OccNeRF: Advancing 3D Occupancy Prediction in LiDAR-Free Environments

Chubin Zhang, Juncheng Yan, Yi Wei, Jiaxin Li, Li Liu,
Yansong Tang, *Member, IEEE*, Yueqi Duan, *Member, IEEE*, and Jiwen Lu, *Fellow, IEEE*

***Abstract*—Occupancy prediction reconstructs 3D structures of surrounding environments. It provides detailed information for autonomous driving planning and navigation. However, most existing methods heavily rely on the LiDAR point clouds to generate occupancy ground truth, which is not available in the vision-based system. In this paper, we propose an OccNeRF method for training occupancy networks without 3D supervision. Different from previous works which consider a bounded scene, we parameterize the reconstructed occupancy fields and reorganize the sampling strategy to align with the cameras' infinite perceptive range. The neural rendering is adopted to convert occupancy fields to multi-camera depth maps, supervised by multi-frame photometric consistency. Moreover, for semantic occupancy prediction, we design several strategies to polish the prompts and filter the outputs of a pretrained open-vocabulary 2D segmentation model. Extensive experiments for both self-supervised depth estimation and 3D occupancy prediction tasks on nuScenes and SemanticKITTI datasets demonstrate the effectiveness of our method. The code is available at https://github.com/LinShan-Bin/OccNeRF.**

***Index Terms*—3D occupancy prediction, LiDAR-free, self-supervised depth estimation**

I. Introduction

RECENT years have witnessed the great process of autonomous driving [1], [2], [3], [4]. As a crucial component, 3D perception helps the model to understand the real 3D world. Although LiDAR provides a direct means to capture geometric data, its adoption is hindered by the expense of sensors and the sparsity of scanned points. In contrast, as a cheap while effective solution, the vision-centric methods [5], [6], [7], [8], [4], [9] have received more and more attention. Among various 3D scene understanding tasks, multi-camera 3D object detection [2], [1], [10], [11] plays an important role in autonomous systems. However, it struggles to detect objects from infinite classes and suffers from long-tail problems as illustrated in [3], [12].

Complementary to 3D object detection, 3D occupancy prediction [13], [14], [15], [16] reconstructs the geometric structure of surrounding scenes directly, which naturally alleviates the problems mentioned above. As mentioned in [3], 3D occupancy is a good 3D representation for multi-camera scene reconstruction since it has the potential to reconstruct occluded parts and guarantees multi-camera consistency. Recently, some methods have been proposed to lift image features to the 3D space and further predict 3D occupancy. However, most of these methods need 3D occupancy labels for supervision. While some previous research [3], [13] has employed multi-frame LiDAR point accumulation to automatically label occupancy ground truth, the volume of LiDAR data is significantly less than that of image data. Collecting LiDAR data requires specialized vehicles equipped with LiDAR sensors, which is costly. Moreover, this approach neglects a vast quantity of unlabeled multi-camera image data. Consequently, investigating LiDAR-free methods for training occupancy presents a promising research avenue.

To address this, we propose an OccNeRF method, which targets at training multi-camera occupancy networks without 3D supervision. The overview of our proposed method is shown in Fig. 1. We first utilize a 2D backbone to extract multi-camera 2D features. To save memory, we directly interpolate 2D features to obtain 3D volume features instead of using heavy cross-view attention. In previous works, the volume features are supervised by the bounded occupancy labels (*e.g.*

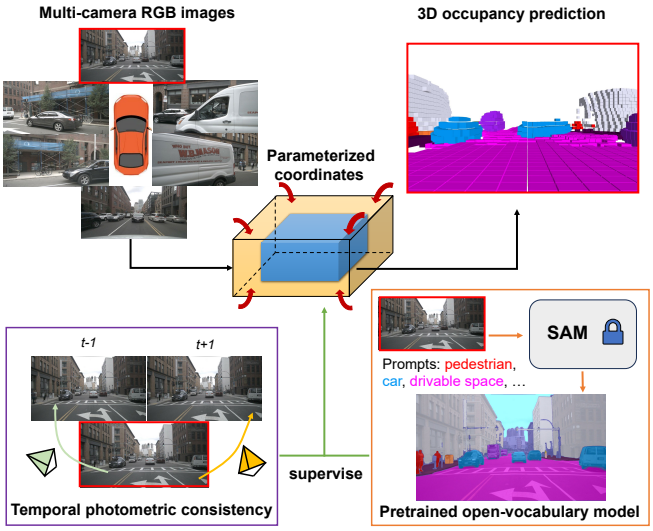


Fig. 1: **The overview of OccNeRF.** To represent unbounded scenes, we propose a parameterized coordinate to contract infinite space to bounded occupancy fields. Without using any LiDAR data or annotated labels, we leverage temporal photometric constraints and pretrained open-vocabulary segmentation models to provide geometric and semantic supervision.

The first three authors contribute equally.
Chubin Zhang and Yansong Tang are with the Shenzhen International Graduate School, Tsinghua University, Shenzhen, 518055, China. Email: zcb24@mails.tsinghua.edu.cn; tang.yansong@sz.tsinghua.edu.cn.
Juncheng Yan, Yi Wei and Jiwen Lu are with the Department of Automation, Yueqi Duan is with the Department of Electronic Engineering, Tsinghua University, Beijing, 100084, China. Jiaxin Li is with Gaussian Robotics, Shanghai, 201100, China. Li Liu is with Xiaomi EV, Beijing, 100085, China.

50m range) and they only need to predict the occupancy with finite resolution (*e.g.* 200 × 200 × 16). Differently, for LiDAR-free training, we should consider unbounded scenes since the RGB images perceive an infinite range. To this end, we parameterize the occupancy fields to represent unbounded environments. Specifically, we split the whole 3D space into the inside and outside regions. The inside one maintains the original coordinate while the outside one adopts a contracted coordinate. A specific sampling strategy is designed to transfer parameterized occupancy fields to 2D depth maps with the neural rendering algorithm.

A straightforward way to supervise predicted occupancy is to calculate loss between rendered images and training images, which is the same as the loss function used in NeRF [17]. However, our experiments indicate this method's ineffectiveness due to the sparse nature of surrounding views, where minimal image overlap fails to supply sufficient geometric information. As an alternative, we take full advantage of temporal information by rendering multiple frames in a sequence and employing photometric consistency between adjacent frames as the primary supervision signal. For semantic occupancy, we propose three strategies to map the class names to the prompts, which are fed to a pretrained open-vocabulary segmentation model [18], [19] to get 2D semantic labels. Then an additional semantic head is employed to render semantic images and supervised by these labels. To verify the effectiveness of our method, we conduct experiments on both self-supervised multi-camera depth estimation and 3D occupancy prediction tasks. Experimental results show that our OccNeRF outperforms other depth estimation methods by a large margin and achieves comparable performance with some methods using stronger supervision on the nuScenes [20] and SemanticKITTI [21] datasets.

In summary, our principal contributions include:

- We develop a system that trains an occupancy network without the need for LiDAR data, addressing the challenge of sparse surrounding views by integrating temporal information for more geometry information.
- We introduce a parameterized occupancy field that enables vision-centric systems to efficiently represent unbounded scenes, aligning with the extensive perceptual capabilities of the cameras.
- We devise a pipeline to generate high-quality pseudo labels with pretrained open-vocabulary segmentation models, with three prompt strategies to improve the accuracy.

II. Related Work

This section examines three interrelated areas in computer vision: 3D occupancy prediction, neural radiance fields, and self-supervised depth estimation. We highlight key advancements and ongoing challenges, providing a critical overview that identifies gaps in current research and suggests avenues for further investigation.

A. 3D Occupancy Prediction

Due to the significance of the vision-centric autonomous driving systems, more and more researchers begin to focus on 3D occupancy prediction tasks [16], [22], [3], [13], [23], [14], [15], [24], [25], [26], [27], [28], [29]. In the industry community, 3D occupancy is treated as an alternative to LiDAR perception. As one of the pioneering works, MonoScene [16] extracts the voxel features generated by sight projection to reconstruct scenes from a single image. TPVFormer [22] further extends it to multi-camera fashion with tri-perspective view representation. Beyond TPVFormer, SurroundOcc [3] designs a pipeline to generate dense occupancy labels instead of using sparse LiDAR points as the ground truth. In addition, a 2D-3D UNet with cross-view attention layers is proposed to predict dense occupancy. RenderOcc [30] uses the 2D depth maps and semantic labels to train the model, reducing the dependence on expensive 3D occupancy annotations. Compared with these methods, our method does not need any annotated 3D or 2D labels. Occ3D [13] establishes the occupancy benchmarks used in CVPR 2023 occupancy prediction challenge and proposes a coarse-to-fine occupancy network. SimpleOccupancy [31] presents a simple while effective framework for occupancy estimation. Although SimpleOccupancy [31] and SelfOcc [**?**] investigate the vision-centric setting, they do not consider the infinite perception range of cameras.

B. Neural Radiance Fields

As one of the most popular topics in 3D area, neural radiance fields (NeRF) [17] have made great achievement in recent years. NeRF [17] learns the geometry of a scene by optimizing a continuous volumetric scene function with multi-view images. To obtain the novel views, volume rendering is performed to convert the radiance fields to RGB images. As a follow-up, mip-NeRF [32] represents the scene at a continuously valued and replaces rays as anti-aliased conical frustums. Beyond mip-NeRF, Zip-NeRF [33] integrates mip-NeRF with a grid-based model for faster training and better quality. There are several extensions of original NeRF, including dynamic scenes [34], [35], [36], [37], [38], [39], [40], 3D reconstruction [41], [42], [43], [44], [45], model accelerating [46], [47], [48], [49], [50], [51], [52], [53], etc. As one of these extensions, some works aim to describe unbounded scenes [54], [55]. NeRF++ [54] split the 3D space as an inner unit sphere and an outer volume and proposes inverted sphere parameterization to represent outside regions. Further, mip-NeRF 360 [55] embeds this idea into mip-NeRF and applies the smooth parameterization to volumes. Inspired by these methods, we also design a parameterization scheme to model unbounded scenes for the occupancy prediction task.

C. Self-supervised Depth Estimation

While early works [56], [57], [58], [59], [60] require dense depth annotations, recent depth estimation methods [61], [62], [63], [64], [65], [66], [67], [68], [69], [70], [71], [72], [73] are designed in a self-supervised manner. Most of these methods predict depth maps and ego-motions simultaneously, adopting the photometric constraints [74], [75] between successive frames as the supervision signal. As a classical work in this field, Monodepth2 [76] proposes some techniques to improve

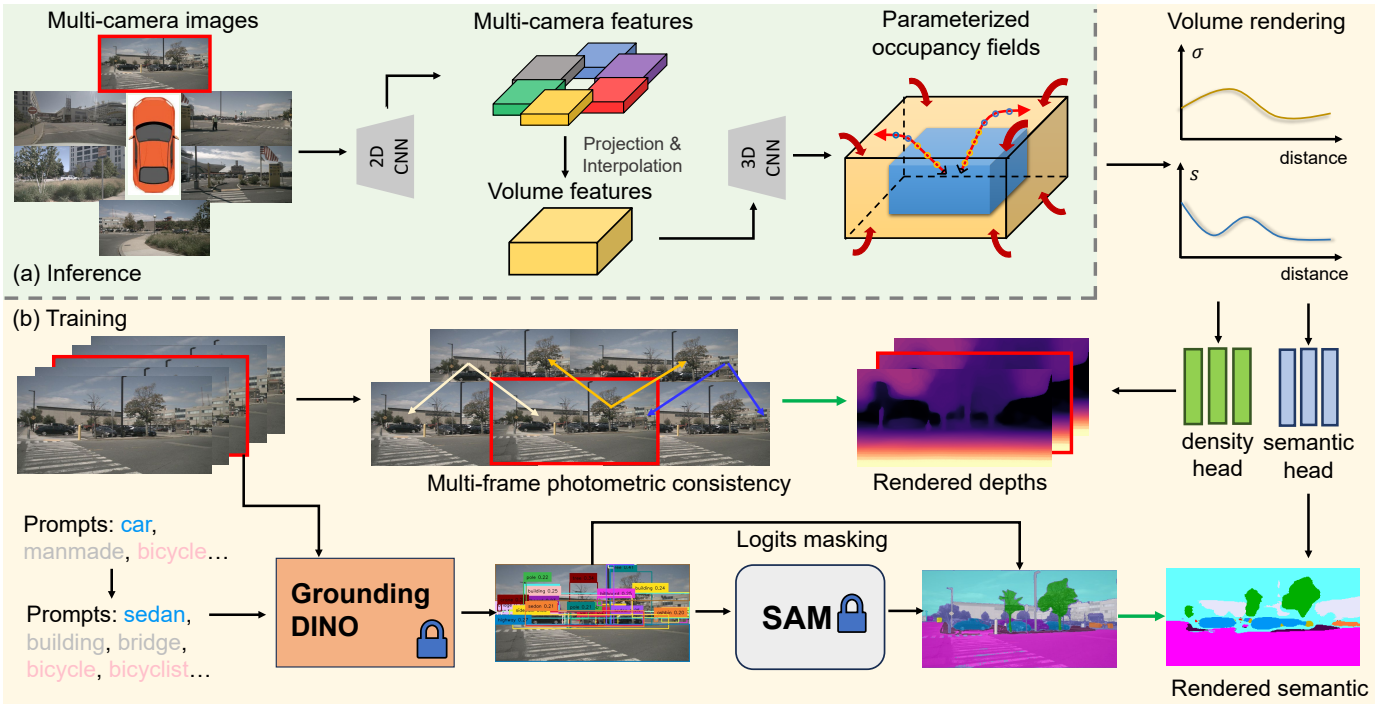


Fig. 2: (a) During inference, a 2D backbone first extracts features from multiple cameras, which are then projected and interpolated into 3D space to create volume features. These are used to reconstruct parameterized occupancy fields that capture the extent of unbounded scenes. (b) During training, to generate rendered depth and semantic maps, we employ volume rendering using a redesigned sampling strategy. The depths from multiple frames are refined through the photometric loss. For the semantic prediction, we utilize pretrained Grounded-SAM model enhanced with prompt cleaning. The green arrow denotes the supervision signal.

the quality of depth predictions, including the minimum reprojection loss, full-resolution multi-scale sampling, and auto-masking loss. Since modern self-driving vehicles are usually equipped with multiple cameras to capture the surrounding views, researchers begin to concentrate on the multi-camera self-supervised depth estimation task [77], [78], [79], [80], [81], [82]. FSM [77] is the first work to extend monocular depth estimation to full surrounding views by leveraging spatiotemporal contexts and pose consistency constraints. To predict the real-world scale, SurroundDepth [78] uses structure-from-motion to generate scale-aware pseudo depths to pretrain the models. Further, it proposes the cross-view transformer and joint pose estimation to incorporate the multi-camera information. Recently, R3D3 [79] combines the feature correlation with bundle adjustment operators for robust depth and pose estimation. Different from these methods, our approach directly extracts features in 3D space, achieving multi-camera consistency and better reconstruction quality.

III. Method

Fig. 2 shows the pipeline of our approach. With the multi-camera images $\{I^i\}_{i=1}^N$ as inputs, we first utilize a 2D backbone to extract N cameras' features $\{X^i\}_{i=1}^N$. Then the 2D features are interpolated to the 3D space to obtain the volume features with known intrinsic $\{K^i\}_{i=1}^N$ and extrinsic $\{T^i\}_{i=1}^N$. As discussed in Section III-A, to represent the unbounded scenes, we propose a coordinate parameterization to contract the infinite range to a limited occupancy field. The volume rendering is performed to convert occupancy fields to multi-frame depth maps, which are supervised by photometric loss. Section III-B introduces this part in detail. Finally, Section III-C shows how we use a pretrained open-vocabulary segmentation model to get 2D semantic labels.

A. Parameterized Occupancy Fields

Different from previous works [3], [14], we need to consider unbounded scenes in the LiDAR-free setting. On the one hand, we should preserve high resolution for the inside region (*e.g.* [-40m, -40m, -1m, 40m, 40m, 5.4m]), since this part covers most regions of interest. On the other hand, the outside region is necessary but less informative and should be represented within a contracted space to reduce memory consumption. Inspired by [55], we propose a transformation function with adjustable regions of interest and contraction threshold to parameterize the coordinates $r = (x, y, z)$ of each voxel grid:

$$f(r) = \begin{cases} \alpha \cdot r' & |r'| \leq 1 \\ \frac{r'}{|r'|} \cdot \left(1 - \frac{a}{|r'|+b}\right) & |r'| > 1 \end{cases}, \tag{1}$$

where $\alpha \in [0, 1]$ represents the proportion of the region of interest in the parameterized space. Higher α indicates we use more space to describe the inside region. $r' = r/r_b$ denotes the normalized coordinate based on the input r and pre-defined inside region bound r_b. The parameters a and b are introduced to maintain the continuity of the first derivative.

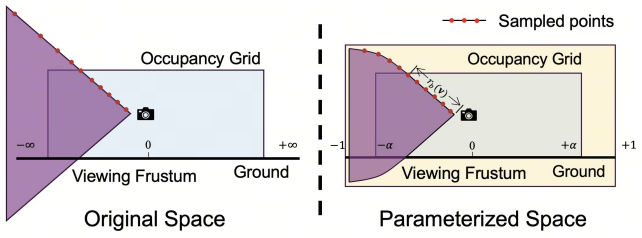


Fig. 3: **Comparison between original space and parameterized space**. The original space utilizes the conventional Euclidean space, emphasizing linear mapping. The parameterized space is divided into two parts: an inner space with linear mapping to preserve high-resolution details and an outer space where point distribution is scaled inversely with distance. This design facilitates the representation of an infinite range within a finite spatial domain.

The determination of these parameters is achieved through the resolution of the ensuing equations:

$$\begin{cases} \lim\limits_{r \to r_b^+} f(r) = \lim\limits_{r \to r_b^-} f(r) \\ \lim\limits_{r \to r_b^+} f'(r) = \lim\limits_{r \to r_b^-} f'(r) \end{cases}, \tag{2}$$

The derived solutions are presented as:

$$\begin{cases} a = \frac{(1-\alpha)^2}{\alpha} \\ b = \frac{1-2\alpha}{\alpha} \end{cases}. \tag{3}$$

To obtain 3D voxel features from 2D views, we first generate the corresponding points $\mathcal{P}_{pc} = [\boldsymbol{x}_{pc}, \boldsymbol{y}_{pc}, \boldsymbol{z}_{pc}]^T$ for each voxel in the parameterized coordinate system and map them back to the ego coordinate system:

$$\mathcal{P} = [f_x^{-1}(\boldsymbol{x}_{pc}), f_y^{-1}(\boldsymbol{y}_{pc}), f_z^{-1}(\boldsymbol{x}_{pc})]^T. \tag{4}$$

Then we project these points to the 2D image feature planes and use bilinear interpolation to get the 2D features:

$$\mathcal{F}^i = X^i \left\langle \text{proj}(\mathcal{P}, T^i, K^i) \right\rangle. \tag{5}$$

where proj is the function projecting 3D points $\mathcal{P}$ to the 2D image plane defined by the camera extrinsic T^i and intrinsic K^i, $\langle\rangle$ is the bilinear interpolation operator, $\mathcal{F}^i$ is the interpolation result. To simplify the aggregation process and reduce computation costs, we directly average the multi-camera 2D features to get volume features, which is the same as the method used in [31], [83]. Finally, a 3D convolution network [84] is employed to extract features and predict the final occupancy outputs.

B. Multi-frame Depth Estimation

To project the occupancy fields to multi-camera depth maps, we adopt volume rendering [85], which is widely used in NeRF-based methods [17], [54], [32]. To render the depth value of a given pixel, we cast a ray from the camera center o along the direction d pointing to the pixel. The ray is represented by $\mathrm{v}(t) = \mathrm{o} + t\mathrm{d}, t \in [t_n, t_f]$. Then, we sample L points $\{t_k\}_{k=1}^{L}$ along the ray in 3D space to get the density $\sigma(t_k)$. For the selected L quadrature points, the depth of the corresponding pixel is computed by:

$$D(\mathrm{v}) = \sum_{k=1}^{L} T(t_k)(1 - \exp(-\sigma(t_k)\delta_k))t_k, \tag{6}$$

where $T(t_k) = \exp\left(-\sum_{k'=1}^{k-1} \sigma(t_k)\delta_k\right)$, and $\delta_k = t_{k+1} - t_k$ are intervals between sampled points.

A vital problem here is how to sample $\{t_k\}_{k=1}^{L}$ in our proposed coordinate system. Uniform sampling in the depth space or disparity space will result in an unbalanced series of points in either the outside or inside region of our parameterized grid, which is to the detriment of the optimization process. With the assumption that o is around the coordinate system's origin, we directly sample $L(\mathrm{r})$ points from $U[0, 1]$ in parameterized coordinate and use the inverse function of Equation 1 to calculate the $\{t_k\}_{k=1}^{L(\mathrm{v})}$ in the ego coordinates. The specific $L(\mathrm{v})$ and $r_b(\mathrm{v})$ for a ray are calculated by:

$$\begin{aligned} r_b(\mathrm{v}) &= \frac{\sqrt{(\mathrm{d} \cdot \mathrm{i} l_x)^2 + (\mathrm{d} \cdot \mathrm{j} l_y)^2 + (\mathrm{d} \cdot \mathrm{k} l_z)^2}}{2\|\mathrm{d}\|}, \\ L(\mathrm{v}) &= \frac{2 r_b(\mathrm{v})}{\alpha d_v} \end{aligned} \tag{7}$$

where $\mathrm{i}, \mathrm{j}, \mathrm{k}$ are the unit vectors in the x, y, z directions, l_x, l_y, l_z are the lengths of the inside region, and d_v is the voxel size. To better adapt to the occupancy representation, we directly predict the rendering weight instead of the density.

A conventional supervision method is to calculate the difference between rendered images and raw images, which is employed in NeRF [17]. However, our experimental results show that it does not work well. The possible reason is that the large-scale scene and few view supervision make it difficult for NeRF to converge. To better make use of temporal information, we employ the photometric loss proposed in [76], [74]. Specifically, we project adjacent frames to the current frames according to the rendered depths and given relative poses. Then we calculate the reconstruction error between projected images and raw images:

$$\mathcal{L}_{pe}^i = \frac{\beta}{2}(1 - \text{SSIM}(I^i, \hat{I}^i)) + (1 - \beta)\|I^i, \hat{I}^i\|_1, \tag{8}$$

where $\hat{I}^i$ is the projected image and $\beta = 0.85$. Moreover, we adopt the techniques introduced in [76], *i.e.* per-pixel minimum reprojection loss and auto-masking stationary pixels. For each camera view, we render a short sequence instead of a single frame and perform multi-frame photometric loss.

C. Semantic Supervision

To enhance the richness of occupancy voxel information and facilitate comparison with existing methods, we introduce 2D labels to provide semantic supervision. Previous works [13], [86] project 3D LiDAR points with segmentation labels to the image space to avoid the expensive cost of annotating dense 3D occupancy. However, we aim to predict semantic occupancy in a fully vision-centric system and use 2D data only. To this end, we leverage a pretrained open-vocabulary

(a) RGB image (b) LiDAR projection labels

(c) Our Grounding DINO bboxs (d) Our Grounded-SAM labels

Fig. 4: **Label generation**. Detection bounding boxes generated by our Grounding DINO and semantic labels predicted by SAM in our method exhibit precision, which is comparable with that of LiDAR points projection labels.

model Grounded-SAM [18], [19], [87] to generate 2D semantic segmentation labels. Without any 2D or 3D ground truth data, the pretrained open-vocabulary model enables us to obtain 2D labels which closely match the semantics of the given category names. This method can easily extend to any dataset, making our approach efficient and generalizable.

Specifically, when dealing with $\boldsymbol{c}$ categories, we employ three strategies to determine the prompts provided to the Grounding DINO. These strategies consist of synonymous substitution, where we replace words with their synonyms (e.g., changing 'car' to 'sedan' to enable the model to distinguish it from 'truck' and 'bus'); splitting single words into multiple entities (e.g., 'manmade' is divided into 'building', 'billboard', and 'bridge' etc. to enhance differentiation); and incorporating additional information (e.g., introducing 'bicyclist' to facilitate the detection of a person on a bike). See TABLE I for more details. Subsequently, we obtain detection bounding boxes along with their corresponding logits and phrases, which are fed to SAM [18] to generate M precise segmentation binary masks. After multiplying the Grounding DINO logits with binary masks, every pixel has $\{l_i\}_{i=1}^{M}$ logits. We get the per-pixel label $\mathcal{S}^{pix}$ using:

$$\mathcal{S}^{pix} = \psi(\arg\max_i l_i), \tag{9}$$

where $\psi(\cdot)$ is a function that maps the index of l_i to the category label according to the phrases. If a pixel does not belong to any categories and gets M zero logits, we will give it an 'uncertain' label. The generated detection bounding boxes and semantic labels are shown in Fig. 4.

To leverage the 2D semantic supervision, we initially utilize a semantic head with $\boldsymbol{c}$ output channels to map volume features extracted to semantic outputs, denoted as $S(x)$. Similar to the method outlined in Section III-B, we engage in volume rendering once more using the subsequent equation:

$$\hat{\mathcal{S}}^{pix}(\mathrm{r}) = \sum_{k=1}^{L_s} T(t_k)(1-\exp(-\sigma(t_k)\delta_k))S(t_k), \tag{10}$$

TABLE I: **Details of prompt strategy.**

Original labels	Ours
car	sedan
bicycle	bicycle bicyclist
vegetation	tree
motorcycle	motorcycle motorcyclist
drivable surface	highway
traffic cone	cone
construction vehicle	crane
manmade	building compound bridge pole billboard light ashbin

where $\hat{\mathcal{S}}^{pix}$ represents the per-pixel semantic rendering output. To save the memory and improve efficiency, we do not render the pixels that are assigned with 'uncertain' labels. Moreover, we only render the central frame instead of multiple frames and reduce the sample ratio to $L_s = L/4$. Our overall loss function is expressed as:

$$\mathcal{L}_{total} = \sum_i \mathcal{L}_{pe}^i + \lambda \mathcal{L}_{sem}^i(\hat{\mathcal{S}}^{pix}, \mathcal{S}^{pix}) \tag{11}$$

where $\mathcal{L}_{sem}$ is the cross-entropy loss function and λ is the semantic loss weight.

IV. Experiment

A. Experimental Setup

Dataset: Our experiments are conducted on nuScenes [20] and SemanticKITTI [21] datasets. NuScenes [20] is a large-scale autonomous driving dataset which contains 600 scenes for training, 150 scenes for validation, and 150 for testing. The dataset has about 40000 frames and 17 classes in total. For self-supervised depth estimation, we project LiDAR point clouds to each view to get depth ground truth for evaluation. Following SurroundDepth [78], we clip the depth prediction and ground truth from 0.1m to 80m. To evaluate the semantic occupancy prediction, we use Occ3D-nuScenes [13] benchmark. The range of each sample is [-40 m, -40 m, -1 m, 40 m, 40 m, 5.4 m] and the voxel size is 0.4 m. Among 17 classes, we do not consider 'other' and 'other flat' classes for evaluation since open-vocabulary models cannot recognize the semantic-ambiguous text. Following [78], [13], we evaluate models on validation sets. Additionally, we explore 3D occupancy prediction using the SemanticKITTI [21] dataset. SemanticKITTI [21] is composed of 22 sequences (10 for training, 1 for validation and 11 for test) of scans and each scan contains voxelized LiDAR data and corresponding stereo images. The range of each sample is [-25.6 m, 0, -2.0 m, 25.6 m, 51.2 m, 4.4 m] and the voxel size is 0.2 m.

Implementation Details: We adopt ResNet-101 [88] with ImageNet [89] pretrained weights as the 2D backbone to extract multi-camera features. For nuScenes [20], the predicted occupancy field has the shape 300x300x24. The central 200x200x16 voxels represent inside regions: -40m to 40m for the X and Y axis, and -1m to 5.4m for the Z axis, which is the same as the scope defined in Occ3D-nuScenes. For SemanticKITTI [21], the shape of the predicted occupancy field is 320x320x40. The central part is 256x256x32, also

TABLE II: **Comparisons for self-supervised multi-camera depth estimation on the nuScenes dataset [20]**. The results are averaged over all views without median scaling at test time. 'FSM*' is the reproduced result in [80] .

Method	Abs Rel	Sq Rel	RMSE	RMSE log	$\delta < 1.25$	$\delta < 1.25^2$	$\delta < 1.25^3$
FSM [77]	0.297	-	-	-	-	-	-
FSM* [77]	0.319	7.534	7.860	0.362	0.716	0.874	0.931
SurroundDepth [78]	0.280	4.401	7.467	0.364	0.661	0.844	0.917
Kim *et al.* [80]	0.289	5.718	7.551	0.348	0.709	0.876	0.932
R3D3 [79]	0.253	4.759	7.150	-	0.729	-	-
SimpleOcc [31]	0.224	3.383	7.165	0.333	0.753	0.877	0.930
OccNeRF	**0.202**	**2.883**	**6.697**	**0.319**	**0.768**	**0.882**	**0.931**

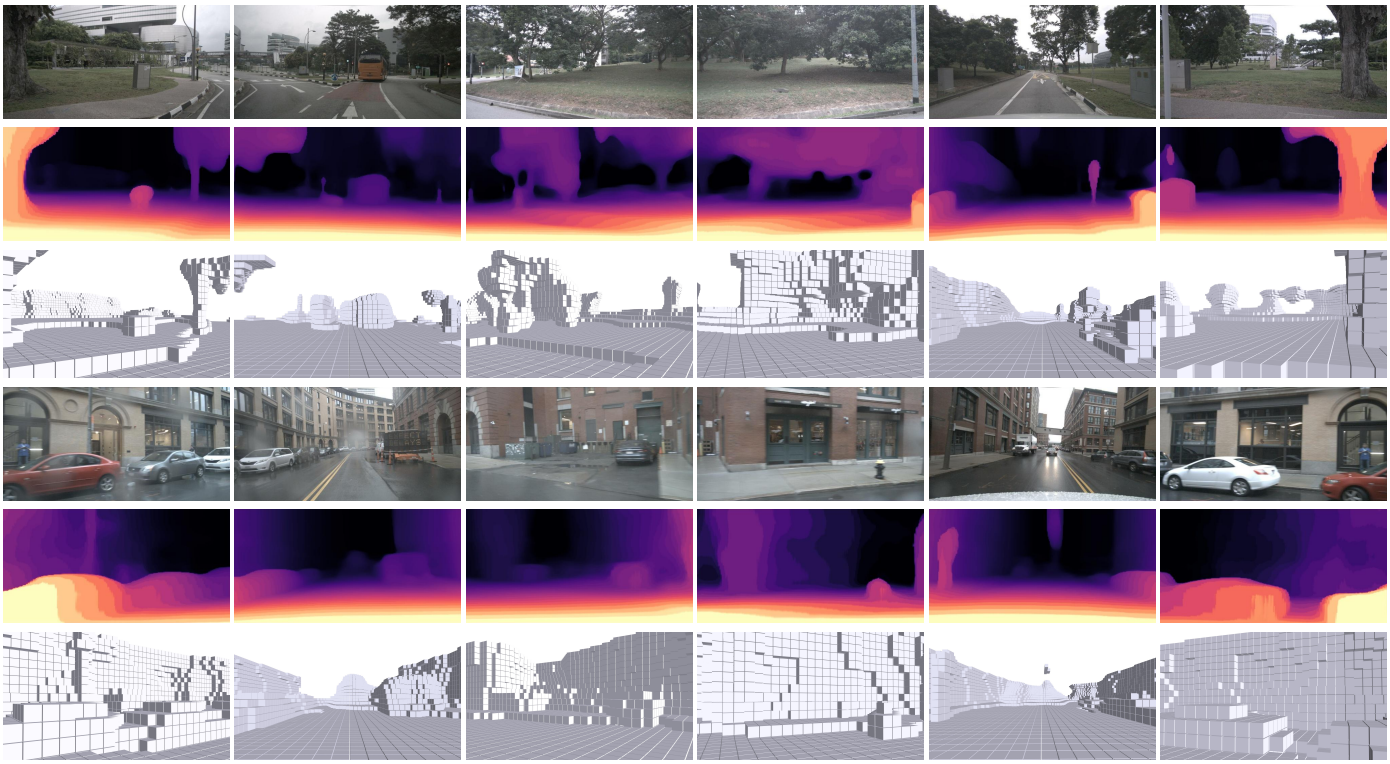


Fig. 5: **Qualitative results on nuScenes dataset [20]**. Our method can predict visually appealing depth maps with texture details and fine-grained occupancy. Better viewed when zoomed in.

representing the scope defined by the dataset. We render 3 frame depth maps, which are supervised by the photometric loss with a sequence of 5 frame raw images (1 keyframe with 4 neighbored non-key frames). The α is set as 0.667. To predict semantic occupancy, the Grounded-SAM [19], [18] is employed as our pretrained open-vocabulary model. The text and box thresholds are set as 0.2 and we use the loss weight $\lambda = 0.05$. All experiments are conducted on 8 A100.

B. Self-supervised Depth Estimation

Evaluation Metric: For depth estimation, we use the commonly used depth evaluation metrics [76], [74], [78] as outlined in the following:

- Abs Rel: $\frac{1}{|T|}\sum_{d\in T}|d-d^*|/d^*$,
- Sq Rel: $\frac{1}{|T|}\sum_{d\in T}|d-d^*|^2/d^*$,
- RMSE: $\sqrt{\frac{1}{|T|}\sum_{d\in T}|d-d^*|^2}$,
- RMSE log: $\sqrt{\frac{1}{|T|}\sum_{d\in T}|\log d-\log d^*|^2}$,
- $\delta < t$: % of d s.t. $\max(\frac{d}{d^*},\frac{d^*}{d}) = \delta < t$,

where d and d^* indicate predicted and ground truth depths respectively, and T indicates all pixels on the depth image D. In our experiments, all the predicted depth maps are scale-aware and we do not perform any scale alignment. The Abs Rel is the main metric for depth estimation tasks and it reveals the relative errors of estimated depths. Note that during evaluation we do not perform median scaling since our method can predict real-world scale given ground truth poses. For the 3D occupancy predictiontask, we use the mean intersection over union (mIoU) of all classes as the semantic-aware metric and the intersection over union (IoU), precision and recall as the semantic-agnostic metrics.

TABLE II shows the self-supervised multi-camera depth estimation results on nuScenes dataset. We do not use pre-

TABLE III: **3D Occupancy prediction performance on the Occ3D-nuScenes dataset [13].** 'GT' indicates occupancy ground truth. Since 'other' and 'other flat' classes are the invalid prompts for open-vocabulary models, we do not consider these two classes during evaluation. 'mIoU*' is the original result, and 'mIoU' is the result ignoring the classes.

Method	GT	mIoU	mIoU*	barrier	bicycle	bus	car	const. veh.	motorcycle	pedestrian	traffic cone	trailer	truck	drive. suf.	sidewalk	terrain	manmade	vegetation
MonoScene [16]	✓	6.33	6.06	7.23	4.26	4.93	9.38	5.67	3.98	3.01	5.90	4.45	7.17	14.91	7.92	7.43	1.01	7.65
TPVFormer [22]	✓	28.69	27.83	38.90	13.67	**40.78**	**45.90**	**17.23**	19.99	18.85	14.30	**26.69**	**34.17**	55.65	37.55	30.70	19.40	16.78
BEVDet [11]	✓	20.03	19.38	30.31	0.23	32.26	34.47	12.97	10.34	10.36	6.26	8.93	23.65	52.27	26.06	22.31	15.04	15.10
OccFormer [14]	✓	22.39	21.93	30.29	12.32	34.40	39.17	14.44	16.45	17.22	9.27	13.90	26.36	50.99	34.66	22.73	6.76	6.97
BEVFormer [2]	✓	28.13	26.88	37.83	17.87	40.44	42.43	7.36	23.88	21.81	20.98	22.38	30.70	55.35	36.0	28.06	20.04	17.69
CTF-Occ [13]	✓	**29.54**	**28.53**	**39.33**	**20.56**	38.29	42.24	16.93	**24.52**	**22.72**	**21.05**	22.98	31.11	53.33	37.98	33.23	**20.79**	18.00
RenderOcc [30]	✓	24.53	23.93	27.56	14.36	19.91	20.56	11.96	12.42	12.14	14.34	20.81	18.94	**68.85**	**42.01**	**43.94**	17.36	**22.61**
SimpleOcc [31]	×	7.99	7.05	0.67	1.18	3.21	7.63	1.02	0.26	1.80	0.26	1.07	2.81	40.44	18.30	17.01	13.42	10.84
OccNeRF	×	10.81	9.53	0.83	0.82	5.13	12.49	3.50	0.23	3.10	1.84	0.52	3.90	52.62	20.81	24.75	18.45	13.19

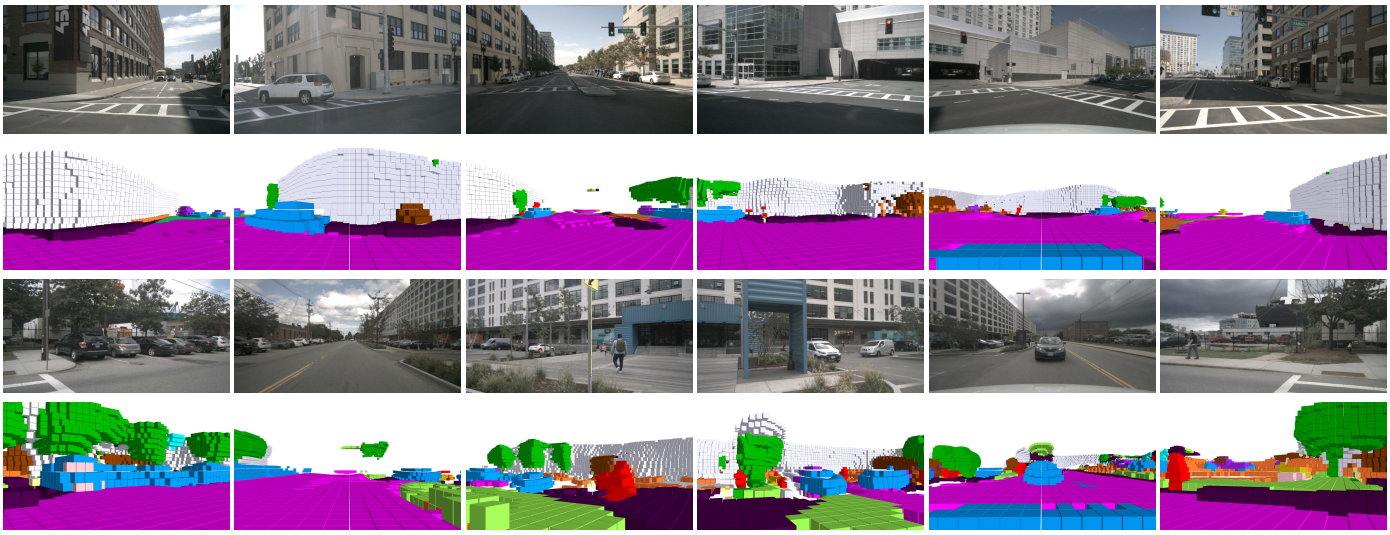


Fig. 6: **Qualitative results of semantic occupancy on nuScenes dataset [20]**. Our method can predict visually appealing semantic occupancy with well geometry correspondence. Better viewed when zoomed in.

trained segmentation models in this experiment. The results are averaged over 6 cameras and 'FSM*' is the reproduced FSM [77] result reported in [80]. We can see that our method outperforms other SOTA methods by a large margin, demonstrating the effectiveness of OccNeRF. Compared to previous depth estimation methods, our approach directly predicts an occupancy field in 3D space instead of predicting per-pixel depths. This naturally guarantees multi-camera consistency. Furthermore, our method eliminates the need for post-processing steps that lift 2D depths to 3D point clouds.

C. Occupancy Prediction

We conduct experiments on semantic occupancy prediction using the Occ3D-nuScenes dataset. The pretrained open-vocabulary model [18], [19] struggles with ambiguous prompts like 'other' and 'other flat', so we exclude them during the evaluation. To compare with the SimpleOcc method [31], we add a semantic head to the original model and leverage the generated 2D semantic labels to train it. Our approach significantly outperforms SimpleOcc, as detailed in TABLE III, and achieves competitive results against some fully-supervised methods. Notably, it excels in predicting 'drivable space' and 'manmade' classes, outdoing all supervised methods. However, it falls short in detecting small objects, such as bicycles and pedestrians, where it lags behind state-of-the-art supervised methods, likely due to the open-vocabulary model's limitations in capturing small details.

We further explored the geometry occupancy prediction task with the nuScenes [20] and SemanticKITTI [21]. Since most works reported in occ3D-nuScenes [13] do not provide codes,

TABLE IV: **3D Occupancy prediction performance on the SemanticKITTI dataset [21].** The results of other methods are from the table in SceneRF [25]. MonoScene* is supervised by depth predictions from [76]

Method	Supervision 3D	Depth	Image	IoU	Prec.	Rec.
MonoScene [16]	✓			37.14	49.90	59.24
LMSCNet^{rgb} [90]		✓		12.08	13.00	63.16
3DSketch^{rgb} [91]		✓		12.01	12.95	62.31
AICNet^{rgb} [92]		✓		11.28	11.84	70.89
MonoScene [16]		✓		13.53	16.98	40.06
MonoScene* [16]			✓	11.18	13.15	40.22
SceneRF [25]			✓	13.84	17.28	**40.96**
OccNeRF			✓	**22.81**	**35.25**	39.27

TABLE V: **The scene reconstruction performance on the Occ3D-nuScenes dataset [13]**. The results of other methods are reproduced with their released codes.

Method	GT	IoU	Prec.	Rec.
RenderOcc [30]	✓	53.09	59.97	82.23
SimpleOcc [31]	×	33.92	41.91	64.02
OccNeRF	×	39.20	57.20	55.47

we can only evaluate RenderOcc [30] and SeimpleOcc [31]. As detailed in TABLE IV and TABLE V, our approach outperforms other methods supervised by images and achieves competitive results against methods with stronger supervision.

D. Ablation Study

Supervision Method: A straightforward supervision signal is a difference between the rendered and true pixel colours, which is the same as the loss function used in NeRF [17]. However, as shown in TABLE VI, this supervision method yields terrible performance. We attribute this to the challenge NeRF faces in learning the scene structure with only six views. On the contrary, temporal photometric loss ('Depth' in the table) can better leverage geometric cues in adjacent frames, which is the golden metric in self-supervised depth estimation methods. Moreover, multi-frame training provides stronger supervision, further boosting the model's performance.

Coordinate Parameterization: TABLE VII shows the ablation study of coordinate parameterization. Different from occupancy labels, the photometric loss assumes that the images perceive an infinite range. The contracted coordinate aims to represent the unbounded scene in a bounded occupancy. From the table, we can see that the contracted coordinate greatly improves the model's performance. In addition, since the parameterized coordinate is not the Euclidean 3D space, the proposed sampling strategy works better than normal uniform sampling in the original ego coordinate.

Semantic Label Generation: In this subsection, we conduct ablation studies of semantic label generation on the nuScenes [20] dataset. First, we change grounding DINO [19] logits as SAM logits [18] to get semantic labels. As shown in TABLE VIII and Fig. 8, we find that the SAM logits are more noisy and discontinuous. Then, we also feed raw category names to the open-vocabulary model without proposed prompting strategies. However, this method leads to worse results since the original class names cannot provide fine-grained semantic guidance and bring ambiguity.

TABLE VI: **The ablation study of supervision method**. 'Depth' means whether we use the temporal photometric constraints to train the model. 'Multi' indicates whether we employ multi-frame rendering and supervision.

Depth	Multi	Abs Rel	RMSE	$\delta < 1.25$
		0.627	15.901	0.051
	✓	0.489	9.352	0.362
✓		0.216	6.752	0.764
✓	✓	**0.202**	**6.697**	**0.768**

TABLE VII: **The ablation study of coordinate parameterization**. 'CC' means whether we adopt contracted coordinates. 'Resample' indicates whether we leverage the proposed sampling strategy.

CC	Resample	Abs Rel	RMSE	$\delta < 1.25$
		0.216	8.465	0.694
✓		0.208	7.339	0.743
✓	✓	**0.202**	**6.697**	**0.768**

E. Visualization

To further demonstrate the superiority of our method, we provide some qualitative results in Fig. 5 and 6. From Fig. 5, we can see that our method can generate high-quality depth maps and occupancy with fine-grained details. See supplementary material for more qualitative comparisons with other methods. For semantic occupancy prediction, as shown in Fig. 6, our OccNeRF can reconstruct dense results of the surrounding scenes, especially for the large-area categories, such as 'drivable space' and 'manmade'.

F. Analysis

Supervised fine-tuning (SFT). A large success has been achieved in the field of natural language processing by utilizing a methodology combining self-supervised pre-training and supervised fine-tuning. Our work makes it possible to extend this mothodology to 3D occupancy prediction. We explored supervised fine-tuning with our model using 3D occupancy labels from Occ3D [13]. Fig. 9 demonstrates that integrating a fraction of the 3D ground truth significantly boosts performance, nearing that of fully supervised models. Complete fine-tuning with all 3D ground truth data even surpasses the non-pretrained model, highlighting our method's data efficiency.

Computational cost. The inference latency and memory requirements for our approach are detailed in TABLE IX. It is important to note that our method does not require rendering during inference. Our method achieves a comparable efficiency with convolution-based methods like SimpleOcc [31] and is significantly faster than transformer-based methods like SurroundOcc [3]. However, volume rendering will increase

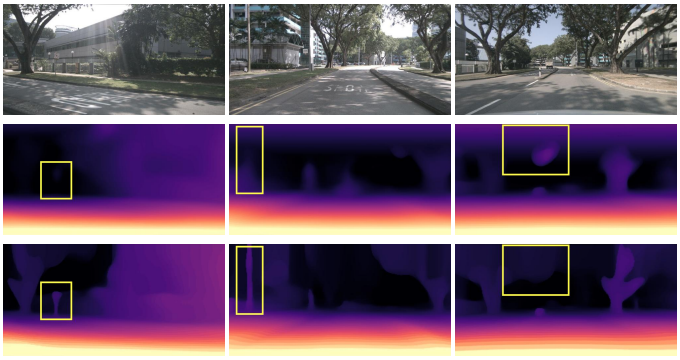


Fig. 7: **Qualitative comparison of different coordinates.** The second line indicates the results without using coordinate parameterization. With the ability to represent unbounded environments, our method can get better results in far scenes, such as the sky.

TABLE VIII: **The ablation study of semantic label generation**. 'SAM logits' means that we directly use the logits from SAM [18]. 'Catagory names' means that we do not conduct the prompting strategies.

Method	SAM logits	Catagory names	Ours
mIoU	7.50	8.23	10.81

training time. For an epoch on 8 A100 GPUs, our method takes 128 mins while SurroundOcc takes 75 mins.

V. Conclusion

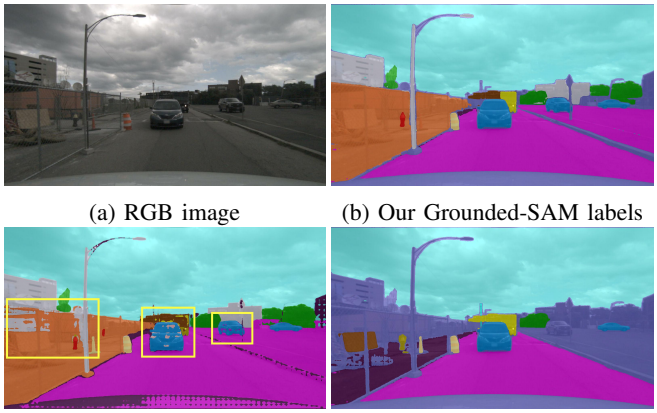
In this paper, we have introduced OccNeRF, a novel approach to train multi-camera occupancy networks without relying on 3D supervision. By leveraging temporal photometric constraints and pretrained open-vocabulary segmentation models, OccNeRF has effectively addressed the challenges of sparse surrounding views and the representation of unbounded scenes. Our method has offered a cost-effective and scalable solution for autonomous driving by eliminating the dependency on expensive LiDAR data and utilizing vast amounts of unlabeled multi-camera image data, similar to the successful self-supervised method used in natural language processing.

Our experiments on the nuScenes and SemanticKITTI datasets have demonstrated that OccNeRF significantly outperforms existing methods in self-supervised multi-camera depth estimation, providing accurate and consistent depth predictions across multiple views. In semantic occupancy prediction, OccNeRF has achieved competitive performance with fully-supervised methods, excelling in large-area categories such as 'drivable space' and 'manmade' structures. However, its performance in detecting smaller objects like bicycles and pedestrians has been limited by the capabilities of open-vocabulary segmentation models in capturing fine details.

While promising, OccNeRF has some limitations, including its current inability to predict dynamic occupancy flows due to the lack of multi-frame information during inference. Future work could address this by incorporating optical flow models and multi-frame inputs. Additionally, improving the granularity and accuracy of open-vocabulary segmentation models could further enhance our system's detection and representation of smaller objects. Overall, OccNeRF represents a significant advancement in vision-centric 3D scene understanding for autonomous driving, offering a robust and efficient approach to developing autonomous systems.

Method	Latency (s)	Memory (G)
BEVFormer [2]	0.28	**4.5**
TPVFormer [22]	0.29	5.1
MonoScene [16]	0.77	20.3
SimpleOcc [31]	**0.16**	8.7
SurroundOcc [3]	0.31	5.9
Ours	0.18	11.0

TABLE IX: **Computational costs.** The results are measured on a single NVIDIA A100.



(a) RGB image (b) Our Grounded-SAM labels

(c) Labels with SAM logits (d) Labels w/o prompt strategies

Fig. 8: **Comparison of different semantic label generation methods**. Compared with generating semantic labels with SAM logits or feeding raw category names, our semantic labels are preciser and have better continuity.

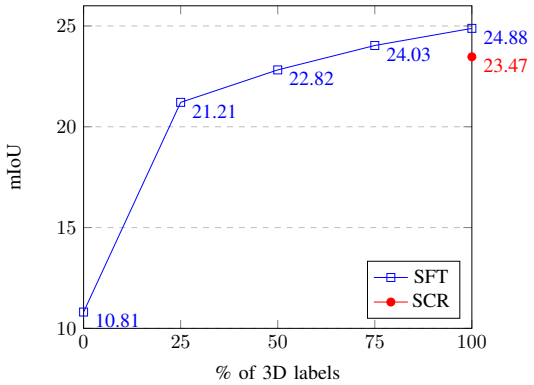


Fig. 9: **Supervised fine-tuning experiment.** 'SFT' stands for fine-tuning the pretrained model with 3D labels. 'SCR' means training from scratch.

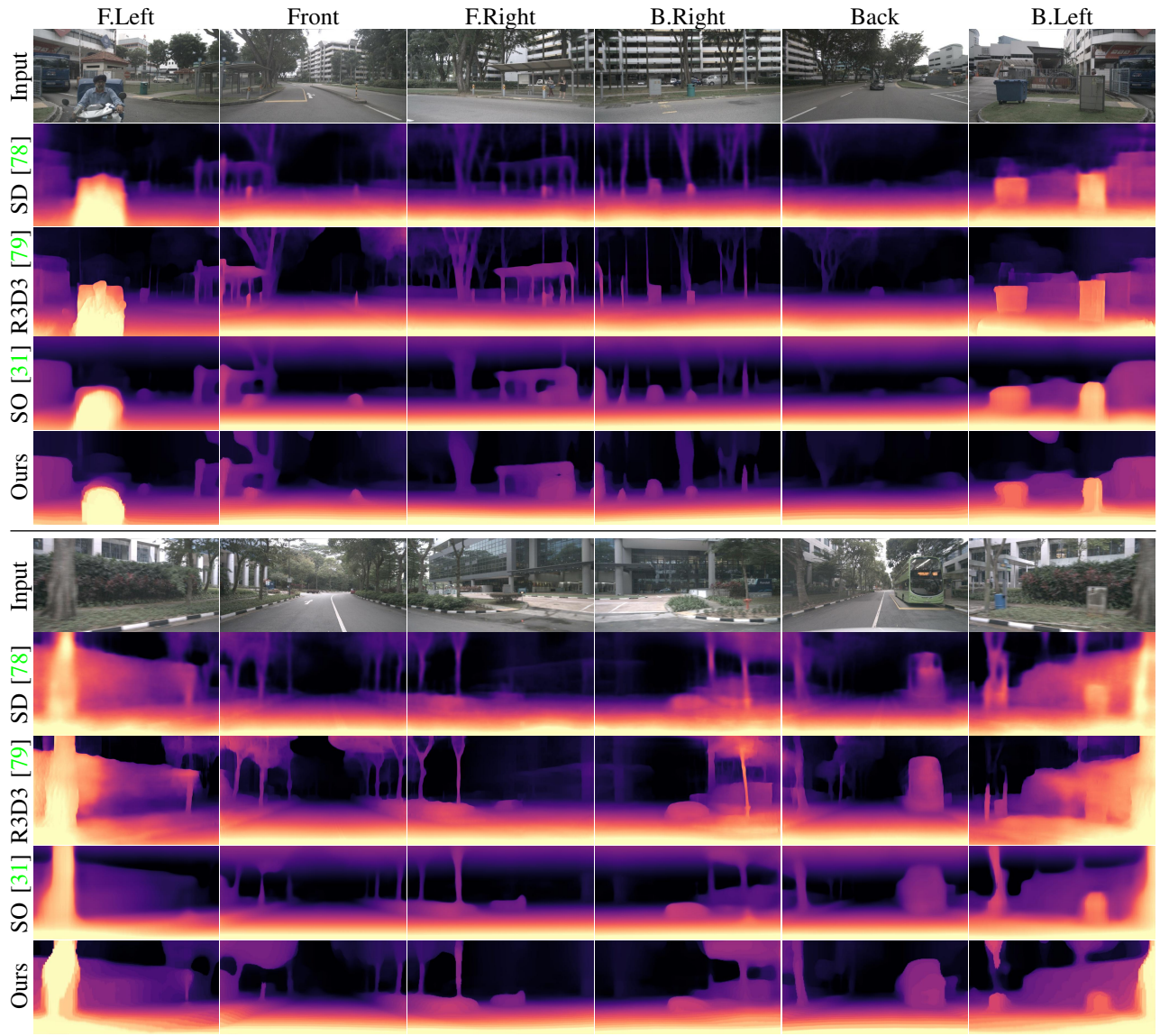


Fig. 10: **Qualitative comparison of the depth estimation task on the nuScenes [20] dataset.** The output depths are presented in absolute terms and normalized between their maximum and minimum values for better visualization.

ACKNOWLEDGMENT

This work was supported in part by the National Natural Science Foundation of China under Grant 62125603, Grant 62321005, and Grant 62336004, and Shenzhen Key Laboratory of Ubiquitous Data Enabling (Grant No. ZDSYS20220527171406015).

APPENDIX A
MORE EXPERIMENTAL RESULTS

Per-camera evaluation: We present per-camera comparisons of our method for the depth estimation task against previous works [77], [78], [79] using the nuScenes dataset [20] in TABLE X. Our approach consistently outperforms other methods across all camera views, showing particularly significant improvements in side views. Side views typically contain more intricate details, posing greater challenges for models to accurately estimate the depths.

TABLE X: **Per-camera comparisons for scale-aware multi-camera depth estimation on the nuScenes dataset**. Tests are conducted within 80 meters.

	Abs Rel ↓					
Method	*Front*	*F.Left*	*F.Right*	*B.Left*	*B.Right*	*Back*
FSM [77]	0.186	0.287	0.375	0.296	0.418	0.221
SurroundDepth [78]	0.179	0.260	0.340	0.282	0.403	0.212
R3D3 [79]	0.174	0.230	0.302	0.249	0.360	0.201
Ours	**0.132**	**0.190**	**0.227**	**0.204**	**0.289**	**0.169**

Qualitative Comparisons: Fig. 10 presents qualitative comparisons on the nuScenes [20] validation set. We visualize results from various state-of-the-art depth estimation and occupancy prediction methods using their official implementations. Compared to these methods, our occupancy-based approach exhibits fewer artifacts and enhanced overall accuracy.

APPENDIX B
THE BENEFITS OF OPEN-VOCABULARY.

(a) Input (b) 2D pseudo labels (c) Occupancy

Fig. 11: **Generating 3D occupancy for an unlabeled class in Occ3D [13].** When prompted with the term 'billboard', our model is capable of predicting the corresponding occupancy, despite the absence of this category in the official annotations.

Fig. 11 illustrates that our pipeline utilizing the open-vocabulary models can generate results for undefined categories within the nuScenes dataset (e.g., 'billboard' in the demonstration case). This adaptability enables our method to extend to arbitrary classes and be applicable to both publicly available and internally collected datasets. Furthermore, it demonstrates the potential for generalizing to rare classes, often referred to as corner cases, that are crucial in autonomous driving scenarios yet typically suffer from insufficient data. In contrast, traditional segmentation models pre-trained on the nuScenes dataset are confined to predefined classes and lack the flexibility to deal with extraordinary situations.

REFERENCES

[1] Y. Li, Z. Ge, G. Yu, J. Yang, Z. Wang, Y. Shi, J. Sun, and Z. Li, "Bevdepth: Acquisition of reliable depth for multi-view 3d object detection," *arXiv preprint arXiv:2206.10092*, 2022.

[2] Z. Li, W. Wang, H. Li, E. Xie, C. Sima, T. Lu, Q. Yu, and J. Dai, "Bevformer: Learning bird's-eye-view representation from multi-camera images via spatiotemporal transformers," in *European Conference on Computer Vision*, 2022.

[3] Y. Wei, L. Zhao, W. Zheng, Z. Zhu, J. Zhou, and J. Lu, "Surroundocc: Multi-camera 3d occupancy prediction for autonomous driving," in *International Conference on Computer Vision*, 2023, pp. 21 729–21 740.

[4] Z. Liu, H. Tang, A. Amini, X. Yang, H. Mao, D. Rus, and S. Han, "Bevfusion: Multi-task multi-sensor fusion with unified bird's-eye view representation," *arXiv preprint arXiv:2205.13542*, 2022.

[5] C. Lu, M. J. G. van de Molengraft, and G. Dubbelman, "Monocular semantic occupancy grid mapping with convolutional variational encoder–decoder networks," *IEEE Robotics and Automation Letters*, vol. 4, no. 2, pp. 445–452, 2019.

[6] Y. Zhang, Z. Zhu, W. Zheng, J. Huang, G. Huang, J. Zhou, and J. Lu, "Beverse: Unified perception and prediction in birds-eye-view for vision-centric autonomous driving," *arXiv preprint arXiv:2205.09743*, 2022.

[7] A. Hu, Z. Murez, N. Mohan, S. Dudas, J. Hawke, V. Badrinarayanan, R. Cipolla, and A. Kendall, "Fiery: Future instance prediction in bird's-eye view from surround monocular cameras," in *International Conference on Computer Vision*, 2021, pp. 15 273–15 282.

[8] A. K. Akan and F. Güney, "Stretchbev: Stretching future instance prediction spatially and temporally," *arXiv preprint arXiv:2203.13641*, 2022.

[9] T. Liang, H. Xie, K. Yu, Z. Xia, Z. Lin, Y. Wang, T. Tang, B. Wang, and Z. Tang, "Bevfusion: A simple and robust lidar-camera fusion framework," *arXiv preprint arXiv:2205.13790*, 2022.

[10] Y. Liu, T. Wang, X. Zhang, and J. Sun, "Petr: Position embedding transformation for multi-view 3d object detection," *arXiv preprint arXiv:2203.05625*, 2022.

[11] J. Huang, G. Huang, Z. Zhu, and D. Du, "Bevdet: High-performance multi-camera 3d object detection in bird-eye-view," *arXiv preprint arXiv:2112.11790*, 2021.

[12] Z. Yu, C. Shu, J. Deng, K. Lu, Z. Liu, J. Yu, D. Yang, H. Li, and Y. Chen, "Flashocc: Fast and memory-efficient occupancy prediction via channel-to-height plugin," *arXiv preprint arXiv:2311.12058*, 2023.

[13] X. Tian, T. Jiang, L. Yun, Y. Wang, Y. Wang, and H. Zhao, "Occ3d: A large-scale 3d occupancy prediction benchmark for autonomous driving," *arXiv preprint arXiv:2304.14365*, 2023.

[14] Y. Zhang, Z. Zhu, and D. Du, "Occformer: Dual-path transformer for vision-based 3d semantic occupancy prediction," in *International Conference on Computer Vision*, 2023.

[15] X. Wang, Z. Zhu, W. Xu, Y. Zhang, Y. Wei, X. Chi, Y. Ye, D. Du, J. Lu, and X. Wang, "Openoccupancy: A large scale benchmark for surrounding semantic occupancy perception," in *International Conference on Computer Vision*, 2023.

[16] A.-Q. Cao and R. de Charette, "Monoscene: Monocular 3d semantic scene completion," in *IEEE Conference on Computer Vision Pattern Recognision*, 2022, pp. 3991–4001.

[17] B. Mildenhall, P. P. Srinivasan, M. Tancik, J. T. Barron, R. Ramamoorthi, and R. Ng, "Nerf: Representing scenes as neural radiance fields for view synthesis," in *European Conference on Computer Vision*. Springer, 2020, pp. 405–421.

[18] A. Kirillov, E. Mintun, N. Ravi, H. Mao, C. Rolland, L. Gustafson, T. Xiao, S. Whitehead, A. C. Berg, W.-Y. Lo *et al.*, "Segment anything," in *International Conference on Computer Vision*, 2023.

[19] S. Liu, Z. Zeng, T. Ren, F. Li, H. Zhang, J. Yang, C. Li, J. Yang, H. Su, J. Zhu *et al.*, "Grounding dino: Marrying dino with grounded pre-training for open-set object detection," *arXiv preprint arXiv:2303.05499*, 2023.

[20] H. Caesar, V. Bankiti, A. H. Lang, S. Vora, V. E. Liong, Q. Xu, A. Krishnan, Y. Pan, G. Baldan, and O. Beijbom, "nuscenes: A multimodal dataset for autonomous driving," in *IEEE Conference on Computer Vision Pattern Recognision*, 2020, pp. 11 621–11 631.

[21] J. Behley, M. Garbade, A. Milioto, J. Quenzel, S. Behnke, C. Stachniss, and J. Gall, "Semantickitti: A dataset for semantic scene understanding of lidar sequences," in *International Conference on Computer Vision*, 2019, pp. 9297–9307.

[22] Y. Huang, W. Zheng, Y. Zhang, J. Zhou, and J. Lu, "Tri-perspective view for vision-based 3d semantic occupancy prediction," in *IEEE Conference on Computer Vision Pattern Recognision*, 2023, pp. 9223–9232.

[23] S. Zuo, W. Zheng, Y. Huang, J. Zhou, and J. Lu, "Pointocc: Cylindrical tri-perspective view for point-based 3d semantic occupancy prediction," *arXiv preprint arXiv:2308.16896*, 2023.

[24] W. Tong, C. Sima, T. Wang, L. Chen, S. Wu, H. Deng, Y. Gu, L. Lu, P. Luo, D. Lin *et al.*, "Scene as occupancy," in *International Conference on Computer Vision*, 2023, pp. 8406–8415.

[25] A.-Q. Cao and R. de Charette, "Scenerf: Self-supervised monocular 3d scene reconstruction with radiance fields," in *International Conference on Computer Vision*, 2023, pp. 9387–9398.

[26] A. Hayler, F. Wimbauer, D. Muhle, C. Rupprecht, and D. Cremers, "S4c: Self-supervised semantic scene completion with neural fields," *arXiv preprint arXiv:2310.07522*, 2023.

[27] Y. Li, Z. Yu, C. Choy, C. Xiao, J. M. Alvarez, S. Fidler, C. Feng, and A. Anandkumar, "Voxformer: Sparse voxel transformer for camera-based 3d semantic scene completion," in *IEEE Conference on Computer Vision Pattern Recognision*, 2023, pp. 9087–9098.

[28] Z. Li, Z. Yu, W. Wang, A. Anandkumar, T. Lu, and J. M. Alvarez, "Fb-bev: Bev representation from forward-backward view transformations," in *International Conference on Computer Vision*, 2023, pp. 6919–6928.

[29] Z. Li, Z. Yu, D. Austin, M. Fang, S. Lan, J. Kautz, and J. M. Alvarez, "Fb-occ: 3d occupancy prediction based on forward-backward view transformation," *arXiv preprint arXiv:2307.01492*, 2023.

[30] M. Pan, J. Liu, R. Zhang, P. Huang, X. Li, L. Liu, and S. Zhang, "Renderocc: Vision-centric 3d occupancy prediction with 2d rendering supervision," in *IEEE International Conference on Robotics and Automation*, 2024.

[31] W. Gan, N. Mo, H. Xu, and N. Yokoya, "A simple attempt for 3d occupancy estimation in autonomous driving," *arXiv preprint arXiv:2303.10076*, 2023.

[32] J. T. Barron, B. Mildenhall, M. Tancik, P. Hedman, R. Martin-Brualla, and P. P. Srinivasan, "Mip-nerf: A multiscale representation for anti-aliasing neural radiance fields," in *International Conference on Computer Vision*, 2021, pp. 5855–5864.

[33] J. T. Barron, B. Mildenhall, D. Verbin, P. P. Srinivasan, and P. Hedman, "Zip-nerf: Anti-aliased grid-based neural radiance fields," in *International Conference on Computer Vision*, 2023.

[34] K. Park, U. Sinha, J. T. Barron, S. Bouaziz, D. B. Goldman, S. M. Seitz, and R. Martin-Brualla, "Nerfies: Deformable neural radiance fields," in *International Conference on Computer Vision*, 2021, pp. 5865–5874.

[35] C. Gao, A. Saraf, J. Kopf, and J.-B. Huang, "Dynamic view synthesis from dynamic monocular video," in *International Conference on Computer Vision*, 2021, pp. 5712–5721.

[36] A. Pumarola, E. Corona, G. Pons-Moll, and F. Moreno-Noguer, "D-nerf: Neural radiance fields for dynamic scenes," in *IEEE Conference on Computer Vision Pattern Recognision*, 2021, pp. 10 318–10 327.

[37] E. Tretschk, A. Tewari, V. Golyanik, M. Zollhöfer, C. Lassner, and C. Theobalt, "Non-rigid neural radiance fields: Reconstruction and novel view synthesis of a dynamic scene from monocular video," in *International Conference on Computer Vision*, 2021, pp. 12 959–12 970.

[38] Z. Li, S. Niklaus, N. Snavely, and O. Wang, "Neural scene flow fields for space-time view synthesis of dynamic scenes," in *IEEE Conference on Computer Vision Pattern Recognision*, 2021, pp. 6498–6508.

[39] Y. Du, Y. Zhang, H.-X. Yu, J. B. Tenenbaum, and J. Wu, "Neural radiance flow for 4d view synthesis and video processing," in *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*. IEEE Computer Society, 2021, pp. 14 304–14 314.

[40] K. Park, U. Sinha, P. Hedman, J. T. Barron, S. Bouaziz, D. B. Goldman, R. Martin-Brualla, and S. M. Seitz, "Hypernerf: a higher-dimensional representation for topologically varying neural radiance fields," *ACM Transactions on Graphics (TOG)*, vol. 40, no. 6, pp. 1–12, 2021.

[41] P. Wang, L. Liu, Y. Liu, C. Theobalt, T. Komura, and W. Wang, "Neus: Learning neural implicit surfaces by volume rendering for multi-view reconstruction," *Advances in Neural Information Processing Systems*, vol. 34, pp. 27 171–27 183, 2021.

[42] L. Yariv, J. Gu, Y. Kasten, and Y. Lipman, "Volume rendering of neural implicit surfaces," *Advances in Neural Information Processing Systems*, vol. 34, pp. 4805–4815, 2021.

[43] Z. Yu, S. Peng, M. Niemeyer, T. Sattler, and A. Geiger, "Monosdf: Exploring monocular geometric cues for neural implicit surface reconstruction," *Advances in Neural Information Processing Systems*, vol. 35, pp. 25 018–25 032, 2022.

[44] M. Oechsle, S. Peng, and A. Geiger, "Unisurf: Unifying neural implicit surfaces and radiance fields for multi-view reconstruction," in *International Conference on Computer Vision*, 2021, pp. 5589–5599.

[45] Y. Wei, S. Liu, Y. Rao, W. Zhao, J. Lu, and J. Zhou, "Nerfingmvs: Guided optimization of neural radiance fields for indoor multi-view stereo," in *International Conference on Computer Vision*, 2021, pp. 5610–5619.

[46] A. Chen, Z. Xu, A. Geiger, J. Yu, and H. Su, "Tensorf: Tensorial radiance fields," in *European Conference on Computer Vision*. Springer, 2022, pp. 333–350.

[47] S. J. Garbin, M. Kowalski, M. Johnson, J. Shotton, and J. Valentin, "Fastnerf: High-fidelity neural rendering at 200fps," in *International Conference on Computer Vision*, 2021, pp. 14 346–14 355.

[48] C. Reiser, S. Peng, Y. Liao, and A. Geiger, "Kilonerf: Speeding up neural radiance fields with thousands of tiny mlps," in *International Conference on Computer Vision*, 2021, pp. 14 335–14 345.

[49] S. Fridovich-Keil, A. Yu, M. Tancik, Q. Chen, B. Recht, and A. Kanazawa, "Plenoxels: Radiance fields without neural networks," in *IEEE Conference on Computer Vision Pattern Recognision*, 2022, pp. 5501–5510.

[50] T. Müller, A. Evans, C. Schied, and A. Keller, "Instant neural graphics primitives with a multiresolution hash encoding," *ACM Transactions on Graphics*, vol. 41, no. 4, pp. 1–15, 2022.

[51] C. Sun, M. Sun, and H.-T. Chen, "Direct voxel grid optimization: Super-fast convergence for radiance fields reconstruction," in *IEEE Conference on Computer Vision Pattern Recognision*, 2022, pp. 5459–5469.

[52] W. Zhang, R. Xing, Y. Zeng, Y.-S. Liu, K. Shi, and Z. Han, "Fast learning radiance fields by shooting much fewer rays," *IEEE Transactions on Image Processing*, vol. 32, pp. 2703–2718, 2023.

[53] M. Chen, L. Wang, Y. Lei, Z. Dong, and Y. Guo, "Learning spherical radiance field for efficient 360° unbounded novel view synthesis," *IEEE Transactions on Image Processing*, 2024.

[54] K. Zhang, G. Riegler, N. Snavely, and V. Koltun, "Nerf++: Analyzing and improving neural radiance fields," *arXiv preprint arXiv:2010.07492*, 2020.

[55] J. T. Barron, B. Mildenhall, D. Verbin, P. P. Srinivasan, and P. Hedman, "Mip-nerf 360: Unbounded anti-aliased neural radiance fields," in *IEEE Conference on Computer Vision Pattern Recognision*, 2022, pp. 5470–5479.

[56] J. H. Lee, M.-K. Han, D. W. Ko, and I. H. Suh, "From big to small: Multi-scale local planar guidance for monocular depth estimation," *arXiv preprint arXiv:1907.10326*, 2019.

[57] Z. Zhang, C. Xu, J. Yang, J. Gao, and Z. Cui, "Progressive hard-mining network for monocular depth estimation," *IEEE Transactions on Image Processing*, vol. 27, no. 8, pp. 3691–3702, 2018.

[58] H. Fu, M. Gong, C. Wang, K. Batmanghelich, and D. Tao, "Deep ordinal regression network for monocular depth estimation," in *IEEE Conference on Computer Vision Pattern Recognision*, 2018, pp. 2002–2011.

[59] F. Liu, C. Shen, G. Lin, and I. Reid, "Learning depth from single monocular images using deep convolutional neural fields," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 38, no. 10, pp. 2024–2039, 2015.

[60] A. Roy and S. Todorovic, "Monocular depth estimation using neural regression forest," in *IEEE Conference on Computer Vision Pattern Recognision*, 2016, pp. 5506–5514.

[61] V. Guizilini, R. Ambrus, S. Pillai, A. Raventos, and A. Gaidon, "3d packing for self-supervised monocular depth estimation," in *IEEE Conference on Computer Vision Pattern Recognision*, 2020, pp. 2485–2494.

[62] J. Watson, M. Firman, G. J. Brostow, and D. Turmukhambetov, "Self-Supervised Monocular Depth Hints," in *International Conference on Computer Vision*, 2019, pp. 2162–2171.

[63] R. Mahjourian, M. Wicke, and A. Angelova, "Unsupervised learning of depth and ego-motion from monocular video using 3d geometric constraints," in *IEEE Conference on Computer Vision Pattern Recognision*, 2018, pp. 5667–5675.

[64] Z. Yin and J. Shi, "Geonet: Unsupervised learning of dense depth, optical flow and camera pose," in *IEEE Conference on Computer Vision Pattern Recognision*, 2018, pp. 1983–1992.

[65] C. Wang, J. Miguel Buenaposada, R. Zhu, and S. Lucey, "Learning depth from monocular videos using direct methods," in *IEEE Conference on Computer Vision Pattern Recognision*, 2018, pp. 2022–2030.

[66] A. Ranjan, V. Jampani, L. Balles, K. Kim, D. Sun, J. Wulff, and M. J. Black, "Competitive Collaboration: Joint Unsupervised Learning of Depth, Camera Motion, Optical Flow and Motion Segmentation," in *IEEE Conference on Computer Vision Pattern Recognision*, 2019, pp. 12 240–12 249.

[67] F. Tosi, F. Aleotti, M. Poggi, and S. Mattoccia, "Learning monocular depth estimation infusing traditional stereo knowledge," in *IEEE Conference on Computer Vision Pattern Recognision*, 2019, pp. 9799–9809.

[68] J.-W. Bian, Z. Li, N. Wang, H. Zhan, C. Shen, M.-M. Cheng, and I. Reid, "Unsupervised Scale-consistent Depth and Ego-motion Learning from Monocular Video," in *Advances in Neural Information Processing Systems*, 2019, pp. 35–45.

[69] Y. Zhang, M. Gong, J. Li, M. Zhang, F. Jiang, and H. Zhao, "Self-supervised monocular depth estimation with multiscale perception," *IEEE Transactions on Image Processing*, vol. 31, pp. 3251–3266, 2022.

[70] X. Xu, Z. Chen, and F. Yin, "Multi-scale spatial attention-guided monocular depth estimation with semantic enhancement," *IEEE Transactions on Image Processing*, vol. 30, pp. 8811–8822, 2021.

[71] X. Ye, X. Fan, M. Zhang, R. Xu, and W. Zhong, "Unsupervised monocular depth estimation via recursive stereo distillation," *IEEE Transactions on Image Processing*, vol. 30, pp. 4492–4504, 2021.

[72] J. L. G. Bello, J. Moon, and M. Kim, "Self-supervised monocular depth estimation with positional shift depth variance and adaptive disparity quantization," *IEEE Transactions on Image Processing*, 2024.

[73] G. Li, R. Huang, H. Li, Z. You, and W. Chen, "Sense: Self-evolving learning for self-supervised monocular depth estimation," *IEEE Transactions on Image Processing*, 2023.

[74] T. Zhou, M. Brown, N. Snavely, and D. G. Lowe, "Unsupervised learning of depth and ego-motion from video," in *IEEE Conference on Computer Vision Pattern Recognision*, 2017, pp. 1851–1858.

[75] C. Godard, O. Mac Aodha, and G. J. Brostow, "Unsupervised monocular depth estimation with left-right consistency," in *IEEE Conference on Computer Vision Pattern Recognision*, 2017, pp. 270–279.

[76] C. Godard, O. Mac Aodha, M. Firman, and G. J. Brostow, "Digging into self-supervised monocular depth estimation," in *International Conference on Computer Vision*, 2019, pp. 3828–3838.

[77] V. Guizilini, I. Vasiljevic, R. Ambrus, G. Shakhnarovich, and A. Gaidon, "Full surround monodepth from multiple cameras," *IEEE Robotics and Automation Letters*, vol. 7, no. 2, pp. 5397–5404, 2022.

[78] Y. Wei, L. Zhao, W. Zheng, Z. Zhu, Y. Rao, G. Huang, J. Lu, and J. Zhou, "Surrounddepth: Entangling surrounding views for self-supervised multi-camera depth estimation," in *Conference on Robot Learning*. PMLR, 2023, pp. 539–549.

[79] A. Schmied, T. Fischer, M. Danelljan, M. Pollefeys, and F. Yu, "R3d3: Dense 3d reconstruction of dynamic scenes from multiple cameras," in *International Conference on Computer Vision*, 2023, pp. 3216–3226.

[80] J.-H. Kim, J. Hur, T. P. Nguyen, and S.-G. Jeong, "Self-supervised surround-view depth estimation with volumetric feature fusion," *Advances in Neural Information Processing Systems*, vol. 35, pp. 4032–4045, 2022.

[81] J. Xu, X. Liu, Y. Bai, J. Jiang, K. Wang, X. Chen, and X. Ji, "Multi-camera collaborative depth prediction via consistent structure estimation," in *ACM International Conference on Multimedia*, 2022, pp. 2730–2738.

[82] Y. Shi, H. Cai, A. Ansari, and F. Porikli, "Ega-depth: Efficient guided attention for self-supervised multi-camera depth estimation," in *IEEE Conference on Computer Vision Pattern Recognision Workshops*, 2023, pp. 119–129.
[83] A. W. Harley, Z. Fang, J. Li, R. Ambrus, and K. Fragkiadaki, "Simple-BEV: What really matters for multi-sensor bev perception?" in *IEEE International Conference on Robotics and Automation*, 2023.
[84] W. Gan, W. Wu, S. Chen, Y. Zhao, and P. K. Wong, "Rethinking 3d cost aggregation in stereo matching," *Pattern Recognition Letters*, vol. 167, pp. 75–81, 2023.
[85] N. Max, "Optical models for direct volume rendering," *IEEE Transactions on Visualization and Computer Graphics*, vol. 1, no. 2, pp. 99–108, 1995.
[86] Y. Li, S. Li, X. Liu, M. Gong, K. Li, N. Chen, Z. Wang, Z. Li, T. Jiang, F. Yu *et al.*, "Sscbench: A large-scale 3d semantic scene completion benchmark for autonomous driving," *arXiv preprint arXiv:2306.09001*, 2023.
[87] IDEA-Research, "Grounded segment anything," https://github.com/IDEA-Research/Grounded-Segment-Anything.
[88] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *IEEE Conference on Computer Vision Pattern Recognision*, 2016, pp. 770–778.
[89] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *IEEE Conference on Computer Vision Pattern Recognision*, 2009, pp. 248–255.
[90] L. Roldao, R. de Charette, and A. Verroust-Blondet, "Lmscnet: Lightweight multiscale 3d semantic completion," in *International Conference on 3D Vision*. IEEE, 2020, pp. 111–119.
[91] X. Chen, K.-Y. Lin, C. Qian, G. Zeng, and H. Li, "3d sketch-aware semantic scene completion via semi-supervised structure prior," in *IEEE Conference on Computer Vision Pattern Recognision*, 2020, pp. 4193–4202.
[92] J. Li, K. Han, P. Wang, Y. Liu, and X. Yuan, "Anisotropic convolutional networks for 3d semantic scene completion," in *IEEE Conference on Computer Vision Pattern Recognision*, 2020, pp. 3351–3359.