3D Instance Segmentation via Enhanced Spatial and Semantic Supervision

Salwa Al Khatib[1] Mohamed El Amine Boudjoghra[1] Jean Lahoud[1] Fahad Shahbaz Khan[1,2]
[1]Mohamed Bin Zayed University of Artificial Intelligence (MBZUAI), UAE
[2] Linköping University, Sweden
{salwa.khatib,mohamed.boudjoghra,jean.lahoud,fahad.khan}@mbzuai.ac.ae

Abstract

3D instance segmentation has recently garnered increased attention. Typical deep learning methods adopt point grouping schemes followed by hand-designed geometric clustering. Inspired by the success of transformers for various 3D tasks, newer hybrid approaches have utilized transformer decoders coupled with convolutional backbones that operate on voxelized scenes. However, due to the nature of sparse feature backbones, the extracted features provided to the transformer decoder are lacking in spatial understanding. Thus, such approaches often predict spatially separate objects as single instances. To this end, we introduce a novel approach for 3D point clouds instance segmentation that addresses the challenge of generating distinct instance masks for objects that share similar appearances but are spatially separated. Our method leverages spatial and semantic supervision with query refinement to improve the performance of hybrid 3D instance segmentation models. Specifically, we provide the transformer block with spatial features to facilitate differentiation between similar object queries and incorporate semantic supervision to enhance prediction accuracy based on object class. Our proposed approach outperforms existing methods on the validation sets of ScanNet V2 and ScanNet200 datasets, establishing a new state-of-the-art for this task.

1. Introduction

In recent years, remarkable advances have been made in 3D scene understanding, owing to the rapid development of 3D sensors (Kinect, RealSense, Velodyne laser scanner, among others) and the increase in the number of large-scale datasets. Data-driven deep learning models with a focus on either point or sparse voxel approaches have been widely explored. 3D instance segmentation on point clouds is the task of simultaneously localizing and recognizing 3D objects from a set of 3D points. The desired output is a set of binary masks representing the objects with their corresponding semantic categories. This perception task serves as the basis for a wide variety of applications, including autonomous driving, mixed and virtual reality, and robot navigation.

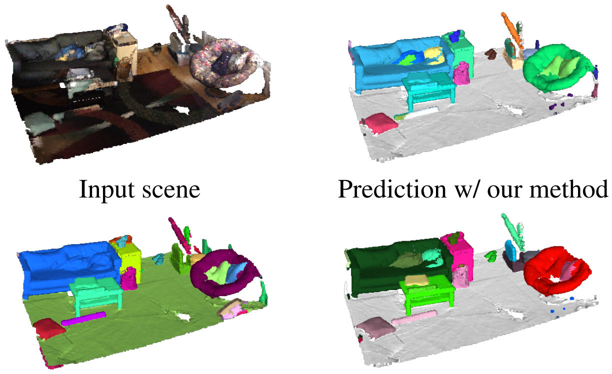

Input scene
Prediction w/ our method
Semantic instance ground truth
Prediction w/ Mask3D

Figure 1. Samples predictions of our approach on scenes from the ScanNet200 [35] dataset. Our proposed approach utilizes both semantic and spatial supervision to generate distinct instance labels for objects in a given scene, by processing a 3D point cloud as input. This enables the model to generate instance masks for objects that are similar in appearance but located in different positions, resulting in highly accurate and comprehensive labeling.

2D instance segmentation is a critical computer vision task that involves identifying and distinguishing individual objects or instances within an image and assigning semantic classes to them. Unlike semantic segmentation, which assigns a label to each pixel in an image, instance segmentation aims to accurately identify each object in an image and provide a unique mask or bounding box for each one. Thus, instance segmentation lies at the intersection of object detection and semantic segmentation. Numerous studies have been conducted in this area, with many works focusing on top-down approaches [4, 9, 6, 15], in which instance-level proposals are generated initially to predict instance masks that are later classified into one of the recognized classes. One popular example of these approaches is BMask R-CNN [9], which is an extension of Mask R-CNN. It was developed to address the challenges associated with segmenting objects with complex shapes and fine details. BMask R-CNN introduces a boundary-sensitive branch to the Mask R-CNN architecture, which predicts object boundaries in addition to object masks.

In contrast to 2D instance segmentation, bottom-up pipelines dominate 3D instance segmentation, where, generally speaking, point-level semantic labels are learned and then spatially close points of the same classes are grouped together into instances. Thus, there has been extensive work on developing grouping strategies for this purpose [21, 7, 24, 43]. The remarkable results of transformers have motivated numerous researchers to explore their usage in the instance segmentation task. To overcome the challenge of CNNs' insufficient long-range dependencies, hybrid-based techniques used attention mechanisms along with CNN-based backbones for feature extraction. One such method is presented in [38] for 3D instance segmentation. It relies on the successive and iterative refinement of queries to learn masks and semantic labels by attending to multi-scale features obtained from a CNN backbone. This approach has been proven to achieve state-of-the-art results, owing to CNNs' proficiency in producing features for objects of varying scales and the attention mechanism's capacity to capture contextual information. Nevertheless, these approaches do not allow enough information exchange between the encoder and the decoder because of the structural differences between the transformer blocks and the sparse convolutional backbone.

In this paper, we propose to improve the learned features for the modules of a hybrid-based instance segmentation technique that combines a sparse convolutional backbone with a transformer decoder for query refinement. Enhanced supervision that targets the encoder specifically is proposed to achieve this aim. Given the 3D geometry of a scene, the model labels all the geometry that belongs to a single object with a unique label and assigns a class to this object. In particular, we propose a learning technique to regress per-voxel coordinates and learn per-voxel semantic labels in the encoder. Despite the benefits of using 3D point cloud voxelization to enable regular 2D convolution on 3D point clouds, the location and geometry information of 3D objects may be lost. This arises from the fact that the decoder exclusively relies on the encoder features, derived solely from the RGB color of the voxel. Moreover, the process of voxelization can compound this issue by grouping small objects into a limited number of voxels. Consequently, these aggregated voxels fail to completely capture the geometry of the original objects. However, the utilization of the coordinates in voxel space after the sparse quantization step, which consists of the X, Y, and Z values, can aid in recovering the lost information.

To this end, our contributions are as follows:

- We explore various ways of improving information exchange between the convolutional encoder and transformer decoder of a hybrid 3D instance segmentation technique (i) spatial and semantic supervision in the 3D encoder, (ii) appending raw coordinates to 3D backbone features before feeding them to the decoder.
- We enrich the highest-resolution features used for mask prediction with existing voxel positions to assist in the prediction of higher quality and more precise masks.
- We achieve state-of-the-art performance on ScanNet V2 [12] ($+1.3$ mAP_{50}) and ScanNet200 [35] ($+2.7$ mAP_{50}).

2. Related Work

In this section, we present some of the works related to 3D point clouds, 3D instance segmentation, as well as recent paper which use transformers as building blocks in their architecture for the 3D point clouds instance segmentation task.

2.1. Deep Learning for 3D Point Clouds

Prior to the deep learning era, early methods [1, 2, 3, 37, 36] extracted hand-crafted features from point clouds based on statistical analysis. On the other hand, recent methods resort to learning feature extraction from the point cloud.

Early deep learning-based methods, e.g. PointNet [33, 34], processed point clouds directly through Multi-Layer Perceptrons (MLPs) and max pooling operations to capture both local and global 3D geometries. Other approaches [31, 30] opted to voxelized point clouds to go from an unordered point set to regular 3D-ordered vectors compatible with 3D convolutions. Some other approaches [41, 27, 10], on the other hand, aimed to exploit the sparsity of voxelized 3D point clouds to reduce the computation complexity of the regular 3D convolution operation using sparse convolution, which was first introduced in 2D [25]. In order to make the latter operation user-friendly, a universal sparse convolution and auto-differentiation framework was introduced with MinkowskiEngine in [10]. even though voxel-based methods are highly efficient, they suffer from severe information loss since the voxelization process performs aggressive downsampling on the points [11, 16].

2.2. 3D Instance Segmentation

The problem of instance segmentation for 3D scenes has been explored and addressed in numerous works; some methods adopted the top-down approach, where proposals are generated from a stream, then combined with the local features generated from a parallel stream to predict the final masks for each instance. In this direction, 3D-BoNet [45] predicts the final masks using regressed bounding boxes and point-wise semantic labels generated from

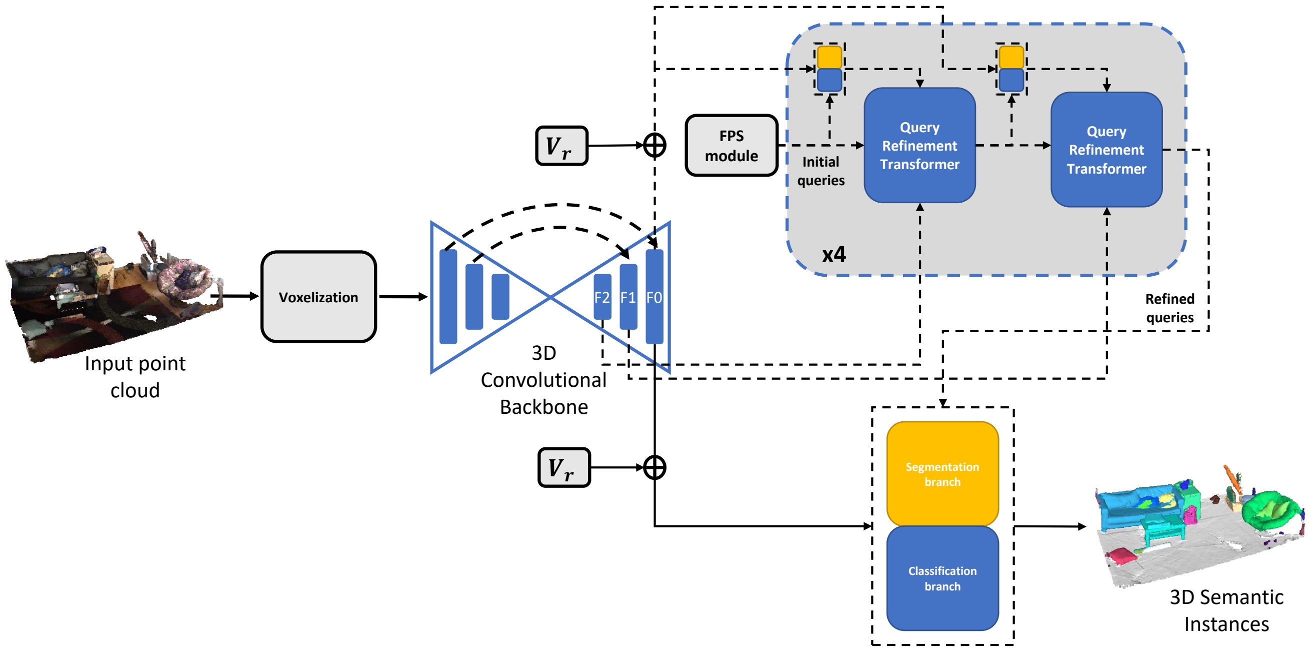


Figure 2. This figure illustrates the architecture of our proposed instance segmentation technique which is composed of (1) a 3D convolutional backbone, (2) a farthest point sampling module, (3) a query refinement strategy, and (4) a prediction head made up of a segmentation branch and a classification branch. Here, V_r represents the voxel-level coordinates.

parallel pipelines. 3D-SIS [19] follows the same reasoning as [45], but uses RGB-D scans for training, while NeuralBF [40] uses learnable bilateral filtering and shows that this can further improve the instance proposal generation. An alternative strategy is a bottom-up method, whereby instance prediction is accomplished through semantic segmentation, followed by instances prediction in the same stream. Numerous methods were proposed [21, 22, 19, 26], [22] employed Multi-Task Metric Learning and clustering to generate the instance proposals, while [7] suggested a different framework through hierarchical aggregation. In order to benefit from the two approaches, SoftGroup [44] proposes a two-stage architecture, the first is bottom-up-based, for semantic information extraction, while the second stage is to generate instance proposals through refinement in a top-down manner.

2.3. Transformers

Transformers were first introduced in [42] and have since gained significant popularity in the field of natural language processing due to their ability to capture similarities between embeddings of various categories. The computer vision community later adopted this mechanism, and its potential was first demonstrated in [14]. The global contextual learning capability of transformers has motivated computer vision researchers to utilize them in 3D object detection [28, 32, 39, 3], 3D semantic segmentation [23], and, more recently, in 3D instance segmentation [38]. In contrast to the conventional top-down approaches, Mask3D [38] and Group-Free [28] have used a transformer for query refinement in the instance proposal generation stage while using a multi-scale sparse convolutional backbone for feature extraction. Mask3D addresses some major problems in previous 3D instance segmentation approaches. State-of-the-art methods heavily rely on hand-crafted voting mechanisms that require the points to each vote for manually-tuned geometric properties of the instance such as centers, bounding boxes, or occupancy. Furthermore, these models are trained with proxy losses based on the point votes, so the learning objective is not to explicitly predict the instance masks. Even though Mask3D proved to be significantly more powerful compared to the other state-of-the-art methods, it still has some shortcomings. A systematic mistake often observed in the qualitative results is that objects of the same class having similar geometry but existing far apart in the scene are combined into a single instance. The authors attribute this to the attention mechanism that simultaneously attends to the entire point cloud.

3. Methodology

Fig. 2 illustrates our proposed instance segmentation architecture composed of a *3D Convolutional backbone* that extracts features from the voxel space, a *furthest point sampling* method to generate K object queries, *stacked attention modules* to refine those queries based on sampled point features, and a *mask prediction module* that predicts instance heatmaps and class probabilities given full-resolution feature maps extracted by the backbone and the corresponding raw coordinates for each voxel. Furthermore, intermediary supervision is imposed on the encoder from existing labelled information.

3.1. Problem Formulation

3D point cloud processing. Given an input point cloud $P \in \mathbb{R}^{N\times 6}$, where N is the number of points of which each holds RGB colors and 3D full-resolution coordinates, the goal is to predict K binary masks, each classified into one of C classes. P is first voxelized into $V_c \in \mathbb{R}^{M\times 3}$ voxels, each holding average RGB color information of the points it includes. Given a sparse 3D convolutional backbone with a symmetric encoder and decoder, R feature maps are extracted from V_c at different resolutions. These feature maps form a hierarchy of multiple scales.

Query top-down refinement. Since it is impractical to follow a top-down strategy in choosing initial object queries in 3D due to the fact that the search space is too huge, we follow the recent practice of sampling object queries from the point cloud in a bottom-up approach via farthest point sampling (FPS) [34]. This method has been used to downsample a point cloud or to choose initial object candidates from a point cloud. This simple method relies on randomly choosing a point and then iteratively sampling the farthest point from the already chosen points until the number of desired points is chosen.

The goal is to refine the $K^{'}$ initial instance queries for L stages to end up at K spatially accurate and scene-relevant queries. After sampling initial object queries with FPS, a transformer decoder is used to iteratively refine these queries by allowing them to (1) cross-attend to masked scene features and (2) self-attend among each other. Thus, the transformer is composed of stacked multi-head *masked* cross-attention and multi-head self-attention. The masked variant of cross attention is used, as in [8], to force the transformer to ignore out-of-context features by design. Self-attention among queries is necessary to establish inter-query communication and avoid duplicate or overlapping instance masks.

Classification branch. The classification branch takes as input instance queries X and applies a softmax function on them after projecting them to $C + 1$ dimensions with a linear layer. This facilitates the prediction of a semantic class for each of the K' queries. It is worth noting that the instances are classified into one of the C classes or an ignored class which helps filter out the queries from K' to K.

Segmentation branch. To predict the binary foreground masks, $F_{r=0} \in \mathbb{R}^{M_0\times D}$, the highest-resolution feature map with dimension D, is used alongside $K^{'}$ instance queries, where $K^{'} \geq K$. The binary masks $B \in \{0,1\}^{M,K}$ are calculated as follows:

$$B = \{b_{i,j} = [\sigma\big(F_0 f_{mask}(X)^T\big)_{i,j} > \tau]\} \tag{1}$$

where $f_{mask}(.)$ is a linear layer that maps the queries X to D dimensions, σ is a sigmoid function, and τ is a thresholding scalar $\in [0,1]$. τ is set to 0.5.

Learning objectives. The whole network can be trained in an end-to-end manner with a weighted dual-task loss as follows:

$$\mathcal{L}_{decoder} = \sum_{l}^{L} \lambda_{mask}\mathcal{L}^{(l)}_{mask} + \lambda_{class}\mathcal{L}^{(l)}_{class} \tag{2}$$

where $\mathcal{L}_{mask}$ is composed of a binary cross-entropy loss over both the foreground and background of the mask and Dice loss [13], $\mathcal{L}_{class}$ is a multi-class cross-entropy loss on the instance level, and L is the number of query refinement steps.

3.2. Enhanced Supervision

We argue that a shortcoming in existing solutions is that limiting the supervision at the decoder level of the architecture hinders the ability of the network, specifically the convolutional encoder, to learn semantically and spatially rich features. This limitation arises primarily from the weak spatial and geometric information encoded in the 3D backbone, due to sparse convolution and voxelization. To mitigate this, we utilize readily available information including per-voxel semantic labels and per-voxel raw coordinates to introduce intermediary supervision that targets the convolutional backbone of the network. The goal is to encode both spatial and semantic understanding into the features that are ultimately used in the query refinement process and in the prediction of the final set of class-labeled masks.

Spatial Supervision. Hybrid models, which take as input a set of voxels with features that encode the RGB color do not include features to represent the points' locations. As a result, the transformer block ends up taking as input the features from the highest level of the backbone, which does not necessarily encode the voxels' positions, given the sparse nature of point clouds. In order to fill this gap, spatial supervision takes into account the output voxels' locations when predicting the mask used for refining the instance queries, a vital property in the context of 3D localization tasks. The first learning target imposed on the convolutional encoder is the prediction of the per-voxel raw coordinates $V_r \in \mathbb{R}^{M\times 3}$ from the highest-resolution features F_0. A projection head $f_{spatial}(.)$ composed of a single linear layer is used to map F_0 from $M\times D$ to $M\times$ 3, to finally use it for the heatmap prediction step.

Semantic Supervision. The second learning objective aims to enhance the semantic understanding of the network by guiding the network to learn per-voxel class labels instead of limiting its semantic context to the instance

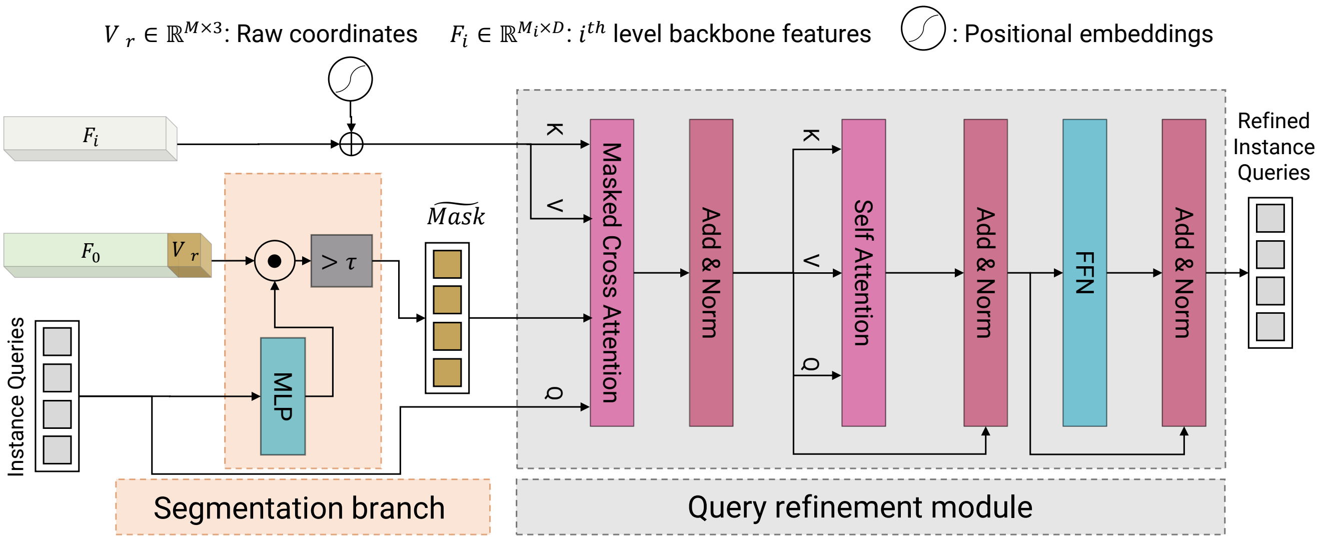


Figure 3. Architecture of the transformer module inspired by [5]. The multi-head cross attention is masked with the predicted heatmaps. These heatmaps are obtained from the interaction between (1) the instance queries and (2) point features to which V_r, the voxel-level raw coordinates, were appended. Refer to Eq. 5.

level. The goal of this supervision is to bridge the semantic gap between point and instance identity, and have the voxel-level label assist in better instance-level label prediction. To achieve this, F_0 is projected to $M \times (C+1)$ using $f_{semantic}(.)$, a single-layer MLP, after which a softmax function is applied. The ground truth labels utilized for computing the loss in this case are the same labels used to supervise the instance class prediction task, where voxels not belonging to any instance are labeled with an ignore class. Thus, no additional supervision signal is required.

The loss objective for the sparse convolutional encoder is:

$$\mathcal{L}_{encoder} = \lambda_{semantic}\mathcal{L}_{semantic} + \lambda_{spatial}\mathcal{L}_{spatial} \quad (3)$$

where $L_{semantic}$ is a cross-entropy loss and $L_{spatial}$ is a mean squared error loss. Thus, the overall objective function of the network is:

$$\mathcal{L} = \mathcal{L}_{decoder} + \mathcal{L}_{encoder} \quad (4)$$

This objective function can be used to train the network in an end-to-end fashion.

3.3. Enhancing Spatial Localization

Further, we argue that the features produced by the sparse convolutional backbone are lacking in terms of spatial information necessary for the accurate prediction and localization of the masks in 3D space. To this end, we aim to enrich the information used to predict the binary masks. Eq. 1 is altered as follows. V_r is used alongside $F_{r=0} \in \mathbb{R}^{M_0\times D}$ and the $K^{'}$ instance queries to predict the binary masks as shown in Fig. 3. More specifically, B is computed using the following:

$$B = \{b_{i,j} = [\sigma\big((F_0 \oplus V_r) f_{mask}(X)^T\big)_{i,j} > \tau]\} \quad (5)$$

where f_{mask} maps the queries X to $D+3$ dimensions. This primarily influences the MLP in each of the L segmentation branches that projects the queries X to a common dimensionality with the point features F_0.

4. Experiments

In the following section, we first present the used datasets and the evaluation protocol (Sec. 4.1). After this, we provide details on the implementation of the method (Sec. 4.2). Quantitative results (Sec. 4.3) and qualitative results (Sec. 4.4) follow. Finally, ablation studies are presented in Sec. 4.5.

4.1. Datasets and Evaluation Protocol

We evaluate our approach on ScanNet V2 [12] and ScanNet200 [35], widely-used indoor 3D instance segmentation datasets. We adopt the standard data splits for these datasets.

ScanNet V2 [12] is a dataset composed of 1513 richly annotated 3D reconstructed indoor scenes. It includes hundreds of reconstructed rooms such as hotels, libraries, and offices. Provided labels include per-point instances, semantic labels, and 3D bounding boxes for 18 classes. The metric used for evaluation of this dataset is mean average precision

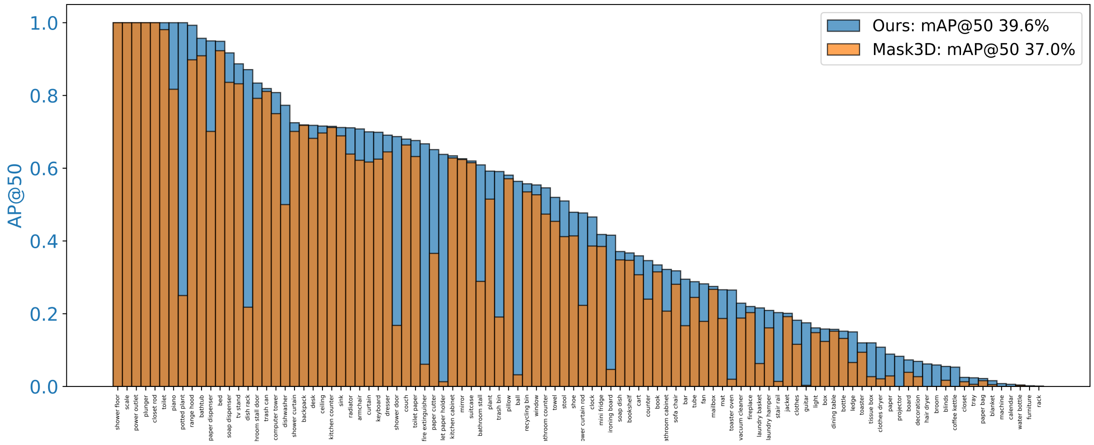


Figure 4. Performance comparison with Mask3D [38], the currently best performing 3D instance segmentation approach. Per-class AP_{50} results for 100 classes of ScanNet200 are presented.

(mAP) over various Intersection over Union (IoU) thresholds: mAP at IoU of 25%, 50%, and $[50\%, 95\%]$ (averaged at 5% steps).

ScanNet200 [35] is a more recent extension of ScanNet V2 that includes 200 class labels for the same scenes. ScanNet200 facilitates the evaluation of an algorithm's performance on naturally occurring imbalanced data since it follows a long-tail distribution with 66 head classes, 68 common classes, and 66 tail classes. We use the same evaluation protocol followed for ScanNet V2 to evaluate on this dataset.

4.2. Implementation Details

We conduct all of our experiments on a single NVIDIA Tesla A100 GPU device. Data augmentations on the scenes include horizontal flipping, random rotations along the z-axis, elastic distortion, random scaling, color jitter, and brightness/contrast alterations. The convolutional encoder is a Minkowski Res16UNet34C [11] used in Mask3D [38]. We train for 600 epochs with AdamW [29] optimizer and a learning rate of $1e-4$. the scheduler used is the one-cycle learning rate scheduler. We set λ_{BCE} to 5, λ_{dice} to 5, λ_{CE} to 5, $\lambda_{semantic}$ to 2, and finally $\lambda_{spatial}$ to 2. The voxelization is done at $2cm$ voxel size.

4.3. Quantitative Results

Table 2 shows the results of our approach and the most prominent methods on validation set of ScanNet V2 [12] in terms of mAP, mAP_{50}, and mAP_{25}. The proposed method achieves the highest score, with a 1.3% margin on mAP_{50}. Table 1 further shows per-class AP_{50} for the 18 classes in the dataset. Our approach achieves the best performance in 14 out of the 18 categories.

Similarly, we report metrics on the validation set of ScanNet200 [35]. Table 3 shows that the proposed method surpasses the state-of-the-art by a significant margin of 2.7% on mAP_{50}. Since this dataset exhibits a long-tailed distribution, we also report on mAP_{50} per data split of head, common, and tail in Table 4. Our approach significantly improves the performance of tail classes specifically, with an improvement of 5.9%. Additionally, Fig. 4 presents per-class AP_{50} on 100 classes out of the 198 in the dataset as compared to Mask3D [38].

4.4. Qualitative Results

In Fig. 5, we present qualitative 3D instance segmentation results on ScanNet V2 [12] as compared to the ground truth masks and those of Mask3D. The showcased scenes are from the validation splits of this dataset, and they are diverse in terms of the type of challenges they exhibit, e.g. background clutter, similar objects, and scanning artifacts. It can be observed that, with the proposed method, more precise instance segmentation masks are obtained as compared to the current state-of-the-art. The highlighted sections clearly outline examples where Mask3D predicts merged instances that are far apart while our method, which explicitly encodes the spatial locations of features, is able to distinguish between those instances.

4.5. Ablation Studies

We ablate the effects of the key designs on ScanNet200 as shown in Table 5. We observe a boost in performance as each component is added, and the best results are attained with the three components combined which gave a 2.7% increase in mAP_{50} and 2.0% increase in mAP on the validation set. It can be observed that carrying out the mask prediction with appending of V_r to the point features gives the biggest individual boost in performance, which shows

Method	bathtub	bed	bkshelf	cabinet	chair	counter	curtain	desk	door	picture	fridge	s. curtain	sink	sofa	table	toilet	window	other	mAP_{50}
MTML [22]	70.8	54.0	21.9	14.5	79.2	0.8	39.9	14.2	32.4	10.9	42.1	64.3	36.4	48.8	42.7	96.5	32.7	21.5	40.2
PointGroup [21]	80.5	69.6	54.9	48.1	87.7	22.4	44.9	41.6	42.0	37.7	37.2	64.4	61.1	71.5	62.9	98.3	46.2	53.0	56.9
DyCo3D [18]	77.4	70.4	48.4	52.3	90.2	34.9	47.5	52.3	40.5	44.7	51.5	70.3	74.3	69.6	94.8	47.2	46.2	56.4	61.0
Mask3D [38]	87.0	**79.1**	**66.7**	65.5	94.4	63.1	**73.6**	63.5	**74.4**	**65.8**	**77.1**	**71.4**	**77.5**	**78.0**	**82.8**	**100.0**	**65.1**	73.2	73.7
(Ours)	**90.1**	76.7	54.5	**67.6**	**95.0**	**63.7**	71.0	**66.3**	73.5	**65.8**	**77.1**	**71.4**	**77.5**	**78.0**	**82.8**	**100.0**	**65.1**	**73.9**	**75.0**

Table 1. 3D instance segmentation results in terms of AP_{50} scores on ScanNet V2 [12]. The table shows the AP_{50} score of all semantic categories as well as the average score, which is sorted in ascending order of mAP_{50}. The proposed method outperforms the current state-of-the-art on the average AP_{50}.

Method	Val		
	mAP	mAP_{50}	mAP_{25}
3D-SIS [19]	-	18.7	35.7
GSPN [46]	19.3	37.8	53.4
MTML [22]	20.3	40.2	55.4
3D-MPA	35.5	59.1	72.4
DyCo3D	35.4	57.6	72.9
PointGroup [21]	34.8	56.7	71.3
MaskGroup [47]	27.4	42.0	63.3
OccuSeg [17]	44.2	60.7	71.9
SSTNet [24]	49.4	64.3	74.0
HAIS [7]	43.5	64.1	75.6
SoftGroup [43]	46.0	67.6	78.9
Mask3D [38]	55.2	73.7	83.5
(Ours)	**56.1**	**75.0**	**83.7**

Table 2. State-of-the-art comparison on the ScanNet V2 [12] 3D instance segmentation dataset. This table shows mAP with various IoU thresholds of our method as well as all recent methods.

Method	Val		
	mAP	mAP_{50}	mAP_{25}
CSC [20]	-	25.24	-
LGround [35]	-	26.09	-
Mask3D [38]	27.4	37.0	42.3
(Ours)	**29.4**	**39.7**	**44.9**

Table 3. mAP for 3D instance segmentation on validation split of ScanNet 200 [35] with various IoU thresholds of our method as well as other recent methods.

Method	Val mAP_{50}		
	head	common	tail
Mask3D [38]	54.9	30.6	23.2
(Ours)	**55.5**	**32.8**	**29.1**

Table 4. mAP_{50} for 3D Instance Segmentation on head, common, and tail class splits of ScanNet 200 [35] of our method and Mask3D [38].

Spatial Supervision	Semantic Supervision	Mask prediction w/ V_r	mAP	mAP_{50}
✓	×	×	27.9	37.1
×	×	✓	28.9	39.2
✓	×	✓	29.0	39.6
×	✓	✓	29.0	38.4
✓	✓	✓	**29.4**	**39.7**

Table 5. Ablation study on the impact of each component on the performance of our proposed method on the ScanNet 200 [35] 3D instance segmentation dataset, specifically the validation set.

the importance of explicitly encoding per-voxel spatial information into the mask prediction module.

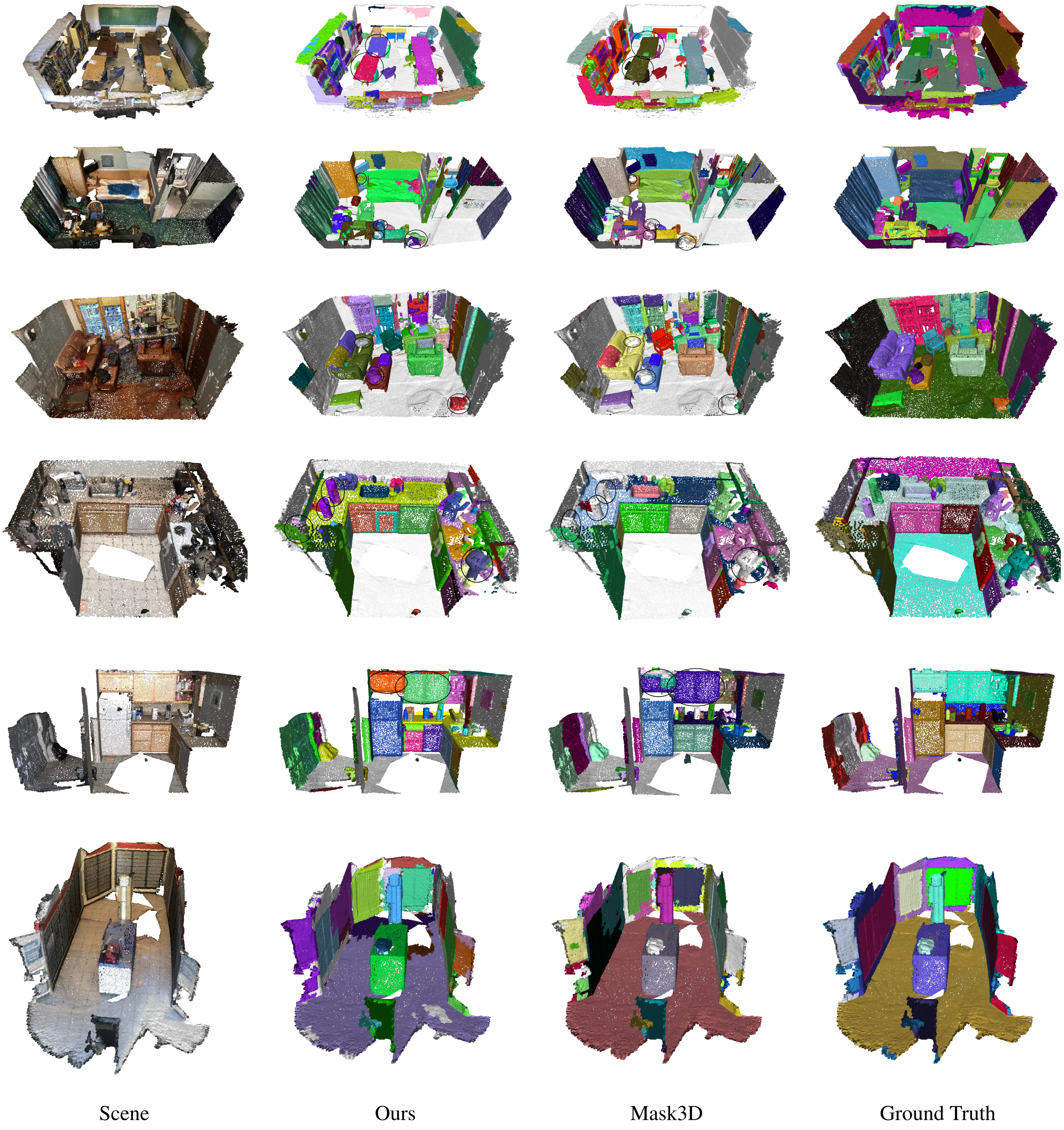
In our final design, we use RGB information in the voxelized point cloud for subsequent feature extraction. Next, we ablate the choice of input in Table 6. We experiment with (1) feeding the full-resolution coordinates as input, (2) the coordinates in conjunction with the RGB colors, and (3) the RGB colors with spatial supervision. Empirically, spatial supervision gave a better performance as can be shown in Table 6.

Experiment	$mAP50$
Raw coordinates as input	36.2
Raw coordinates and color as input	36.6
Spatial supervision with color as input	**38.0**

Table 6. Results for various spatial supervision approaches on ScanNet200 [35] validation set.

5. Conclusion

In this paper, we present a simple yet effective way to learn semantically and spatially rich features for better localization of instance masks in 3D space using a hybrid instance segmentation architecture. The method re-purposes existing supervision signals related to the perceptual task at hand to assist in the semantic and location information flow between the sparse convolutional backbone and the transformer decoder. In addition, readily-available voxel positions are fed to the mask prediction branch to maximize its



Scene Ours Mask3D Ground Truth

Figure 5. Qualitative instance segmentation results on ScanNet V2 scenes from the validation split. This figure shows the original input scene as a textured mesh, the scene with its ground truth masks, predictions of Mask3D [38], and predictions with our method.
Key regions are highlighted in black circles.

performance, specifically in accurately localizing the masks in 3D space, this approach proved to be more effective than using RGB and raw coordinates as input features for the backbone. The proposed method achieves state-of-the-art performance on ScanNet V2 and ScanNet200 validation splits.

References

[1] Luıs A Alexandre. 3d descriptors for object and category recognition: a comparative evaluation. In *Workshop on Color-Depth Camera Fusion in Robotics at the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Vilamoura, Portugal*, volume 1, page 7. Citeseer, 2012.

[2] Mathieu Aubry, Ulrich Schlickewei, and Daniel Cremers. The wave kernel signature: A quantum mechanical approach to shape analysis. In *2011 IEEE international conference on computer vision workshops (ICCV workshops)*, pages 1626–1633. IEEE, 2011.

[3] Xuyang Bai, Zeyu Hu, Xinge Zhu, Qingqiu Huang, Yilun Chen, Hongbo Fu, and Chiew-Lan Tai. Transfusion: Robust lidar-camera fusion for 3d object detection with transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1090–1099, 2022.

[4] Daniel Bolya, Chong Zhou, Fanyi Xiao, and Yong Jae Lee. Yolact: Real-time instance segmentation. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 9157–9166, 2019.

[5] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part I 16*, pages 213–229. Springer, 2020.

[6] Kai Chen, Jiangmiao Pang, Jiaqi Wang, Yu Xiong, Xiaoxiao Li, Shuyang Sun, Wansen Feng, Ziwei Liu, Jianping Shi, Wanli Ouyang, et al. Hybrid task cascade for instance segmentation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4974–4983, 2019.

[7] Shaoyu Chen, Jiemin Fang, Qian Zhang, Wenyu Liu, and Xinggang Wang. Hierarchical aggregation for 3d instance segmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 15467–15476, 2021.

[8] Bowen Cheng, Ishan Misra, Alexander G Schwing, Alexander Kirillov, and Rohit Girdhar. Masked-attention mask transformer for universal image segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1290–1299, 2022.

[9] Tianheng Cheng, Xinggang Wang, Lichao Huang, and Wenyu Liu. Boundary-preserving mask r-cnn. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XIV 16*, pages 660–676. Springer, 2020.

[10] Christopher Choy, JunYoung Gwak, and Silvio Savarese. 4d spatio-temporal convnets: Minkowski convolutional neural networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 3075–3084, 2019.

[11] Christopher Choy, JunYoung Gwak, and Silvio Savarese. 4d spatio-temporal convnets: Minkowski convolutional neural networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 3075–3084, 2019.

[12] Angela Dai, Angel X Chang, Manolis Savva, Maciej Halber, Thomas Funkhouser, and Matthias Nießner. Scannet: Richly-annotated 3d reconstructions of indoor scenes. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5828–5839, 2017.

[13] Ruoxi Deng, Chunhua Shen, Shengjun Liu, Huibing Wang, and Xinru Liu. Learning to predict crisp boundaries. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 562–578, 2018.

[14] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.

[15] Yuxin Fang, Shusheng Yang, Xinggang Wang, Yu Li, Chen Fang, Ying Shan, Bin Feng, and Wenyu Liu. Instances as queries. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 6910–6919, 2021.

[16] Benjamin Graham, Martin Engelcke, and Laurens Van Der Maaten. 3d semantic segmentation with submanifold sparse convolutional networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 9224–9232, 2018.

[17] Lei Han, Tian Zheng, Lan Xu, and Lu Fang. Occuseg: Occupancy-aware 3d instance segmentation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 2940–2949, 2020.

[18] Tong He, Chunhua Shen, and Anton Van Den Hengel. Dyco3d: Robust instance segmentation of 3d point clouds through dynamic convolution. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 354–363, 2021.

[19] Ji Hou, Angela Dai, and Matthias Nießner. 3d-sis: 3d semantic instance segmentation of rgb-d scans. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4421–4430, 2019.

[20] Ji Hou, Benjamin Graham, Matthias Nießner, and Saining Xie. Exploring data-efficient 3d scene understanding with contrastive scene contexts. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15587–15597, 2021.

[21] Li Jiang, Hengshuang Zhao, Shaoshuai Shi, Shu Liu, Chi-Wing Fu, and Jiaya Jia. Pointgroup: Dual-set point grouping for 3d instance segmentation. In *Proceedings of the IEEE/CVF conference on computer vision and Pattern recognition*, pages 4867–4876, 2020.

[22] Jean Lahoud, Bernard Ghanem, Marc Pollefeys, and Martin R Oswald. 3d instance segmentation via multi-task metric learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9256–9266, 2019.

[23] Xin Lai, Jianhui Liu, Li Jiang, Liwei Wang, Hengshuang Zhao, Shu Liu, Xiaojuan Qi, and Jiaya Jia. Stratified transformer for 3d point cloud segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8500–8509, 2022.

[24] Zhihao Liang, Zhihao Li, Songcen Xu, Mingkui Tan, and Kui Jia. Instance segmentation in 3d scenes using semantic superpoint tree networks. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2783–2792, 2021.

[25] Baoyuan Liu, Min Wang, Hassan Foroosh, Marshall Tappen, and Marianna Pensky. Sparse convolutional neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 806–814, 2015.

[26] Chen Liu and Yasutaka Furukawa. Masc: Multi-scale affinity with sparse convolution for 3d instance segmentation. *arXiv preprint arXiv:1902.04478*, 2019.

[27] Zhijian Liu, Haotian Tang, Shengyu Zhao, Kevin Shao, and Song Han. Pvnas: 3d neural architecture search with point-voxel convolution. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(11):8552–8568, 2021.

[28] Ze Liu, Zheng Zhang, Yue Cao, Han Hu, and Xin Tong. Group-free 3d object detection via transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2949–2958, 2021.

[29] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017.

[30] Daniel Maturana and Sebastian Scherer. 3d convolutional neural networks for landing zone detection from lidar. In *2015 IEEE international conference on robotics and automation (ICRA)*, pages 3471–3478. IEEE, 2015.

[31] Daniel Maturana and Sebastian Scherer. Voxnet: A 3d convolutional neural network for real-time object recognition. In *2015 IEEE/RSJ international conference on intelligent robots and systems (IROS)*, pages 922–928. IEEE, 2015.

[32] Xuran Pan, Zhuofan Xia, Shiji Song, Li Erran Li, and Gao Huang. 3d object detection with pointformer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7463–7472, 2021.

[33] Charles R Qi, Hao Su, Kaichun Mo, and Leonidas J Guibas. Pointnet: Deep learning on point sets for 3d classification and segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 652–660, 2017.

[34] Charles Ruizhongtai Qi, Li Yi, Hao Su, and Leonidas J Guibas. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. *Advances in neural information processing systems*, 30, 2017.

[35] David Rozenberszki, Or Litany, and Angela Dai. Language-grounded indoor 3d semantic segmentation in the wild. In *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XXXIII*, pages 125–141. Springer, 2022.

[36] Radu Bogdan Rusu, Nico Blodow, and Michael Beetz. Fast point feature histograms (fpfh) for 3d registration. In *2009 IEEE international conference on robotics and automation*, pages 3212–3217. IEEE, 2009.

[37] Radu Bogdan Rusu, Nico Blodow, Zoltan Csaba Marton, and Michael Beetz. Aligning point cloud views using persistent feature histograms. In *2008 IEEE/RSJ international conference on intelligent robots and systems*, pages 3384–3391. IEEE, 2008.

[38] Jonas Schult, Francis Engelmann, Alexander Hermans, Or Litany, Siyu Tang, and Bastian Leibe. Mask3d for 3d semantic instance segmentation. *arXiv preprint arXiv:2210.03105*, 2022.

[39] Hualian Sheng, Sijia Cai, Yuan Liu, Bing Deng, Jianqiang Huang, Xian-Sheng Hua, and Min-Jian Zhao. Improving 3d object detection with channel-wise transformer. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2743–2752, 2021.

[40] Weiwei Sun, Daniel Rebain, Renjie Liao, Vladimir Tankovich, Soroosh Yazdani, Kwang Moo Yi, and Andrea Tagliasacchi. Neuralbf: Neural bilateral filtering for top-down instance segmentation on point clouds. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 551–560, 2023.

[41] Haotian Tang, Zhijian Liu, Shengyu Zhao, Yujun Lin, Ji Lin, Hanrui Wang, and Song Han. Searching efficient 3d architectures with sparse point-voxel convolution. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXVIII*, pages 685–702. Springer, 2020.

[42] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

[43] Thang Vu, Kookhoi Kim, Tung M Luu, Thanh Nguyen, and Chang D Yoo. Softgroup for 3d instance segmentation on point clouds. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2708–2717, 2022.

[44] Thang Vu, Kookhoi Kim, Tung M Luu, Thanh Nguyen, and Chang D Yoo. Softgroup for 3d instance segmentation on point clouds. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2708–2717, 2022.

[45] Bo Yang, Jianan Wang, Ronald Clark, Qingyong Hu, Sen Wang, Andrew Markham, and Niki Trigoni. Learning object bounding boxes for 3d instance segmentation on point clouds. *Advances in neural information processing systems*, 32, 2019.

[46] Li Yi, Wang Zhao, He Wang, Minhyuk Sung, and Leonidas J Guibas. Gspn: Generative shape proposal network for 3d instance segmentation in point cloud. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3947–3956, 2019.

[47] Min Zhong, Xinghao Chen, Xiaokang Chen, Gang Zeng, and Yunhe Wang. Maskgroup: Hierarchical point grouping and masking for 3d instance segmentation. In *2022 IEEE International Conference on Multimedia and Expo (ICME)*, pages 1–6. IEEE, 2022.