.850000	0.398511
couch	0.900000	0.413625
cabinets	0.700000	0.085733
cabinets	0.750000	0.111837
cabinets	0.800000	0.151894
cabinets	0.850000	0.117980
cabinets	0.900000	0.009071
pillows	0.700000	0.046464
pillows	0.750000	0.076180
pillows	0.800000	0.148636
pillows	0.850000	0.231943
pillows	0.900000	0.054687
closet	0.700000	0.006677
closet	0.750000	0.009756
closet	0.800000	0.019338
closet	0.850000	0.065192
closet	0.900000	0.002099
shoe	0.700000	0.004470
shoe	0.750000	0.006576
shoe	0.800000	0.010154
shoe	0.850000	0.000304
shoe	0.900000	0.000000
bottles	0.700000	0.000257
bottles	0.750000	0.000400
bottles	0.800000	0.001235
bottles	0.850000	0.039104
bottles	0.900000	0.048986
ottoman	0.700000	0.123261
ottoman	0.750000	0.067684
ottoman	0.800000	0.000000
ottoman	0.850000	0.000000
ottoman	0.900000	0.000000
radiator	0.700000	0.051795
radiator	0.750000	0.058768
radiator	0.800000	0.043083
radiator	0.850000	0.000000