

# PanoOcc: Unified Occupancy Representation for Camera-based 3D Panoptic Segmentation

Yuqi Wang<sup>1,2</sup> Yuntao Chen<sup>3</sup>✉ Xingyu Liao<sup>†</sup> Lue Fan<sup>2</sup> Zhaoxiang Zhang<sup>1,2,3,4</sup>✉

<sup>1</sup> School of Artificial Intelligence, University of Chinese Academy of Sciences (UCAS)

<sup>2</sup> CRIPAC, MAIS, Institute of Automation, Chinese Academy of Sciences (CASIA)

<sup>3</sup> Centre for Artificial Intelligence and Robotics (HKISI.CAS) <sup>4</sup> Shanghai AI Laboratory

{wangyuqi2020, fanlue2019, zhaoxiang.zhang}@ia.ac.cn cheniyuntao08@gmail.com

## Abstract

*Comprehensive modeling of the surrounding 3D world is crucial for the success of autonomous driving. However, existing perception tasks like object detection, road structure segmentation, depth & elevation estimation, and open-set object localization each only focus on a small facet of the holistic 3D scene understanding task. This divide-and-conquer strategy simplifies the algorithm development process but comes at the cost of losing an end-to-end unified solution to the problem. In this work, we address this limitation by studying camera-based 3D panoptic segmentation, aiming to achieve a unified occupancy representation for camera-only 3D scene understanding. To achieve this, we introduce a novel method called PanoOcc, which utilizes voxel queries to aggregate spatiotemporal information from multi-frame and multi-view images in a coarse-to-fine scheme, integrating feature learning and scene representation into a unified occupancy representation. We have conducted extensive ablation studies to validate the effectiveness and efficiency of the proposed method. Our approach achieves new state-of-the-art results for camera-based semantic segmentation and panoptic segmentation on the nuScenes dataset. Furthermore, our method can be easily extended to dense occupancy prediction and has demonstrated promising performance on the Occ3D benchmark. The code will be made available at <https://github.com/Robertwyq/PanoOcc>.*

## 1. Introduction

Holistic 3D scene understanding is vital in autonomous driving. The capability to perceive the environment, identify and categorize objects, and contextualize their positions in the 3D space of the scene is fundamental for developing

a safe and reliable autonomous driving system.

Recent advancements in camera-based Bird’s Eye View (BEV) methods have shown great potential in enhancing 3D scene understanding. By integrating multi-view observations into a unified BEV space, these methods have achieved remarkable success in tasks such as 3D object detection [24, 26, 34, 51, 55], BEV semantic segmentation [16, 42, 67], and vector map construction [28, 35]. However, existing perception tasks have certain limitations as they primarily focus on specific aspects of the scene. Object detection is primarily concerned with identifying foreground objects, BEV semantic segmentation only predicts the semantic map on the BEV plane, and vector map construction emphasizes the static road structure of the scene. To address these limitations, there is a need for a more comprehensive and integrated paradigm for 3D scene understanding. In this paper, we propose *camera-based 3D panoptic segmentation*, which aims to encompass all the elements within the scene in a unified representation for the 3D output space.

However, directly utilizing Bird’s Eye View (BEV) features for camera-based panoptic segmentation leads to poor performance due to the omission of finer geometry details, such as shape and height information, which are crucial for decoding fine-grained 3D structures. This limitation motivates us to explore a more effective 3D feature representation. Occupancy representation has gained popularity as it effectively describes various elements in the scene, including open-set objects (e.g., debris), irregular-shaped objects (e.g., articulated trailers, vehicles with protruding structures), and special road structures (e.g., construction zones). Therefore, a burst of recent methods [4, 18, 25, 27, 38, 54, 57, 65] have focused on dense semantic occupancy prediction. However, simply lifting 2D to 3D occupancy representation has been considered inefficient in terms of memory cost. This limitation has driven methods like TPVFormer [18] to split the 3D representation into

✉ Corresponding author.

† Independent Researcher.

three 2D planes. Although these methods attempt to mitigate the memory issue, they still struggle to capture the complete 3D information and may experience reduced performance. Moreover, these existing works primarily concentrate on the semantic comprehension of the scene and do not tackle instance-level discrimination. Fine-grained foreground segmentation is crucial for 3D perception.

In this work, we propose a novel method called *PanoOcc*, which seamlessly integrates 3D object detection and semantic segmentation in a joint-learning framework, enhancing the understanding of the 3D environment comprehensively. Both detection and segmentation performance can benefit from this joint-learning framework. Our approach employs voxel queries to learn a unified occupancy representation. This occupancy representation is learned in a coarse-to-fine scheme, solving the problem of memory cost and significantly enhancing efficiency. We then take a step further to explore the sparse nature of 3D space and propose an occupancy sparsify module. This module progressively prunes occupancy to a spatially sparse representation during the coarse-to-fine upsampling, greatly boosting memory efficiency. Our contributions are summarized as follows:

- We introduce *camera-based 3D panoptic segmentation* as a new paradigm for holistic 3D scene understanding, which utilizes multi-view images to create a unified occupancy representation for the 3D scene. This allows us to jointly model object detection and semantic segmentation within a single end-to-end model, leading to a more cohesive and holistic understanding of the scene.
- Our proposed framework, PanoOcc, employs a *coarse-to-fine scheme* to learn the unified occupancy representation from multi-frame and multi-view images. We demonstrate that utilizing 3D voxel queries within a coarse-to-fine learning scheme is both effective and efficient. This approach could be further enhanced for memory efficiency by integrating an occupancy sparsify module to make the scheme spatially sparse.
- Experiments on the nuScenes dataset show that our approach achieves state-of-the-art performance on camera-based 3D semantic segmentation and panoptic segmentation. Furthermore, our approach can extend to dense occupancy prediction and has shown promising performance on the Occ3D benchmark.

## 2. Related Work

**Camera-based 3D Perception.** Camera-based 3D perception has received extensive attention in the autonomous driving community due to its cost-effectiveness and rich visual attributes. Previous methods perform 3D object detection and map segmentation tasks independently. Recent BEV-based methods unify these tasks on the problem of feature view transformation from image space to BEV

space. One line of works follows the lifting paradigm proposed in LSS [42]; they explicitly predict a depth map and lift multi-view image features onto the BEV plane [17, 23, 24, 41]. Another line of works inherits the spirit of querying from 3D to 2D in DETR3D [55]; they employ learnable queries to extract information from image features by cross-attention mechanism [19, 26, 37, 56, 61]. While these methods efficiently compress information onto the BEV plane, they sacrifice some of the integral scene structure inherent in 3D space. To address this limitation, our proposed unified occupancy representation is better suited for achieving a holistic 3D understanding, making it ideal for tasks such as 3D semantic segmentation and panoptic segmentation.

**3D Occupancy Prediction.** Occupancy prediction can be traced back to Occupancy Grid Mapping (OGM) [48], a classic task in mobile robot navigation that aims to generate probabilistic maps from sequential noisy range measurements. Recently, there has been considerable attention given to camera-based 3D occupancy prediction, which aims to reconstruct the 3D scene structure from images. Existing tasks in this area can be categorized into two lines based on the type of supervision: sparse prediction and dense prediction. Sparse prediction methods derive supervision from LiDAR points and are evaluated on LiDAR benchmarks. For instance, [18] proposes a tri-perspective view method for predicting 3D occupancy. Dense prediction methods are closely related to Semantic Scene Completion (SSC)[1, 9, 29, 46]. MonoScene[4] first employs U-Net to infer dense 3D occupancy with semantic labels from a single monocular RGB image. VoxFormer [25] utilizes depth estimation to select voxel queries in a two-stage framework. Subsequently, a series of studies have focused on the task of dense occupancy prediction and have introduced new benchmarks. OpenOccupancy [54] offers a carefully annotated occupancy benchmark, while Occ3D [49] proposes an occupancy prediction benchmark using the Waymo and nuScenes datasets. Openocc [50] further provides occupancy flow annotation for dynamic objects modeling on the nuScenes dataset. Our proposed method unifies object detection and semantic segmentation prediction for the first time, applicable under both sparse LiDAR and dense occupancy supervision.

**LiDAR Panoptic Segmentation.** LiDAR panoptic segmentation [40] offers a comprehensive understanding of the environment by unifying semantic segmentation and object detection. However, traditional object detection methods often lose height information, making it challenging to learn fine-grained feature representations for accurate 3D segmentation. Recent LiDAR panoptic methods [15, 44, 68] have been developed based on well-designed semantic segmentation networks [7, 64] to address this limitation. Instead of predicting sparse semantic segmentation on LiDAR points, our proposed camera-based 3D panoptic segmenta-

tion aims to produce dense voxel segmentation of the scene.

### 3. Method

#### 3.1. Problem Setup

**Camera-based 3D panoptic segmentation.** Given multi-view images as input, camera-based 3D panoptic segmentation aims to predict a dense panoptic voxel volume surrounding the ego-vehicle. Specifically, we take current multi-view images denoted as  $\mathbf{I}_t = \{\mathbf{I}_t^1, \mathbf{I}_t^2, \dots, \mathbf{I}_t^n\}$  and previous frames  $\mathbf{I}_{t-1}, \dots, \mathbf{I}_{t-k}$  as input.  $n$  denotes the camera view index, while  $k$  denotes the number of history frames. The model outputs the current frame semantic voxel volume  $\mathbf{Y}_t \in \{w_0, w_1, \dots, w_C\}^{H \times W \times Z}$  and its corresponding instance ID  $\mathbf{N}_t \in \{v_0, v_1, v_2, \dots, v_P\}^{H \times W \times Z}$ . Here,  $C$  denotes the total number of semantic classes in the scene, while  $w_0$  represents the empty voxel grid.  $P$  is the total number of instances in the current frame  $t$ ; for each grid belonging to the foreground classes (*thing*), it would assign a specific instance ID  $v_j$ .  $v_0$  is assigned to all voxel grids belonging to the *stuff* categories and empty.  $H, W$ , and  $Z$  denote the length, width, and height of the surrounding voxel volume.

**Camera-based 3D semantic occupancy prediction.** This can be viewed as a sub-task within camera-based 3D panoptic segmentation, specifically targeting the prediction of the semantic voxel volume  $\mathbf{Y}_t \in w_0, w_1, \dots, w_C^{H \times W \times Z}$ . The emphasis is placed on accurately distinguishing the empty class ( $w_0$ ) from the other classes to determine whether a voxel grid is empty or occupied.

#### 3.2. Overall Architecture

In this section, we introduce the overall architecture of PanoOcc, which serves as a baseline for 3D panoptic segmentation. As illustrated in Figure 1, our approach takes multi-frame and multi-view images as input and outputs 3D panoptic segmentation for the current scene. Firstly, the image backbone extracts multi-scale features from the input images. These features are then processed by the *Occupancy Encoder*, which comprises the *View Encoder* and *Temporal Encoder*, to generate a coarse unified occupancy representation. Specifically, the *View Encoder* utilizes voxel queries to learn voxel features, preserving the actual 3D structure of the scene by explicitly encoding height information. The *Temporal Encoder* aligns and fuses previous voxel features with the current frame, capturing temporal information and enhancing the occupancy representation. The *Occupancy Decoder* employs a coarse-to-fine scheme to recover fine-grained feature representation. The *Coarse-to-fine Upsampling* module restores the high-resolution voxel representation, enabling efficient learning of precise occupancy representation. With the advantage of a unified occupancy representation, the model can jointly

learn object detection and semantic segmentation through the *Task Head*. Finally, the *Refine Module* refines the prediction of *thing* classes and outputs 3D panoptic segmentation results. Our model follows two key design principles: (1) **Unified occupancy representation** for learning and task output. (2) **Efficient feature learning** for 3D scenes. In the following, we provide detailed descriptions of designs in these two aspects.

#### 3.3. Unified Occupancy Representation

Occupancy serves as a unified 3D representation, not only reflected in the unity across different tasks (object detection and semantic segmentation) but also in the integration of feature learning processes. Therefore, we introduce our method from the perspectives of *Unified Learning* and *Unified Task* in the following.

**Unified Learning.** We adopt occupancy as feature representation in the learning process. To achieve this, we use *voxel queries* to aggregate multi-frame multi-view image features within *occupancy encoder*. Occupancy encoder consists of view encoder and temporal encoder. We define a group of 3D-grid-shape learnable parameters  $\mathbf{Q} \in \mathbb{R}^{H \times W \times Z \times D}$  as voxel queries.  $H$  and  $W$  are the spatial shape of the BEV plane, while  $Z$  represents the height dimension, and  $D$  is the embedding dimension. A single voxel query  $\mathbf{q} \in \mathbb{R}^D$  located at  $(i, j, k)$  position of  $\mathbf{Q}$  is responsible for the corresponding 3D voxel grid cell region. Each grid cell in the voxel corresponds to a real-world size of  $(s_h, s_w, s_z)$  meters. Given voxel queries  $\mathbf{Q}$  and extracted image features  $\mathbf{F}$  as input, the occupancy encoder outputs the fused voxel features  $\mathbf{Q}_f \in \mathbb{R}^{H \times W \times Z \times D}$ .

Compared to previous feature transformations based on BEV queries [26], the primary difference lies in the utilization of *attention operations* [69] and *temporal alignments*. In the *view encoder*, we incorporate attention operations into voxel space by designing voxel self-attention and voxel cross-attention. The core difference in lifting BEV queries to voxel queries computation lies in the selection of *reference points*; further details are provided in the appendix. The *temporal encoder* comprises two specific operations: *temporal alignment* and *temporal fusion*. Unlike previous temporal alignment methods [26, 41], which align historical features on the BEV plane, our approach utilizes voxel alignment in 3D space. This allows us to rectify inaccuracies caused by assumptions made in previous BEV-based methods, where road height remains unchanged throughout the scene—an assumption not always valid in real-world driving scenarios, especially in cases involving uphill and downhill terrain. Voxel alignment is crucial for generating fine-grained voxel representations to accurately perceive the environment. Specifically, the process of voxel alignment is formulated as follows:

$$\mathbf{Q}_{t-k \rightarrow t} = \text{GridSample}(\mathbf{Q}_{t-k}, \mathbf{G}_{t-k}) \quad (1)$$

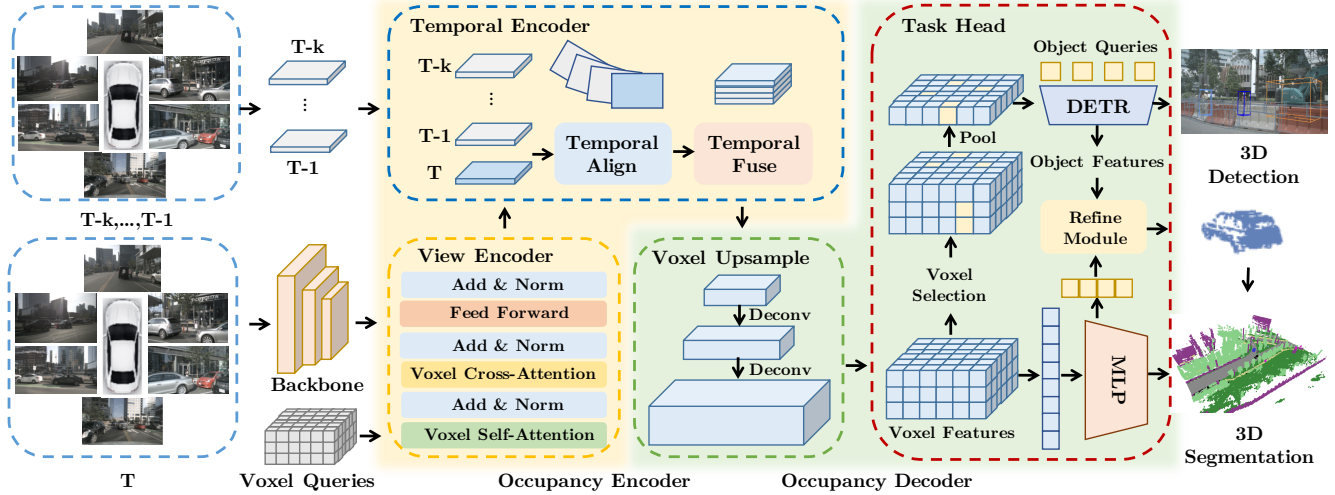


Figure 1. **The overall framework of PanoOcc.** Our framework begins by employing an image backbone network to extract multi-scale features from multi-view images across multiple frames. Subsequently, voxel queries are utilized to learn voxel features through the *View Encoder*. The *Temporal Encoder* then aligns the previous voxel features with the current frame and combines these features. The *Voxel Upsample* module restores the high-resolution voxel representation for fine-grained semantic classification. The *Task Head* predicts object detection and semantic segmentation through two separate heads. The *Refine Module* further refines the object class prediction with the assistance of 3D object detection and assigns instance IDs to generate 3D panoptic segmentation results.

$$\mathbf{G}_{t-k} = \mathbf{T}_{t \rightarrow t-k} \cdot \mathbf{G}_t \quad (2)$$

where  $\mathbf{G}_t \in \mathbb{R}^{H \times W \times Z}$  is the voxel grid at current frame  $t$ ,  $\mathbf{G}_{t-k} \in \mathbb{R}^{H \times W \times Z}$  represents the current frame grid at frame  $t - k$ .  $\mathbf{T}_{t \rightarrow t-k}$  is the transformation matrix for transforming the points at frame  $t$  to previous frame  $t - k$ . Then the queries at frame  $t - k$  are aligned to current frame  $t$  by interpolation sampling, denoted as  $\mathbf{Q}_{t-k \rightarrow t}$ . After alignment, the previous aligned voxel queries  $[\mathbf{Q}_{t-k \rightarrow t}, \dots, \mathbf{Q}_{t-1 \rightarrow t}]$  are concatenated with the current voxel queries  $\mathbf{Q}_t$ . We employ a block of residual 3D convolutions to fuse the queries and output fused voxel queries  $\mathbf{Q}_f$ .

**Unified Task.** With the advantage of occupancy representation, the model has a strong capacity to handle different tasks. We can unify the 3D object detection and semantic segmentation into 3D panoptic segmentation, achieving a more comprehensive understanding of the scene and a finer-grained modeling of objects. This allows us to *train jointly* and benefit from each other through the *foreground information propagation*.

Specifically, our model is trained end-to-end for joint detection and segmentation, while previous methods usually train separately due to conflicting learning objectives. To address this problem, we leverage foreground information propagation between the semantic head and the detection head. The total loss  $\mathcal{L}$  consists of two parts:  $\mathcal{L}_{Det}$  and  $\mathcal{L}_{Seg}$ . The semantic voxel segmentation head is supervised by  $\mathcal{L}_{Seg}$ , a dense loss consisting of focal loss [31] (applied to all voxels) and Lovasz loss [2] (applied to non-empty

voxels). We utilize *voxel selection* to convey foreground information to the detection head, which predicts a binary voxel mask to select the voxel features corresponding to foreground categories (*thing*). The voxel mask is also supervised by focal loss [31] denoted as  $\mathcal{L}_{thing}$ . The total loss  $\mathcal{L}_{Seg}$  is formulated as:

$$\mathcal{L}_{Seg} = \lambda_1 \mathcal{L}_{focal} + \lambda_2 \mathcal{L}_{lovasz} + \lambda_3 \mathcal{L}_{thing} \quad (3)$$

The detection head is supervised by  $\mathcal{L}_{Det}$ , a sparse loss consisting of focal loss [31] for classification and L1 loss for bounding box regression:

$$\mathcal{L}_{Det} = \lambda_4 \mathcal{L}_{cls} + \lambda_5 \mathcal{L}_{reg} \quad (4)$$

The *Refine module* further enhances the predicted foreground (*thing*) voxels using the detection results and generates 3D panoptic segmentation results. We begin by sorting all box predictions based on their confidence scores. Subsequently, we select a set of high-confidence bounding boxes denoted as  $G = \{b_i | s_i > \tau\}$ , where  $b_i$  represents a 3D bounding box,  $s_i$  is the confidence score, and  $\tau$  is a threshold (default:  $\tau = 0.8$ ). For the voxels within each bounding box  $b_i$ , we assign the class prediction  $c_i$  to all of them. To perform panoptic voxel segmentation, we sequentially assign instance IDs based on confidence scores. If the current instance overlaps with previous instances beyond a certain threshold, we ignore it to avoid duplication. Finally, we assign instance ID 0 to all voxels corresponding to the *stuff* class.



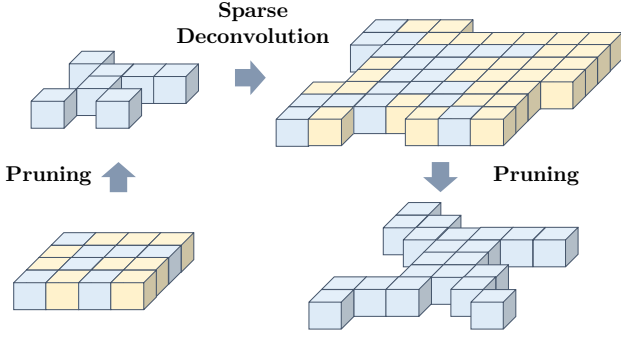


Figure 2. **Illustration of occupancy sparsify.** It serves as an optional technique to boost efficiency. We use BEV representation for simple illustration, while it is actually a 3D process. The light yellow region will be pruned according to occupancy masks.

### 3.4. Efficient Feature Learning

Compared to the information density in image space, 3D space exhibits significantly greater sparsity. Additionally, directly extending Bird’s Eye View (BEV) features to voxel features would result in substantial memory and computational costs. Therefore, within the *occupancy decoder*, we introduce two designs: *Coarse-to-fine Upsampling* and *Occupancy Sparsify*, aimed at alleviating this issue.

**Coarse-to-fine Upsampling.** This design enables the model to only learn a coarse voxel feature  $\mathbf{Q}_f$  in the occupancy encoder. The module then utilizes 3D deconvolutions to upsample the fused voxel query  $\mathbf{Q}_f \in \mathbb{R}^{H \times W \times Z \times D}$  to high-resolution occupancy features  $\mathbf{O} \in \mathbb{R}^{H' \times W' \times Z' \times D'}$ . Such a coarse-to-fine manner not only avoids directly applying expensive 3D convolutions to high-resolution occupancy features, but also leads to no performance loss. We have a quantitative discussion in the Table 5.

**Occupancy Sparsify.** Although the coarse-to-fine manner guarantees the high efficiency of our method, there is a considerable computational waste on the spatially dense feature  $\mathbf{Q}_f$  and  $\mathbf{O}$ . This is because our physical world is essentially sparse in spatial dimensions, which means a large portion of space is not occupied. Dense operations (i.e., dense convolution) violate such essential sparsity. Inspired by the success of sparse architecture in LiDAR-based perception [11, 33, 60], we optionally turn to the Sparse Convolution [13] for occupancy sparsify. In particular, we first learn an occupancy mask for  $\mathbf{Q}_f$  to indicate if positions on  $\mathbf{Q}_f$  are occupied. Then we prune  $\mathbf{Q}_f$  to a sparse feature  $\mathbf{Q}_{sparse} \in \mathbb{R}^{N \times D}$  by discarding those empty positions according to the learned occupancy mask, where  $N \ll HWZ$  and  $N$  is determined by a predefined keeping ratio  $R_{keep}$ . After the pruning, all the following dense convolutions are replaced by corresponding sparse convolutions. Since sparse deconvolution will dilate the sparse features to empty positions and reduce the sparsity, we conduct similar pruning operations after each upsampling to main-

tain the spatial sparsity. Finally, we obtain a high-resolution and sparse occupancy feature  $\mathbf{O}_{sparse} \in \mathbb{R}^{N' \times D'}$ , where  $N' \ll H'W'Z'$ . Figure 2 illustrates the occupancy sparsify process.

## 4. Experiments

### 4.1. Datasets

**nuScenes dataset** [3] contains 1000 scenes in total, split into 700 in the training set, 150 in the validation set, and 150 in the test set. Each sequence is captured at 20Hz frequency with 20 seconds duration. Each sample contains RGB images from 6 cameras with 360° horizontal FOV and point cloud data from 32 beam LiDAR sensor. For the task of object detection, the key samples are annotated at 2Hz with ground truth labels for 10 foreground object classes (*thing*). For the task of semantic segmentation and panoptic segmentation, every LiDAR point in the key samples is annotated using 6 more background classes (*stuff*) in addition to the 10 foreground classes (*thing*).

**Occ3D-nuScenes** [49] contains 700 training scenes and 150 validation scenes. The occupancy scope is defined as  $-40m$  to  $40m$  for X and Y-axis, and  $-1m$  to  $5.4m$  for the Z-axis in the ego coordinate. The voxel size is  $0.4m \times 0.4m \times 0.4m$  for the occupancy label. The semantic labels contain 17 categories (including ‘others’). Besides, it also provides visibility masks for LiDAR and camera modality, indicating which regions are visible from the sensor.

**Evaluation metrics.** nuScenes dataset uses mean Average Precision (mAP) and nuScenes Detection Score (NDS) metrics for the detection task, mean Intersection over Union (mIoU), and Panoptic Quality (PQ) metrics [20] for the 3D semantic and panoptic segmentation.  $PQ^\dagger$  is a modified panoptic quality [43], which maintains the PQ metric for *thing* classes, but modifies the metric for *stuff* classes. The Occ3D-nuScenes benchmark [49] calculates the mean Intersection over Union (mIoU) for 17 semantic categories within the camera’s visible region.

### 4.2. Experimental Settings

**Implementation Details.** For the implementation details of the model, please refer to the appendix A. On the nuScenes dataset [3], we set the point cloud range for the  $x$  and  $y$  axis to  $[-51.2m, 51.2m]$ , and  $[-5m, 3m]$  for the  $z$  axis. The voxel grid size used for loss supervision is  $(0.256m, 0.256m, 0.125m)$ . For the training and inference details, please refer to the appendix D. The input image size is cropped to  $640 \times 1600$ . When using the R101-DCN [10] or InternImage [53] as the backbone, we default to the 1.0 image scale ( $640 \times 1600$ ). However, when using the R50 [14] backbone, we adopt a 0.5 image scale ( $320 \times 800$ ).

**Evaluation.** For sparse evaluation on the LiDAR benchmark, our approach assesses LiDAR semantic segmentation

Method	Input Modality	Image Backbone	mIoU	barrier	bicycle	bus	car	const. veh.	motorcycle	pedestrian	traffic cone	trailer	truck	drive. suf.	other flat	sidewalk	terrain	manmade	vegetation
				■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■
RangeNet++ [39]	LiDAR	-	65.5	66.0	21.3	77.2	80.9	30.2	66.8	69.6	52.1	54.2	72.3	94.1	66.6	63.5	70.1	83.1	79.8
PolarNet [64]	LiDAR	-	71.0	74.7	28.2	85.3	90.9	35.1	77.5	71.3	58.8	57.4	76.1	96.5	71.1	74.7	74.0	87.3	85.7
Salsanext [8]	LiDAR	-	72.2	74.8	34.1	85.9	88.4	42.2	72.4	72.2	63.1	61.3	76.5	96.0	70.8	71.2	71.5	86.7	84.4
Cylinder3D++ [70]	LiDAR	-	76.1	76.4	40.3	91.2	93.8	51.3	78.0	78.9	64.9	62.1	84.4	96.8	71.6	76.4	75.4	90.5	87.4
RPVNet [58]	LiDAR	-	77.6	78.2	43.4	92.7	93.2	49.0	85.7	80.5	66.0	66.9	84.0	96.9	73.5	75.9	76.0	90.6	88.9
TPVFormer [18]	Camera	R50	59.3	64.9	27.0	83.0	82.8	38.3	27.4	44.9	24.0	55.4	73.6	91.7	60.7	59.8	61.1	78.2	<b>76.5</b>
PanoOcc	Camera	R50	<b>68.1</b>	<b>70.7</b>	<b>37.9</b>	<b>92.3</b>	<b>85.0</b>	<b>50.7</b>	<b>64.3</b>	<b>59.4</b>	<b>35.3</b>	<b>63.8</b>	<b>81.6</b>	<b>94.2</b>	<b>66.4</b>	<b>64.8</b>	<b>68.0</b>	<b>79.1</b>	75.6
BEVFormer [26]	Camera	R101-DCN	56.2	54.0	22.8	76.7	74.0	45.8	53.1	44.5	24.7	54.7	65.5	88.5	58.1	50.5	52.8	71.0	63.0
TPVFormer [18]	Camera	R101-DCN	68.9	70.0	40.9	93.7	85.6	49.8	<b>68.4</b>	59.7	38.2	65.3	83.0	93.3	64.4	64.3	64.5	81.6	79.3
OccFormer [65]	Camera	R101-DCN	70.4	70.3	<b>43.8</b>	93.2	85.2	52.0	59.1	<b>67.6</b>	<b>45.4</b>	64.4	84.5	93.8	68.2	<b>67.8</b>	<b>68.3</b>	<b>82.1</b>	<b>80.4</b>
PanoOcc	Camera	R101-DCN	<b>71.6</b>	<b>74.3</b>	43.7	<b>95.4</b>	<b>87.0</b>	<b>56.1</b>	64.6	66.2	41.4	<b>71.5</b>	<b>85.9</b>	<b>95.1</b>	<b>70.1</b>	67.0	68.1	80.9	77.4
PanoOcc	Camera	Intern-XL	<b>74.5</b>	<b>75.3</b>	<b>51.1</b>	<b>96.9</b>	<b>87.5</b>	<b>56.6</b>	<b>85.6</b>	<b>68.0</b>	<b>43.0</b>	<b>74.1</b>	<b>87.1</b>	<b>95.1</b>	<b>71.0</b>	<b>68.7</b>	<b>70.3</b>	<b>82.3</b>	<b>79.3</b>

Table 1. **LiDAR semantic segmentation results on nuScenes validation set.** Our method achieves comparable performance with state-of-the-art LiDAR-based methods and notably surpasses the recently proposed camera-based methods.

by assigning voxel semantic predictions to LiDAR points. We extend this evaluation with object detection results, enabling panoptic evaluation on LiDAR panoptic segmentation [12]. While PQ only considers sparse points and may not comprehensively reflect the understanding of foreground objects, we still use mAP, NDS, and mIoU to measure the effectiveness of our approach in experiments. For dense evaluation on the occupancy benchmark [49], we directly compute mIoU based on the occupancy labels.

### 4.3. Main Results

We validate the performance of our methods on three benchmarks: 3D semantic segmentation, 3D panoptic segmentation, and 3D occupancy prediction on the nuScenes dataset. The results showcase that our PanoOcc achieves state-of-the-art performance across all benchmarks. Notably, we are also the first to implement an end-to-end method for camera-based panoptic segmentation.

**3D Semantic Segmentation.** We evaluate the model performance on the nuScenes test and validation set respectively. In Table 1, we adopt three types of backbone to conduct experiments. Under the R50 [14] and R101-DCN [10] setting, our method achieves 68.1 mIoU and 71.6 mIoU, a new state-of-the-art. To further validate our approach, we experiment with a larger image backbone [53] and achieve an impressive 74.5 mIoU, approaching the performance of current state-of-the-art LiDAR-based methods. The test set performance is provided in the appendix B.

**3D Occupancy Prediction.** In Table 2, we evaluate our method for 3D occupancy prediction on the Occ3D-nuScenes [49] validation set. All methods utilize camera input and are trained for 24 epochs. The performance of MonoScene [4], BEVDet [17], BEVFormer [26], and CTF-

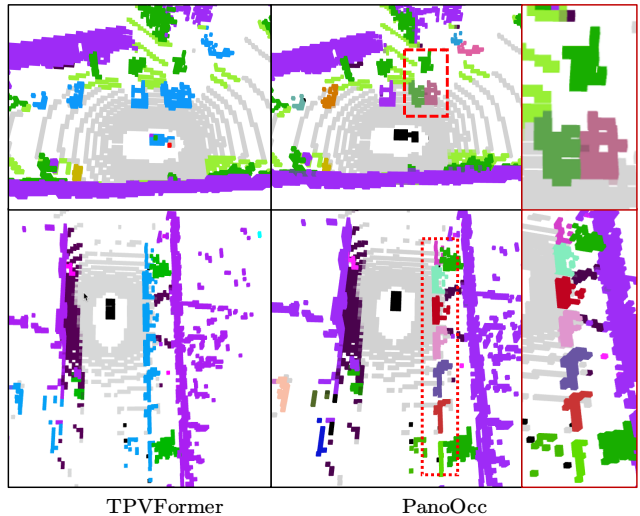


Figure 3. **Visualizations of camera-based panoptic segmentation.** Here, we present a comparison between two samples processed by TPVFormer and our method. Our approach enables the output of panoptic segmentation, particularly highlighting fine-grained instance discrimination (highlighted in the red box).

Occ [49] is reported in the work of [49]. The use of the camera visible mask during training has proven to be an effective technique. We re-implemented BEVFormer [26] with the inclusion of the camera mask during training. Our PanoOcc also use camera visible mask during training and achieves a new state-of-art performance. We adopt the R101-DCN as the backbone and use 4 frames for temporal fusion. Figure 4 illustrates the dense occupancy prediction on the Occ3D-nuScenes validation set.

**3D Panoptic Segmentation.** PanoOcc is the first work to implement an end-to-end trainable model for camera-based

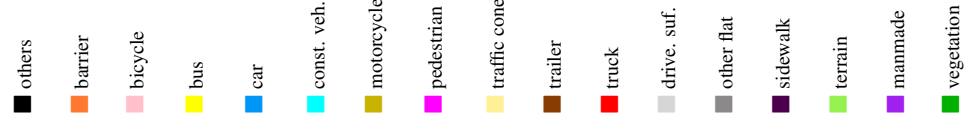
Method	Image Backbone	mIoU																	
			others	barrier	bicycle	bus	car	const. veh.	motorcycle	pedestrian	traffic cone	trailer	truck	drive. suf.	other flat	sidewalk	terrain	manmade	vegetation
MonoScene [4]	R101-DCN	6.06	1.75	7.23	4.26	4.93	9.38	5.67	3.98	3.01	5.90	4.45	7.17	14.91	6.32	7.92	7.43	1.01	7.65
BEVDet [17]	R101-DCN	11.73	2.09	15.29	0.0	4.18	12.97	1.35	0.0	0.43	0.13	6.59	6.66	52.72	19.04	26.45	21.78	14.51	15.26
BEVFormer [26]	R101-DCN	26.88	5.85	37.83	17.87	40.44	42.43	7.36	23.88	21.81	20.98	22.38	30.70	55.35	28.36	36.0	28.06	20.04	17.69
TPVFormer [18]	R101-DCN	27.83	7.22	38.90	13.67	40.78	45.90	17.23	19.99	18.85	14.30	<b>26.69</b>	34.17	55.65	35.47	37.55	30.70	19.40	16.78
CTF-Occ [49]	R101-DCN	28.53	8.09	39.33	20.56	38.29	42.24	16.93	24.52	22.72	21.05	22.98	31.11	53.33	33.84	37.98	<b>33.23</b>	<b>20.79</b>	<b>18.0</b>
PanoOcc	R101-DCN	<b>32.47</b>	<b>10.85</b>	<b>46.93</b>	<b>27.25</b>	<b>43.54</b>	<b>48.74</b>	<b>23.02</b>	<b>31.16</b>	<b>27.59</b>	<b>28.59</b>	26.58	<b>38.27</b>	<b>58.05</b>	<b>38.94</b>	<b>38.15</b>	32.27	15.58	16.41
BEVFormer* [26]	R101-DCN	39.24	10.13	47.91	24.9	47.57	54.52	20.23	28.85	28.02	25.73	33.03	38.56	81.98	40.65	50.93	53.02	43.86	37.15
PanoOcc*	R101-DCN	<b>42.13</b>	<b>11.67</b>	<b>50.48</b>	<b>29.64</b>	<b>49.44</b>	<b>55.52</b>	<b>23.29</b>	<b>33.26</b>	<b>30.55</b>	<b>30.99</b>	<b>34.43</b>	<b>42.57</b>	<b>83.31</b>	<b>44.23</b>	<b>54.40</b>	<b>56.04</b>	<b>45.94</b>	<b>40.40</b>

Table 2. **3D Occupancy prediction performance on the Occ3D-nuScenes dataset.** \* means the performance is achieved by using the camera mask during training. The results indicate that our method achieves state-of-the-art performance in both settings.

Method	Input Modality	PQ	PQ <sup>†</sup>	RQ	SQ	mAP
EfficientLPS [45]	LiDAR	62.0	65.6	73.9	83.4	/
Panoptic-PolarNet [68]	LiDAR	63.4	67.2	75.3	83.9	/
Panoptic-PHNet [21]	LiDAR	74.7	77.7	84.2	88.2	/
LidarMultinet [62]	LiDAR	81.8	/	90.8	89.7	63.8
PanoOcc	Camera	62.1	66.2	75.1	82.1	48.4

Table 3. **LiDAR panoptic segmentation results on nuScenes validation set.** Our PanoOcc based on the camera input has approached LiDAR-based methods’ performance.

panoptic segmentation. We compare our method with previous LiDAR-based panoptic segmentation methods. Table 3 demonstrates that our PanoOcc achieves a PQ of 62.1, showing comparable performance to some LiDAR-based methods like EfficientLPS [45] and PolarNet [64]. However, our approach still exhibits a performance gap compared to state-of-the-art LiDAR-based methods, which can be attributed to inferior detection performance (48.4 mAP vs. 63.8 mAP). As illustrated in Figure 3, we provide a qualitative comparison between TPVFormer [18] and ours. Our method enables panoptic segmentation, particularly emphasizing fine-grained instance discrimination.

#### 4.4. Ablation Study

In this section, we mainly validate the key design choices of PanoOcc on the nuScenes validation set. Please refer to the appendix C for more ablation studies. The ablation studies are conducted for the 3D panoptic segmentation task, evaluated on the LiDAR benchmark of the nuScenes dataset.

**Joint Learning of Detection and Segmentation.** Table 4 demonstrates the significant positive impact of training for joint detection and segmentation. When compared to single-task models, the jointly-trained model excels in both the segmentation and detection tasks. Voxel selection further enhances the interaction between detection and seg-

	Det.	Seg.	Vox. Sel.	mIoU	mAP	NDS
(a)	✓			/	0.252	0.310
(b)		✓		0.652	/	/
(c)	✓	✓		0.656	0.266	0.319
(d)	✓	✓	✓	<b>0.661</b>	<b>0.271</b>	<b>0.324</b>

Table 4. **Effectiveness of joint detection and segmentation.** Det. stands for detection head. Seg. denotes segmentation head. Vox. Sel. represents voxel selection for foreground voxels.

Voxel Resolution	Voxel Upsampling	Memory	Latency	Param	FPS	mIoU
200x200x8		37G / 9.5G	255 ms	117.7 M	4.1	67.9
50x50x16	✓	<b>18G / 5.7G</b>	<b>149 ms</b>	<b>48.7 M</b>	<b>9.2</b>	<b>68.3</b>

Table 5. **Ablation study for the coarse-to-fine design.** We show the train / inference memory consumption, respectively. The experiments were conducted on the A100 GPU.

mentation learning, improving performance in both tasks. **Efficiency of Coarse-to-Fine Design.** Table 5 illustrates the effectiveness of our coarse-to-fine scheme. By comparing it with the direct use of high-resolution voxel queries (200×200×8), we observe that our coarse-to-fine design achieves comparable or even superior performance while consuming nearly half the memory (33.5G v.s. 24G). Our approach, for the first time, reveals that high-resolution semantic occupancy could be effectively learned in the low-resolution latent space via a coarse-to-fine scheme.

#### 4.5. Discussion

In this section, we delve into the benefits of voxel queries and explore the potential of sparse design in the future. All experiments are evaluated for 3D semantic segmentation.

**Voxel v.s. Tri-plane.** Traditionally, it has been widely believed that using 3D voxel grids alone is an inefficient solution due to the memory cost. This belief has led TPVFormer [18] to split the 3D representation into three 2D planes. However, we have demonstrated that employing the



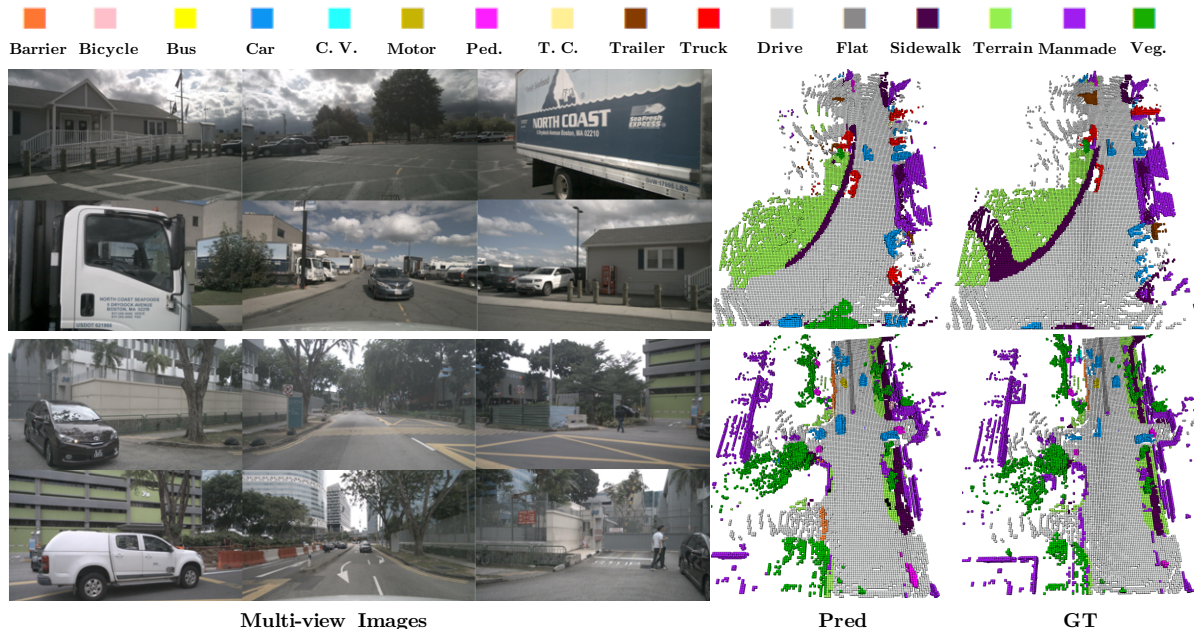


Figure 4. **Qualitative results on Occ3D-nuScenes validation set.** Our PanoOcc takes multi-view images as input and produces dense occupancy predictions, which are visualized at the resolution of  $200 \times 200 \times 16$ .

Method	Query form	Resolution	Memory↓	Latency↓	FPS↑	mIoU↑
TPVFormer*	2D Tri-plane	200x(200+16+16)	33.5G / 7.1G	268 ms	3.7	68.9
PanoOcc	3D Voxel	50x50x16	<b>24G / 6.0G</b>	<b>203 ms</b>	<b>4.8</b>	<b>71.6</b>

Table 6. **Model efficiency comparison with different query forms.** \* denotes performance obtained using the official code and released checkpoints. We report the memory consumption during training and inference in the experiments. The experiments are evaluated for the 3D semantic segmentation task on the nuScenes benchmark.

	Convolution	Latency↓	Memory↓	FPS↑	mIoU↑
(a)	Dense	126 ms	15 G	9.3	<b>0.654</b>
(b)	Sparse	<b>112 ms</b>	<b>9 G</b>	<b>9.7</b>	0.639

Table 7. **Exploration of sparse architecture design.** The experiment is conducted under the R50 setting without temporal fusion.

coarse-to-fine learning scheme can effectively address the memory increase issue. In Table 6, we compare the performance and efficiency of our method with the previous state-of-the-art approach [18], under the same experimental setup. Despite having an additional detection branch, our model still exhibits lower memory consumption and faster inference speed.

**Occupancy Sparsify.** In contrast to 2D space, 3D space exhibits high sparsity, indicating that the majority of voxels are empty. In Table 7, we investigate the effectiveness of the occupancy sparsify strategy. Here we have 3 layers of sparse deconvolution for upsampling in total. In coarse-to-fine order, the keeping ratio after each upsampling is 0.2, 0.5, and 0.5, respectively. It suggests that finally we only keep 5% voxels, and this reduction has not resulted in a significant performance decrease.

## 5. Conclusion

In this paper, we propose *camera-based 3D panoptic segmentation*, aiming for a comprehensive understanding of the scene by a unified occupancy representation. To facilitate learning of occupancy representation, we propose a novel framework called PanoOcc, which leverages voxel queries to integrate information from multi-frame and multi-view images in a coarse-to-fine manner. Extensive experiments conducted on the nuScenes dataset and Occ3D-nuScenes demonstrate the effectiveness of PanoOcc and its potential to advance holistic 3D scene understanding. We envision 3D occupancy representation as a promising new paradigm for future 3D scene perception.

## Acknowledgments

This work was supported in part by the National Key R&D Program of China (No. 2022ZD0160102), the National Natural Science Foundation of China (No. U21B2042, No. 62072457), the innoHK funding, and in part by the 2035 Innovation Program of CAS.



## References

- [1] Iro Armeni, Sasha Sax, Amir R Zamir, and Silvio Savarese. Joint 2d-3d-semantic data for indoor scene understanding. *arXiv preprint arXiv:1702.01105*, 2017. 2
- [2] Maxim Berman, Amal Rannen Triki, and Matthew B Blaschko. The lovasz-softmax loss: A tractable surrogate for the optimization of the intersection-over-union measure in neural networks. In *CVPR*, 2018. 4
- [3] Holger Caesar, Varun Bankiti, Alex H Lang, Sourabh Vora, Venice Erin Liong, Qiang Xu, Anush Krishnan, Yu Pan, Giancarlo Baldan, and Oscar Beijbom. nuscenes: A multi-modal dataset for autonomous driving. In *CVPR*, 2020. 5
- [4] Anh-Quan Cao and Raoul de Charette. Monoscene: Monocular 3d semantic scene completion. In *CVPR*, 2022. 1, 2, 6, 7
- [5] Pengguang Chen, Shu Liu, Hengshuang Zhao, and Jiaya Jia. Gridmask data augmentation. *arXiv preprint arXiv:2001.04086*, 2020. 3
- [6] Qi Chen, Sourabh Vora, and Oscar Beijbom. Polarstream: Streaming object detection and segmentation with polar pillars. *NeurIPS*, 2021. 2
- [7] Ran Cheng, Ryan Razani, Ehsan Taghavi, Enxu Li, and Bingbing Liu. 2-s3net: Attentive feature fusion with adaptive feature selection for sparse semantic segmentation network. In *CVPR*, 2021. 2
- [8] Tiago Cortinhal, George Tzelepis, and Eren Erdal Aksoy. Salsanext: Fast, uncertainty-aware semantic segmentation of lidar point clouds. In *ISVC*, 2020. 6
- [9] Angela Dai, Angel X Chang, Manolis Savva, Maciej Halber, Thomas Funkhouser, and Matthias Nießner. Scannet: Richly-annotated 3d reconstructions of indoor scenes. In *CVPR*, 2017. 2
- [10] Jifeng Dai, Haozhi Qi, Yuwen Xiong, Yi Li, Guodong Zhang, Han Hu, and Yichen Wei. Deformable convolutional networks. In *ICCV*, 2017. 5, 6, 1, 3
- [11] Lue Fan, Yuxue Yang, Feng Wang, Naiyan Wang, and Zhaoxiang Zhang. Super sparse 3d object detection. *TPAMI*, 2023. 5
- [12] Whye Kit Fong, Rohit Mohan, Juana Valeria Hurtado, Lubing Zhou, Holger Caesar, Oscar Beijbom, and Abhinav Valada. Panoptic nusenes: A large-scale benchmark for lidar panoptic segmentation and tracking. *RAL*, 2022. 6
- [13] Benjamin Graham and Laurens van der Maaten. Submanifold Sparse Convolutional Networks. *arXiv preprint arXiv:1706.01307*, 2017. 5
- [14] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016. 5, 6, 1, 3
- [15] Fangzhou Hong, Hui Zhou, Xinge Zhu, Hongsheng Li, and Ziwei Liu. Lidar-based panoptic segmentation via dynamic shifting network. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13090–13099, 2021. 2
- [16] Anthony Hu, Zak Murez, Nikhil Mohan, Sofia Dudas, Jeffrey Hawke, Vijay Badrinarayanan, Roberto Cipolla, and Alex Kendall. Fiery: future instance prediction in bird’s-eye view from surround monocular cameras. In *ICCV*, 2021. 1
- [17] Junjie Huang, Guan Huang, Zheng Zhu, and Dalong Du. Bevdet: High-performance multi-camera 3d object detection in bird-eye-view. *arXiv preprint arXiv:2112.11790*, 2021. 2, 6, 7
- [18] Yuanhui Huang, Wenzhao Zheng, Yunpeng Zhang, Jie Zhou, and Jiwen Lu. Tri-perspective view for vision-based 3d semantic occupancy prediction. In *CVPR*, 2023. 1, 2, 6, 7, 8
- [19] Yanqin Jiang, Li Zhang, Zhenwei Miao, Xiatian Zhu, Jin Gao, Weiming Hu, and Yu-Gang Jiang. Polarformer: Multi-camera 3d object detection with polar transformer. In *AAAI*, 2023. 2
- [20] Alexander Kirillov, Kaiming He, Ross Girshick, Carsten Rother, and Piotr Dollár. Panoptic segmentation. In *CVPR*, 2019. 5
- [21] Jinke Li, Xiao He, Yang Wen, Yuan Gao, Xiaoqiang Cheng, and Dan Zhang. Panoptic-phnet: Towards real-time and high-precision lidar panoptic segmentation via clustering pseudo heatmap. In *CVPR*, 2022. 7
- [22] Shijie Li, Xieyuanli Chen, Yun Liu, Dengxin Dai, Cyrill Stachniss, and Juergen Gall. Multi-scale interaction for real-time lidar data segmentation on an embedded platform. *RAL*, 2021. 2
- [23] Yinhao Li, Han Bao, Zheng Ge, Jinrong Yang, Jianjian Sun, and Zeming Li. Bevestereo: Enhancing depth estimation in multi-view 3d object detection with temporal stereo. In *AAAI*, 2023. 2
- [24] Yinhao Li, Zheng Ge, Guanyi Yu, Jinrong Yang, Zengran Wang, Yukang Shi, Jianjian Sun, and Zeming Li. Bevdepth: Acquisition of reliable depth for multi-view 3d object detection. In *AAAI*, 2023. 1, 2
- [25] Yiming Li, Zhiding Yu, Christopher Choy, Chaowei Xiao, Jose M Alvarez, Sanja Fidler, Chen Feng, and Anima Anandkumar. Voxformer: Sparse voxel transformer for camera-based 3d semantic scene completion. In *CVPR*, 2023. 1, 2
- [26] Zhiqi Li, Wenhao Wang, Hongyang Li, Enze Xie, Chonghao Sima, Tong Lu, Yu Qiao, and Jifeng Dai. Bevformer: Learning bird’s-eye-view representation from multi-camera images via spatiotemporal transformers. In *ECCV*, 2022. 1, 2, 3, 6, 7
- [27] Zhiqi Li, Zhiding Yu, David Austin, Mingsheng Fang, Shiyi Lan, Jan Kautz, and Jose M Alvarez. Fb-occ: 3d occupancy prediction based on forward-backward view transformation. *arXiv preprint arXiv:2307.01492*, 2023. 1
- [28] Bencheng Liao, Shaoyu Chen, Xinggang Wang, Tianheng Cheng, Qian Zhang, Wenyu Liu, and Chang Huang. Maptr: Structured modeling and learning for online vectorized hd map construction. In *ICLR*, 2022. 1
- [29] Yiyi Liao, Jun Xie, and Andreas Geiger. Kitti-360: A novel dataset and benchmarks for urban scene understanding in 2d and 3d. *TPAMI*, 2022. 2
- [30] Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. Feature pyramid networks for object detection. In *CVPR*, 2017. 1

- [31] Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection. In *Proceedings of the IEEE international conference on computer vision*, pages 2980–2988, 2017. 4
- [32] Venice Erin Liong, Thi Ngoc Tho Nguyen, Sergi Widjaja, Dhananjai Sharma, and Zhuang Jie Chong. Amvnet: Assertion-based multi-view fusion network for lidar semantic segmentation. *arXiv preprint arXiv:2012.04934*, 2020. 2
- [33] Jianhui Liu, Yukang Chen, Xiaoqing Ye, Zhuotao Tian, Xiao Tan, and Xiaojuan Qi. Spatial pruned sparse convolution for efficient 3d object detection. *NeurIPS*, 2022. 5
- [34] Yingfei Liu, Junjie Yan, Fan Jia, Shuailin Li, Aqi Gao, Tiancai Wang, and Xiangyu Zhang. Petrv2: A unified framework for 3d perception from multi-camera images. In *ICCV*, 2023. 1
- [35] Yicheng Liu, Tianyuan Yuan, Yue Wang, Yilun Wang, and Hang Zhao. Vectormapnet: End-to-end vectorized hd map learning. In *ICML*, 2023. 1
- [36] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *ICLR*, 2018. 3
- [37] Jiachen Lu, Zheyuan Zhou, Xiatian Zhu, Hang Xu, and Li Zhang. Learning ego 3d representation as ray tracing. In *ECCV*, 2022. 2
- [38] Ruihang Miao, Weizhou Liu, Mingrui Chen, Zheng Gong, Weixin Xu, Chen Hu, and Shuchang Zhou. Occdepth: A depth-aware method for 3d semantic scene completion. *arXiv preprint arXiv:2302.13540*, 2023. 1
- [39] Andres Milioto, Ignacio Vizzo, Jens Behley, and Cyrill Stachniss. Rangenet++: Fast and accurate lidar semantic segmentation. In *IROS*, 2019. 6
- [40] Andres Milioto, Jens Behley, Chris McCool, and Cyrill Stachniss. Lidar panoptic segmentation for autonomous driving. In *IROS*, 2020. 2
- [41] Jinhyung Park, Chenfeng Xu, Shijia Yang, Kurt Keutzer, Kris M Kitani, Masayoshi Tomizuka, and Wei Zhan. Time will tell: New outlooks and a baseline for temporal multi-view 3d object detection. In *ICLR*, 2022. 2, 3
- [42] Jonah Philion and Sanja Fidler. Lift, splat, shoot: Encoding images from arbitrary camera rigs by implicitly unprojecting to 3d. In *ECCV*, 2020. 1, 2
- [43] Lorenzo Porzi, Samuel Rota Buló, Aleksander Colovic, and Peter Kotschieder. Seamless scene segmentation. In *CVPR*, 2019. 5
- [44] Ryan Razani, Ran Cheng, Enxu Li, Ehsan Taghavi, Yuan Ren, and Liu Bingbing. Gp-s3net: Graph-based panoptic sparse semantic segmentation network. In *ICCV*, 2021. 2
- [45] Kshitij Sirohi, Rohit Mohan, Daniel Büscher, Wolfram Burgard, and Abhinav Valada. Efficientlps: Efficient lidar panoptic segmentation. *TRO*, 2021. 7
- [46] Shuran Song, Fisher Yu, Andy Zeng, Angel X Chang, Manolis Savva, and Thomas Funkhouser. Semantic scene completion from a single depth image. In *CVPR*, 2017. 2
- [47] 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 *ECCV*, 2020. 2
- [48] Sebastian Thrun. Probabilistic robotics. *Communications of the ACM*, 2002. 2
- [49] Xiaoyu Tian, Tao Jiang, Longfei Yun, Yucheng Mao, Huitong Yang, Yue Wang, Yilun Wang, and Hang Zhao. Occ3d: A large-scale 3d occupancy prediction benchmark for autonomous driving. *NeurIPS*, 2023. 2, 5, 6, 7
- [50] Wenwen Tong, Chonghao Sima, Tai Wang, Li Chen, Silei Wu, Hanming Deng, Yi Gu, Lewei Lu, Ping Luo, Dahua Lin, et al. Scene as occupancy. In *ICCV*, 2023. 2
- [51] Shihao Wang, Yingfei Liu, Tiancai Wang, Ying Li, and Xiangyu Zhang. Exploring object-centric temporal modeling for efficient multi-view 3d object detection. In *ICCV*, 2023. 1
- [52] Tai Wang, Xinge Zhu, Jiangmiao Pang, and Dahua Lin. Fcos3d: Fully convolutional one-stage monocular 3d object detection. In *ICCV*, 2021. 1
- [53] Wenhai Wang, Jifeng Dai, Zhe Chen, Zhenhang Huang, Zhiqi Li, Xizhou Zhu, Xiaowei Hu, Tong Lu, Lewei Lu, Hongsheng Li, et al. Internimage: Exploring large-scale vision foundation models with deformable convolutions. In *CVPR*, 2023. 5, 6, 1, 3
- [54] Xiaofeng Wang, Zheng Zhu, Wenbo Xu, Yunpeng Zhang, Yi Wei, Xu Chi, Yun Ye, Dalong Du, Jiwen Lu, and Xingang Wang. Openoccupancy: A large scale benchmark for surrounding semantic occupancy perception. In *ICCV*, 2023. 1, 2
- [55] Yue Wang, Vitor Campagnolo Guizilini, Tianyuan Zhang, Yilun Wang, Hang Zhao, and Justin Solomon. Detr3d: 3d object detection from multi-view images via 3d-to-2d queries. In *CoRL*, 2022. 1, 2
- [56] Yuqi Wang, Yuntao Chen, and Zhaoxiang Zhang. Frustumformer: Adaptive instance-aware resampling for multi-view 3d detection. In *CVPR*, 2023. 2
- [57] Yi Wei, Linqing Zhao, Wenzhao Zheng, Zheng Zhu, Jie Zhou, and Jiwen Lu. Surroundocc: Multi-camera 3d occupancy prediction for autonomous driving. In *ICCV*, 2023. 1
- [58] Jianyun Xu, Ruixiang Zhang, Jian Dou, Yushi Zhu, Jie Sun, and Shiliang Pu. Rpvnet: A deep and efficient range-point-voxel fusion network for lidar point cloud segmentation. In *ICCV*, 2021. 6
- [59] Xu Yan, Jiantao Gao, Jie Li, Ruimao Zhang, Zhen Li, Rui Huang, and Shuguang Cui. Sparse single sweep lidar point cloud segmentation via learning contextual shape priors from scene completion. In *AAAI*, 2021. 2
- [60] Yan Yan, Yuxing Mao, and Bo Li. Second: Sparsely embedded convolutional detection. *Sensors*, 2018. 5
- [61] Chenyu Yang, Yuntao Chen, Hao Tian, Chenxin Tao, Xizhou Zhu, Zhaoxiang Zhang, Gao Huang, Hongyang Li, Yu Qiao, Lewei Lu, et al. Bevformer v2: Adapting modern image backbones to bird’s-eye-view recognition via perspective supervision. In *CVPR*, 2023. 2
- [62] Dongqiangzi Ye, Zixiang Zhou, Weijia Chen, Yufei Xie, Yu Wang, Panqu Wang, and Hassan Foroosh. Lidarmultinet: Towards a unified multi-task network for lidar perception. In *AAAI*, 2023. 7, 2

- [63] Maosheng Ye, Rui Wan, Shuangjie Xu, Tongyi Cao, and Qifeng Chen. Drinet++: Efficient voxel-as-point point cloud segmentation. *arXiv preprint arXiv:2111.08318*, 2021. [2](#)
- [64] Yang Zhang, Zixiang Zhou, Philip David, Xiangyu Yue, Zerong Xi, Boqing Gong, and Hassan Foroosh. Polarnet: An improved grid representation for online lidar point clouds semantic segmentation. In *CVPR*, 2020. [2](#), [6](#), [7](#)
- [65] Yunpeng Zhang, Zheng Zhu, and Dalong Du. Occformer: Dual-path transformer for vision-based 3d semantic occupancy prediction. In *ICCV*, 2023. [1](#), [6](#), [2](#)
- [66] Hao Zheng, Zhanlei Yang, Wenju Liu, Jizhong Liang, and Yanpeng Li. Improving deep neural networks using softplus units. In *IJCNN*, 2015. [1](#)
- [67] Brady Zhou and Philipp Krähenbühl. Cross-view transformers for real-time map-view semantic segmentation. In *CVPR*, 2022. [1](#)
- [68] Zixiang Zhou, Yang Zhang, and Hassan Foroosh. Panoptic-polarnet: Proposal-free lidar point cloud panoptic segmentation. In *CVPR*, 2021. [2](#), [7](#)
- [69] Xizhou Zhu, Weijie Su, Lewei Lu, Bin Li, Xiaogang Wang, and Jifeng Dai. Deformable detr: Deformable transformers for end-to-end object detection. In *ICLR*, 2020. [3](#)
- [70] 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. In *CVPR*, 2021. [6](#), [2](#)