IEEE Robotics and Automation Letters (RAL) paper presented at
2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)
October 14-18, 2024. Abu Dhabi, UAE

Co-Occ: Coupling Explicit Feature Fusion with Volume Rendering Regularization for Multi-Modal 3D Semantic Occupancy Prediction

Jingyi Pan[1], Zipeng Wang[1], Lin Wang[1,2,*]

***Abstract*— 3D semantic occupancy prediction is a pivotal task in the field of autonomous driving. Recent approaches have made great advances in 3D semantic occupancy predictions on a single modality. However, multi-modal semantic occupancy prediction approaches have encountered difficulties in dealing with the modality heterogeneity, modality misalignment, and insufficient modality interactions that arise during the fusion of different modalities data, which may result in the loss of important geometric and semantic information. This letter presents a novel multi-modal, *i.e.*, LiDAR-camera 3D semantic occupancy prediction framework, dubbed Co-Occ, which couples explicit LiDAR-camera feature fusion with implicit volume rendering regularization. The key insight is that volume rendering in the feature space can proficiently bridge the gap between 3D LiDAR sweeps and 2D images while serving as a physical regularization to enhance LiDAR-camera fused volumetric representation. Specifically, we first propose a Geometric- and Semantic-aware Fusion (GSFusion) module to explicitly enhance LiDAR features by incorporating neighboring camera features through a K-nearest neighbors (KNN) search. Then, we employ volume rendering to project the fused feature back to the image planes for reconstructing color and depth maps. These maps are then supervised by input images from the camera and depth estimations derived from LiDAR, respectively. Extensive experiments on the popular nuScenes and SemanticKITTI benchmarks verify the effectiveness of our Co-Occ for 3D semantic occupancy prediction. The project page is available at https://rorisis.github.io/Co-Occ_project-page/.**

I. INTRODUCTION

3D semantic occupancy prediction is a task that involves estimating the geometric structure and semantic categories of occupied voxels in a scene simultaneously, which has been widely applied in robot manipulation [1], robot navigation [2] and autonomous driving [3], [4]. While earlier methods primarily focused on indoor environments and utilized dense geometric information from LiDAR or depth sensors, outdoor 3D semantic occupancy prediction relies on sparse point clouds and multi-view or monocular camera images. Besides, unlike other 3D perception tasks that predominantly focus on specific classes of foreground objects (*e.g.*, 3D object detection), 3D semantic occupancy prediction requires a comprehensive perception of the surroundings by predicting the geometry and semantics within the scene [5], [6].

Leveraging the complementary strengths of LiDAR and camera data is crucial in various 3D perception tasks.

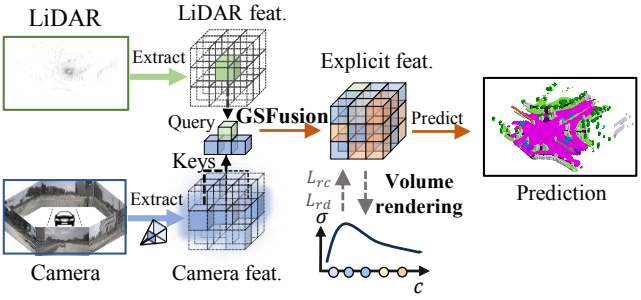

Fig. 1. **The pipeline of our Co-Occ**. Our method utilizes the GSFusion module to acquire explicit fused features that retain both the semantic benefits from the cameras and the geometric benefits from LiDAR. Then, implicit volume rendering-based regularization is applied to bridge the gap between 3D LiDAR and 2D images and enhance fused representation.

Cameras provide rich semantic information but lack precise geometric details, while LiDAR offers accurate depth or spatial information but may suffer from sparse contextual details [7]–[10]. Therefore, many 3D perception methods combine both modalities through camera-to-LiDAR or LiDAR-to-camera projection [9], [11], unification in the voxel space [8], [12] or in bird's eye view (BEV) space [7].

However, the fusion of LiDAR-camera data for 3D semantic occupancy prediction is not a straightforward task due to the heterogeneity between the modalities and the limited interaction between them. Specifically, LiDAR sweeps capture sparse 3D points, whereas cameras capture dense color information on the image plane. Most existing methods [8], [12] merge LiDAR and camera features by elevating 2D image features to the 3D voxel space in a pixel-to-point fashion. However, due to the extrinsic calibration inaccuracies between LiDAR and camera [13], the geometrically-aware 2D to 3D view transformation [14] might not efficiently elevate the 2D feature to the corresponding point in the 3D space. Moreover, the inaccurate fused volumetric representations lead to inconsistent occupancy predictions and loss of semantic information.

In this letter, we present **Co-Occ**, a novel multi-modal 3D semantic occupancy prediction framework. Co-Occ couples explicit 3D feature representation with implicit volume rendering-based regularization to strengthen inter-modal interaction and enhance the fused volumetric representation, as shown in Fig. 1. Firstly, we extract features from LiDAR and camera data and project them into a unified voxel space. We then fuse the features via a Geometric- and Semantic-aware Fusion (GSFusion) module, which enhances LiDAR features with neighboring camera features within the geometric-aligned voxel space. To accomplish this, we use a K-nearest neighbors (KNN) search to identify relevant

*Corresponding author
[1]J. PAN and Z. Wang are with the AI Thrust, The Hong Kong University of Science and Technology (Guangzhou), Guangdong 511458, China. {jpan305, zwang253}@connect.hkust-gz.edu.cn
[1,2]L. Wang is with AI/CMA Thrust, HKUST(GZ) and Dept. of CSE, HKUST, Hong Kong SAR, China, Email: linwang@ust.hk
This work was supported by the Guangzhou Fundamental and Applied Basic Research under Grant No.2024A04J4072.

camera features, which are further selected using a KNN gate operation. The GSFusion module explicitly incorporates the semantic information from camera features into LiDAR features, particularly for the sparse input.

Then, we incorporate implicit volume rendering-based regularization to supervise the fused explicit representations. Inspired by recent advancements in neural rendering [15], we cast rays from the camera into the scene and sample along these rays uniformly. The corresponding features of these samples are retrieved from the fused feature, and two auxiliary heads predict the density and color of these samples. The color and depth are then projected back onto the 2D image plane and supervised by ground truth of color from cameras and depth maps from LiDAR. *This enables us to effectively bridge the gap between 3D LiDAR sweeps and 2D camera images and enhance the fused volumetric representation*. We then employ the fused LiDAR-camera features for decoding and occupancy prediction. It's worth noting that the volume rendering regularization is only applied during training and does not impact the inference time.

We conducted extensive experiments on the nuScenes [5] and SemanticKITTI [6] benchmarks. The results demonstrate that our Co-Occ effectively boosts the accuracy and density of semantic occupancy prediction. We establish the new state-of-the-art, with **41.1%** IoU and **27.1%** mIoU on the nuScenes validation set with a voxel size of [0.5m, 0.5m, 0.5m] [5], [16]. Also, on the SemanticKITTI test set, we achieve **56.6%** IoU and **24.4%** mIoU. To sum up, our major contributions are three-fold: (**I**) We propose a novel multi-modal 3D semantic occupancy prediction framework as it can efficiently take LiDAR and camera inputs; (**II**) We propose the GSFusion module together with the implicit volume rendering-based regularization. This optimally leverages each modality while ensuring consistent, fine-grained unified volumetric representation; (**III**) Extensive experiments on the nuScenes datasets and the SemanticKITTI benchmark demonstrate the superiority of our method.

II. RELATED WORK

3D Semantic Occupancy Prediction. The objective of 3D semantic occupancy prediction is to estimate the surrounding scene by incorporating both geometry and semantics. Earlier works primarily focused on indoor scenarios [17]–[19]. In recent years, S3CNet [20], JS3C-Net [3], and SSA-SC [21] refine predictions using sparse point clouds with different representations in outdoor scenarios. MonoScene [22] is the first work using only RGB inputs. Recent advancements include SCPNet [23], OccFormer [24], and SurroundOcc [16], which use multi-path networks to aggregate multi-scale features for semantic occupancy prediction. Approaches like [25], [26] directly predict 3D semantic occupancy using NeRF [15], but rendering speed limits their efficiency. Openoccupancy [12] introduces a benchmark for LiDAR-Camera semantic occupancy prediction. In this letter, we enhance the performance of 3D semantic occupancy prediction through explicit feature fusion and implicit volume rendering regularization.

LiDAR-Camera Fusion-Based 3D Perception. LiDAR and camera sensors are widely used in various 3D perception tasks. Existing research has developed different LiDAR-camera fusion techniques for tasks like 3D object detection [7], [27], [28] and LiDAR segmentation [9], [29]. Fusion methods can be categorized as: **1)** Project-based fusion [9], [11], combining image features with raw LiDAR points or projecting LiDAR point clouds into a range view and fusing with image features. **2)** Feature-level fusion [7], [29], [30], projecting LiDAR and image data into a shared feature space like BEV or voxels. **3)** Attention-based fusion [27], [31], utilizing LiDAR features as proposals to query image features through cross-attention. In contrast to implicit fusion methods designed for specific objects or sparse 3D perception, our method initially leverages KNN-based fusion methods to expand the semantic perception field with aligned geometric representations of two modalities, reducing errors from inaccurate calibrations. Additionally, we utilize volume rendering to bridge the gap between 2D and 3D representations and enhance the volumetric fused features across different modalities.

Scene Understanding with Volume Rendering. Recently, 3D perception approaches inspired by Neural Radiance Field (NeRF) [15] have employed volume rendering to generate 3D scene representations or estimate 3D geometry information from multi-view images. Previous studies, such as [32], [33], have focused on generating implicit scene representations through per-scene optimization using multi-view images. However, this approach can pose challenges when it comes to generalizing the results to different scenes. Approaches like [25], [26], [34] utilize NeRF to directly obtain 3D semantic occupancy predictions or acquire the rendered 2D semantic to supervise the semantic information. Unlike previous methods, we propose an implicit volume rendering-based regularization to enforce regularization on the fusion of LiDAR and camera data, based on the RGB and depth, to further enhance the unified volumetric representation.

III. PROPOSED METHOD

Our objective is to predict the 3D occupancy of surrounding scenes using LiDAR sweeps, denoted as L, and its corresponding surround-view images, represented as $I = \{I_1, ..., I_N\}$, where N indicates the total number of camera views in a scene. Our Co-Occ framework comprises two key components: an explicit Geometric- and Semantic-Aware fusion module (GSFusion, Sec. III-A), and an implicit volume rendering -based regularization achieved in the feature space (Sec. III-B). We provide a detailed description of the optimization of our framework in Sec. III-C.

A. Geometric- and Semantic-aware Fusion

Directly projecting images into LiDAR space [8], [31], [35] may introduce errors due to inaccurate extrinsic calibration and depth prediction [13]. Also, a substantial amount of empty voxel space may impede the interaction between the two modalities, leading to inconsistent occupancy predictions

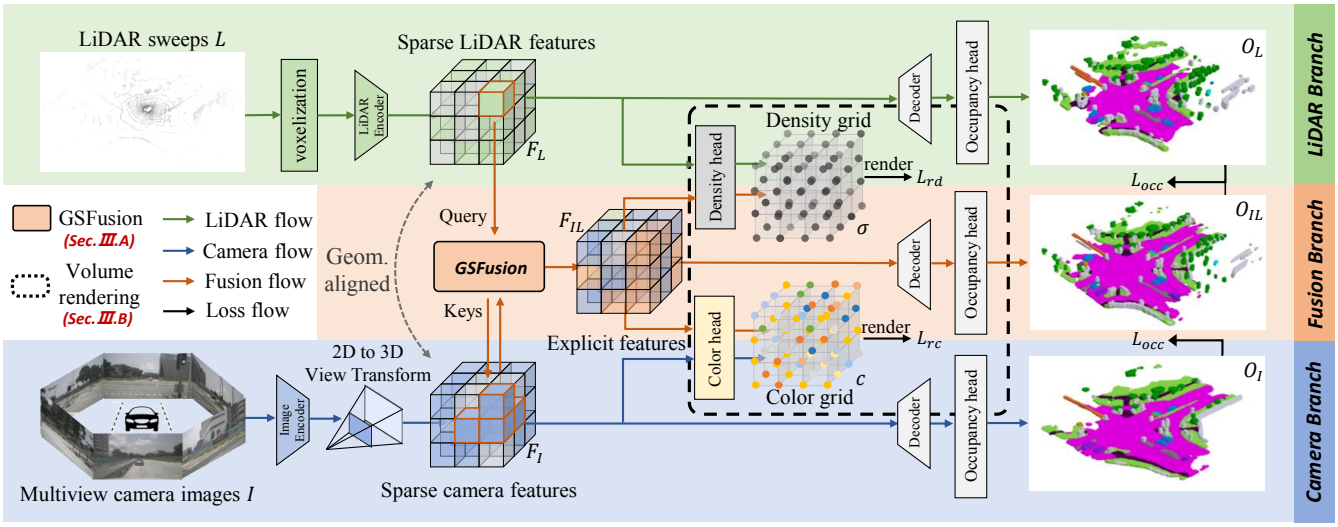

Fig. 2. **Our Co-Occ framework**. It consists of an explicit GSFusion module and implicit volume rendering regularization. The GSFusion module (Fig. 3) takes advantage of both the semantic benefits derived from camera features and the geometric benefits obtained from LiDAR. Meanwhile, the implicit volume rendering regularization (Fig. 4) guarantees the fusion of explicit LiDAR-camera features in an accurate and detailed manner, which further enhances the performance of 3D semantic prediction. Notably, implicit volume rendering regularization is only utilized during the training process.

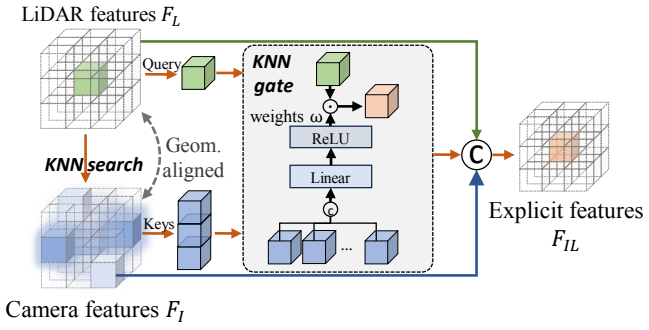

Fig. 3. The workflow of the GSFusion module begins with searching for K nearest neighbors from camera features to supplement the semantic context of LiDAR features. A KNN gate is then used to obtain weights to boost the LiDAR features. The final step involves concatenating the features.

and loss of semantic information. To overcome these challenges, we propose GSFusion, which incorporates a KNN search and KNN gate, as illustrated in Fig. 3.

KNN Search. We employ a 2D image encoder [36]–[38] to derive camera features, followed by a 2D-to-3D view transformation [12], [14] to project these 2D features into 3D sparse image features F_I as voxel representation. Simultaneously, we encode the LiDAR sweep into a 3D sparse feature F_L utilizing voxelization and 3D LiDAR encoder [39]. Consequently, the camera features F_I and LiDAR features F_L are aligned with the same dimension $\mathbb{R}^{D\times H\times W\times C}$, where C signifies the feature dimensions.

We then select the 3D coordinates of non-empty LiDAR and camera features, denoted as $P_L \in \mathbb{R}^{N_L\times 3}$ and $P_I \in \mathbb{R}^{N_I\times 3}$. Here, N_L and N_I represent the quantity of non-empty features, and each entry corresponds to a 3D coordinate (x, y, z). Due to F_I and F_L being geometrically aligned, we can directly search K nearest neighbors of a given LiDAR coordinate in the voxel space within a specific radius r. For the i-th non-empty LiDAR feature, we denote its neighboring color features as $\{N_i^0, N_i^1, ..., N_i^k\}$, where k is the hyper-parameter of the neighboring number. We implement a fast KNN search algorithm with CUDA kernel, which ensures the efficiency of KNN search.

KNN Gate. After obtaining the K nearest neighborhood coordinates from the camera features, we propose a learnable KNN gate (see Fig. 3) to obtain the semantic weight ω_i of each i-th non-empty LiDAR feature for further fusion with LiDAR features.

$$\omega_i = \text{Linear}(\text{Concat}(N_i^0, N_i^1, ..., N_i^k)), \tag{1}$$

And we derive the fused LiDAR-camera features F_{IL} as:

$$F_{IL} = \text{Concat}(F_I, F_L, F_L \cdot \omega), \tag{2}$$

where ω represents the total semantic weight of the LiDAR features from w_i.

B. Implicit Volume Rendering Regularization

After fusing LiDAR-camera features using the GSFusion module, we propose a volume rendering-based regularization method depicted in Fig. 4. Unlike existing scene understanding methods that focus on processing data solely from images or points using volume rendering techniques [26], [40], [41], our approach applies volume rendering in a generalized manner to regulate explicit features fused from LiDAR and camera data across different scenes, instead of a per-scene basis. This allows us to impose a physical constraint and promote consistency within the fused features.

Sampling along Rays in Feature Space. Following neural rendering techniques [15], we cast rays from the camera into the space and uniformly sample those rays, leading to 3D coordinates $X \in \mathbb{R}^{N\times n_s\times h\times w\times 3}$ in ego-car space. Here, n_s denotes the number of sampling points on each ray and h, w denotes the height and width of the rendered image size. To reduce the computational cost, we set h and w to be $1/16$ of the original image. We then use trilinear interpolation to generate a frustum feature F_I^s corresponding the sampled coordinates X from the fused feature F_{IL}:

$$F_I^s = \text{Interpolate}(F_{IL}, X) \tag{3}$$

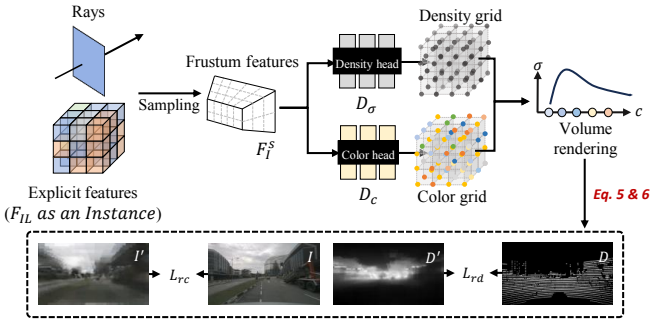

Fig. 4. The implicit volume rendering-based regularization involves obtaining frustum features from rays and explicit features. Frustum features are used to create the density grid and color grid, which are then utilized to generate the depth map and color map.

where $F_I^s \in \mathbb{R}^{N\times n_s\times h\times w\times C}$, and $\mathrm{Interpolate}(*)$ queries the feature F_{IL} at location X.

We employ two Multi-Layer Perceptron (MLP) networks, D_c and D_σ, to generate color grid c and density grid σ of points respectively:

$$\begin{aligned} c &= \mathrm{Sigmoid}(D_c(F_I^s)), \\ \sigma &= \mathrm{ReLU}(D_\sigma(F_I^s)) \end{aligned} \tag{4}$$

where $c \in \mathbb{R}^{N\times n_s\times h\times w\times 3}$ and $\sigma \in \mathbb{R}^{N\times n_s\times h\times w\times 1}$.

Volume Rendering for Depth and Color. We render the predicted color map $\hat{I}$ using volume rendering:

$$\hat{I} = \sum_{i=1}^{n_s} T_i(1-\exp(-\sigma_i\delta_{t_i}))c_i, \tag{5}$$

where $T_i = \exp(-\sum_{j=1}^{i-1}\sigma_j\delta_{t_j})$ and δ_t is the distance between two sampled points. Similarly, we obtain the depth map $\hat{D}$ from the density field:

$$\hat{D} = \sum_{i=1}^{n_s} T_i(1-\exp(-\sigma_i\delta_{t_i})). \tag{6}$$

We then upsample $\hat{I}$ and $\hat{D}$ to image resolution as $\{\hat{I}', \hat{D}'\} \in \{\mathbb{R}^{h'\times w'\times 3}, \mathbb{R}^{h'\times w'\times 1}\}$. Here, h' and w' represent the height and the width of the image resolution. Such rendered RGB images and depth are supervised by color loss, depth loss, and opacity loss with input camera and LiDAR in Sec. III-C.

C. Optimization

Based above of our methods above, there are four main terms in our loss function.

Occupancy Loss. we use cross-entropy loss $\mathcal{L}_{ce}$ and lovasz-softmax loss $\mathcal{L}_{ls}$ to supervise between predicted semantic occupancy O and ground truth occupancy $\hat{O}$ as:

$$\mathcal{L}_{occ} = \mathcal{L}_{ce} + \mathcal{L}_{ls}. \tag{7}$$

Explicit Depth Loss. In the 2D to 3D view transform, we will use a Depth Net [14] to learn the depth with $\mathcal{L}_d$.

Rendering Color Regularization. The color loss is calculated by MSE Loss between rendered color maps $\hat{I}'$ and the input camera multi-view images I as:

$$\mathcal{L}_{rc} = \lambda_{rc}\cdot\left\|\hat{I}'-I\right\|_2, \tag{8}$$

where λ_{color} is a hyperparameter of $\mathcal{L}_{color}$.

Rendering Depth Regularization. The ground truth depth maps are obtained by projecting paired LiDAR points onto images plane as D, and we use L1 loss to calculate $\mathcal{L}_{rd}$:

$$\mathcal{L}_{rd} = \lambda_{rd}\cdot\left\|\hat{D}'-D\right\|_1. \tag{9}$$

where λ_{rd} is a hyperparameter of $\mathcal{L}_{rd}$.

The overall objective function is:

$$\mathcal{L} = \mathcal{L}_{occ} + \mathcal{L}_d + \mathcal{L}_{rc} + \mathcal{L}_{rd}. \tag{10}$$

Discussion on different modal branch. In Fig. 2, our LiDAR-camera fusion branch utilizes GSFusion and associated losses. In the LiDAR branch, depth regularization enhances LiDAR feature consistency without using GSFusion. In the camera branch, both color and depth regularization are applied with ground truth depth input. Without ground truth depth, only color regularization is employed.

IV. EXPERIMENT RESULTS

A. Implementation Details and Dataset

Implementation Details. We evaluate 3D semantic occupancy performance on the nuScenes occupancy validation set [5], [16] and the SemanticKITTI semantic scene completion set [6]. For the camera branch, we use ResNet50 or ResNet101 [36] with FPN [38] for both datasets. The view transformer [14] generates a 3D feature volume of size $100\times100\times8$ and $128\times128\times16$, with 128 channels for nuScenes and SemanticKITTI, respectively. In the LiDAR branch, we voxelize 10 LiDAR sweeps and employ a voxel encoder for the nuScenes datasets. LiDAR features are augmented by selecting the 2 nearest neighborhoods from camera features. We use the same occupancy decoder and head as in [12] with a cascade ratio of 2 for refining predictions. Our implicit volume rendering regularization incorporates two MLP networks as the density head and color head. The color head consists of three MLP layers, while the density head consists of either one MLP layer or a linear layer. The model is trained for 24 epochs on the nuScenes dataset and 30 epochs on the SemanticKITTI dataset, using a batch size of 8 across 8 A40 GPUs. We employ the AdamW [45] optimizer with a weight decay of 0.01 and an initial learning rate of 1e-4, along with a multi-step learning rate scheduler.

NuScenes Dataset. nuScenes [5] is a large-scale autonomous driving benchmark including 1K driving scenarios, including 700 scenes for training, 150 scenes for validations, and 150 scenes for testing. The 3D occupancy ground truth provided by [16] of each sample with a voxel size of $[0.5m, 0.5m, 0.5m]$ and represents $[200, 200, 16]$ dense voxel grids, which have 17 classes (16 semantics, 1 free).

SemanticKITTI Dataset. The SemanticKITTI dataset [6] focuses on the semantic scene understanding with LiDAR point clouds and camera images. Specifically, the ground truth semantic occupancy is represented as the $[256, 256, 32]$ voxel grids. Each voxel is $[0.2m, 0.2m, 0.2m]$ large and annotated with 21 semantic classes (19 semantics, 1 free, 1 unknown). Following [20], [22], the 22 sequences are split into 10 sequences, 1 sequence, and 11 sequences for training, validation, and testing.

TABLE I

3D SEMANTIC OCCUPANCY PREDICTION RESULTS ON NUSCENES VALIDATION SET. WE REPORT THE GEOMETRIC METRIC IOU, SEMANTIC METRIC MIOU, AND THE IOU FOR EACH SEMANTIC CLASS. THE C AND L DENOTES CAMERA AND LIDAR, RESPECTIVELY.

Method	Modality	IoU	mIoU	barrier	bicycle	bus	car	const. veh.	motorcycle	pedestrian	traffic cone	trailer	truck	drive. suf.	other flat	sidewalk	terrain	manmade	vegetation	Input Size	2D Backbone
MonoScene [22]	C	24.0	7.3	4.0	0.4	8.0	8.0	2.9	0.3	1.2	0.7	4.0	4.4	27.7	5.2	15.1	11.3	9.0	14.9	900 × 1600	R101-DCN
BEVFormer [42]	C	30.5	16.7	14.2	6.5	23.4	28.2	8.6	10.7	6.4	4.0	11.2	17.7	37.2	18.0	22.8	22.1	13.8	22.2	900 × 1600	R101-DCN
SurroundOcc [16]	C	31.4	20.3	20.5	11.6	28.1	30.8	10.7	15.1	14.0	12.0	14.3	22.2	37.2	23.7	24.4	22.7	14.8	21.8	900 × 1600	R101-DCN
OccFormer [24]	C	29.9	20.1	21.1	11.3	28.2	30.3	10.6	15.7	14.4	11.2	14.0	22.6	37.3	22.4	24.9	23.5	15.2	21.1	896 × 1600	R101
C-CONet [12]	C	26.1	18.4	18.6	10.0	26.4	27.4	8.6	15.7	13.3	9.7	10.9	20.2	33.0	20.7	21.4	21.8	14.7	21.3	896 × 1600	R101
FB-Occ [43]	C	31.5	19.6	20.6	11.3	26.9	29.8	10.4	13.6	13.7	11.4	11.5	20.6	38.2	21.5	24.6	22.7	14.8	21.6	896 × 1600	R101
RenderOcc [26]	C	29.2	19.0	19.7	11.2	28.1	28.2	9.8	14.7	11.8	11.9	13.1	20.1	33.2	21.3	22.6	22.3	15.3	20.9	896 × 1600	R101
LMSCNet [44]	L	36.6	14.9	13.1	4.5	14.7	22.1	12.6	4.2	7.2	7.1	12.2	11.5	26.3	14.3	21.1	15.2	18.5	34.2	-	-
L-CONet [12]	L	39.4	17.7	19.2	4.0	15.1	26.9	6.2	3.8	6.8	6.0	14.1	13.1	39.7	19.1	24.0	23.9	25.1	35.7	-	-
M-CONet [12]	C&L	39.2	24.7	24.8	13.0	31.6	34.8	14.6	18.0	20.0	14.7	20.0	26.6	39.2	22.8	26.1	26.0	26.0	37.1	896 × 1600	R101
Co-Occ (Ours)	C	30.0	20.3	22.5	11.2	28.6	29.5	9.9	15.8	13.5	8.7	13.6	22.2	34.9	23.1	24.2	24.1	18.0	24.8	896 × 1600	R101
Co-Occ (Ours)	L	**42.2**	22.9	22.0	6.90	25.7	32.4	14.5	13.5	**21.0**	10.5	18.0	22.5	36.6	21.8	24.6	25.7	**31.2**	**39.9**	-	-
Co-Occ (Ours)	C&L	41.1	**27.1**	**28.1**	**16.1**	**34.0**	**37.2**	**17.0**	**21.6**	20.8	**15.9**	**21.9**	**28.7**	**42.3**	**25.4**	**29.1**	**28.6**	28.2	38.0	896 × 1600	R101

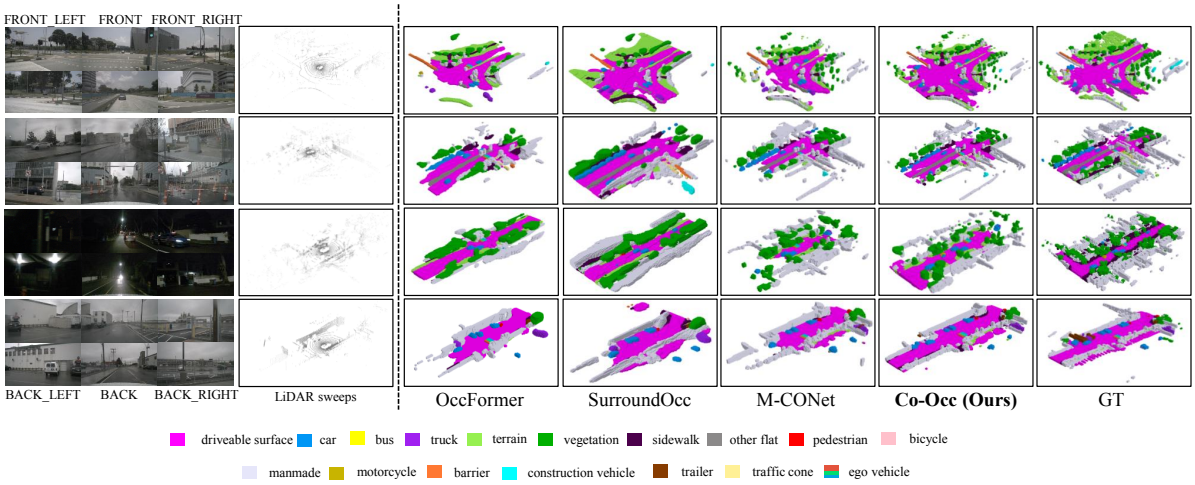

Fig. 5. The qualitative comparisons results on nuScenes validation set. The leftmost column shows the input surrounding images and LiDAR sweeps, the following three columns visualize the 3D semantic occupancy prediction from OccFormer [24], SurroundOcc [16] (these two predicts results using only cameras), M-CONet [12], our Co-Occ, and the annotation from [16]. **Better viewed when zoomed in.**

TABLE II

ABLATION STUDY ON THE IMPACT OF DIFFERENT COMPONENTS.

Base	Fusion	Rendering		IoU	mIoU
	GSFusion	$\mathcal{L}_{rc}$	$\mathcal{L}_{rd}$		
✓				38.2	24.2
✓	✓			40.1	25.9
✓	✓	✓		40.8	26.4
✓	✓	✓	✓	**41.1**	**27.1**

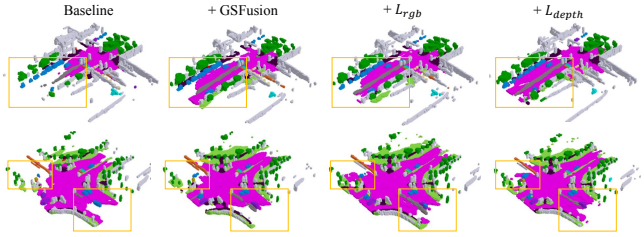

Fig. 6. The visualization of ablation study on the impact of different components in nuScenes validation set.

B. Main Results

To ensure fair comparisons, all results are either implemented by their authors or reproduced using official codes. **Results on NuScenes Dataset:** We first conduct experiments on the nuScenes [5] dataset and compare with several SOTA methods with different modalities, *i.e.*, camera-only methods [12], [16], [22], [24], [26], [42], [43], LiDAR-only methods [12], [44], and LiDAR-camera fusion methods [12]. The settings for MonoScene [22], BEVFormer [42], and SurroundOcc [16] involve an input size of 900×1600 and utilize a ResNet101-DCN backbone, consistent with the test settings used in the previous work [16]. As for OccFormer [24], C-CONet [12], M-CONet [12], FB-Occ [43], and RenderOcc [26], we combine the multi-camera or LiDAR-camera occupancy predictions and voxelize them with a voxel size of $0.5m$, matching our setting where the input image size is 896×1600 and the 2D backbone is ResNet101. Tab. I demonstrates that our Co-Occ with LiDAR and the camera achieves a notable increase of **2.4%** mIoU compared to M-CONet [12], which also utilizes both camera and LiDAR data, thus showcasing the effectiveness of our fusion techniques. Besides, our methods utilizing the camera-only branch show improvements in both mIoU and IoU with the same input size and 2D backbone methods [12], [24], [26], [43]. Additionally, our Co-Occ with the LiDAR-only branch also demonstrates significant improvement [12], [44]. It is important to note that the IoU score of LiDAR-only is better than the LiDAR-camera method, suggesting that the LiDAR

TABLE III

3D SEMANTIC OCCUPANCY PREDICTION RESULTS ON SEMANTICKITTI TEST SET. THE C AND L DENOTE CAMERA AND LIDAR.

Method	Modality	mIoU	road (15.30%)	sidewalk (11.13%)	parking (1.12%)	other-grnd (0.56%)	building (14.1%)	car (3.92%)	truck (0.16%)	bicycle (0.03%)	motorcycle (0.03%)	other-veh. (0.20%)	vegetation (39.3%)	trunk (0.51%)	terrain (9.17%)	person (0.07%)	bicyclist (0.07%)	motorcyclist. (0.05%)	fence (3.90%)	pole (0.29%)	traf.-sign (0.08%)
MonoScene [22]	C	11.1	54.7	27.1	24.8	5.7	14.4	18.8	3.3	0.5	0.7	4.4	14.9	2.4	19.5	1.0	1.4	0.4	11.1	3.3	2.1
SurroundOcc [16]	C	11.9	56.9	28.3	30.2	6.8	15.2	20.6	1.4	1.6	1.2	4.4	14.9	3.4	19.3	1.4	2.0	0.1	11.3	3.9	2.4
OccFormer [24]	C	12.3	55.9	30.3	31.5	6.5	15.7	21.6	1.2	1.5	1.7	3.2	16.8	3.9	21.3	2.2	1.1	0.2	11.9	3.8	3.7
RenderOcc [26]	C	12.8	57.2	28.4	16.1	0.9	18.2	24.9	6.0	3.1	0.28	3.6	26.2	4.8	3.6	1.9	3.3	0.3	9.1	6.2	3.3
LMSCNet [44]	L	17.0	64.0	33.1	24.9	3.2	38.7	29.5	2.5	0.0	0.0	0.1	40.5	19.0	30.8	0.0	0.0	0.0	20.5	15.7	0.5
JS3C-Net [3]	L	23.8	64.0	39.0	34.2	**14.7**	39.4	33.2	**7.2**	**14.0**	**8.1**	**12.2**	43.5	19.3	39.8	**7.9**	**5.2**	0.0	30.1	17.9	15.1
SSC-RS [4]	L	24.2	73.1	**44.4**	38.6	17.4	**44.6**	36.4	5.3	10.1	5.1	11.2	**44.1**	26.0	41.9	4.7	2.4	0.9	30.8	15.0	7.2
M-CONet [12]	C&L	20.4	60.6	36.1	29.0	13.0	38.4	33.8	4.7	3.0	2.2	5.9	41.5	20.5	35.1	0.8	2.3	**0.6**	26.0	18.7	15.7
Co-Occ (Ours)	C&L	**24.4**	**72.0**	43.5	**42.5**	10.2	35.1	**40.0**	6.4	4.4	3.3	8.8	41.2	**30.8**	**40.8**	1.6	3.3	0.4	**32.7**	**26.6**	**20.7**

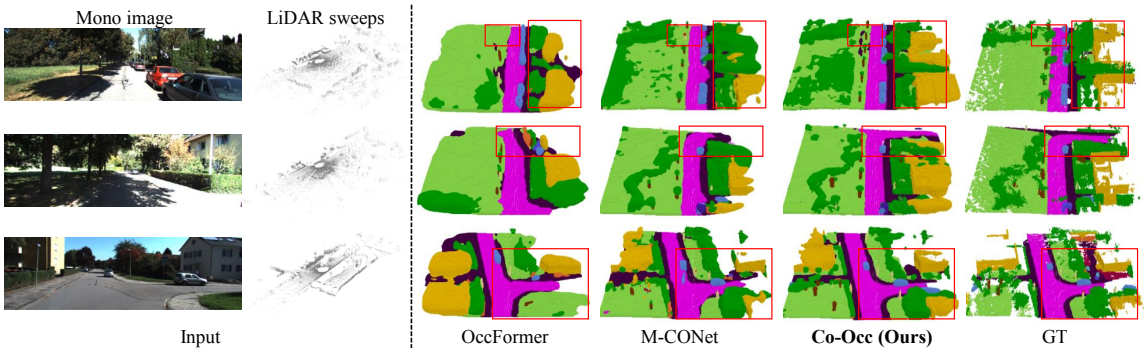

Fig. 7. The qualitative comparisons results on SemanticKITTI validation set. The input monocular image and LiDAR sweeps are shown on the left and the 3D semantic occupancy results from OccFormer [24] (OccFormer predicts results using only mono image), M-CONet [12], our Co-Occ, and the annotations are then visualized sequentially. **Better viewed when zoomed in.**

sweeps more regions but with less accurate categorization for the semantic classes in the context of LiDAR-camera fusion methods. Furthermore, as shown in Fig. 5, the qualitative results illustrate that our methods predict a more fine-grained and accurate 3D semantic occupancy.

Results on SemanticKITTI Dataset: To further validate the effectiveness of our techniques, we conducted a comparative analysis with single-modal and multi-modal state-of-the-art methods [3], [12], [16], [22], [24], [44] on the SemanticKITTI test set [6]. As shown in Tab. III, our methods outperform JS3CNet [3] by 0.6% mIoU and SSC-RS [4] by 0.2% mIoU, even though they utilize additional LiDAR segmentation supervision. Furthermore, our methods show a significant improvement of **4.0%** mIoU over M-CONet [12]. Fig. 7 depicts the qualitative results on SemanticKITTI validation, indicating our superior performance in both scene occupancy complementary and accuracy of object details. These results further confirm the effectiveness of our techniques.

C. Ablation Studies

Ablation on Architectural Components. As depicted in Tab. II and Fig. 6, our GSFusion module exhibits a 1.7% mIoU improvement compared to our baseline, which employs the concatenation fusion between LiDAR and camera features. Furthermore, incorporating the implicit volume rendering regularization through the inclusion of $\mathcal{L}_{rc}$ and $\mathcal{L}_{rd}$ losses leads to additional performance enhancement in our method. This showcases the effectiveness of each module and the loss in our approach.

TABLE IV

ABLATION STUDY ON FUSION STRATEGY AND PARAMETER SELECTION.

Fusion Method	k	n_s	IoU↑	mIoU↑
GSFusion	1	56	39.9	25.9
	1	112	40.5	26.3
	2	112	<u>41.1</u>	<u>27.1</u>
	3	112	**41.3**	**27.2**
Concatenated	2	112	38.2	24.5
Weighted [12]	2	112	39.3	25.6

Ablation on Different Fusion Strategies. In Tab. IV, we conduct an ablation study on fusion strategies, comparing concatenated fusion and weighted fusion, *e.g.*, [12], used in previous LiDAR-camera fusion-based works. We find that straight concatenated fusion methods underperform compared to other fusion techniques. This may be due to limited utilization of the inaccurate calibration of the two modalities. In contrast, our GSFusion strategy leverages both geometric and semantic information, leading to a significant 1.5% improvement over weighted fusion. We analyze that although weighted fusion methods use a network to adaptively learn the fusion weights of camera and LiDAR features, inaccurate calibrations among LiDAR and camera features may still lead to errors in the weighted fusion process.

Ablation on Parameter Selection. We perform an internal ablation study on our two components, focusing on the

TABLE V

EFFICIENCY ANALYSIS ON CO-OCC ON A SINGLE RTX A40 GPU.

Image Size	2D backbone	IoU	mIoU	Memory (G)	Latency (s)
256×704	R50	40.9	25.0	10.75	0.45
896×1600	R50	**41.4**	26.7	11.65	0.55
896×1600	R101	41.1	**27.1**	11.78	0.58

TABLE VI

3D SEMANTIC OCCUPANCY RESULTS WITH DIFFERENT RANGES.

Method	IoU			mIoU		
	25m	50m	100m	25m	50m	100m
M-CONet [12]	60.9	51.0	39.2	36.9	31.8	24.7
Co-Occ (ours)	62.7	53.0	41.1	40.3	34.6	27.1
Improvements (%)	+1.8	+2.0	+2.0	+3.4	+2.8	+2.4

selection of the numbers for the KNN strategies and the sampling points.

1) The numbers of neighbors k: As shown in Tab. IV, we notice a correlation between performance and the number of selected neighbors. Selecting two nearest neighbors improves performance by approximately 0.8% mIoU to select one nearest neighbor. However, selecting three neighbors does not provide an obvious enhancement and adds a computational burden. Thus, we choose $k = 2$ as our optimal number.

2) The numbers of sampling points n_s: We conduct this study using the same depth bound and half depth bound as the sampling point parameters as shown in Tab. IV. After careful evaluation, we determined that setting the sampling parameter to 112 yields the best performance.

Analysis of model efficiency. Tab. V evaluates the inference time and memory usage for different image sizes and 2D backbones in our LiDAR-camera fusion-based branch. These experiments were conducted on a single RTX A40 GPU. A model with low image resolution and ResNet50 exhibits reduced computational costs and latency, but its performance is limited. Increasing the image size and depth of ResNet does not significantly increase memory usage or latency, but it leads to a significant improvement in performance.

Analysis of results within different ranges. We further propose to evaluate different ranges surrounding the car comprehensively. We present the IoU and mIoU data separately for the volumes of $25m \times 25m \times 8m$, $50m \times 50m \times 8m$, and $100m \times 100m \times 8m$. Understanding short-range areas is crucial as it allows less time for autonomous vehicles to react. As depicted in Tab. VI, our method shows substantial improvements (3.4%) over M-CONet [12] in short-range areas. Due to the sparsity of LiDAR sweeps in long-range areas, only a few pixels determine the depth of a large area. Despite the diminishing improvements of our methods over M-CONet [12], it still maintains a 2.4% improvement in mIoU scores, demonstrating the effectiveness of our methods across different ranges.

V. CONCLUSION

In this letter, we proposed a multi-modal 3D semantic occupancy prediction framework that combines explicit GSFusion and implicit volume rendering regularization. Our approach improves the interaction between LiDAR and camera data and enables dense semantic occupancy predictions. We introduced a KNN-based GSFusion method for optimal feature fusion in the voxel space. Besides, we incorporated implicit volume rendering regularization to enhance the fused representation by connecting 2D images and 3D LiDAR sweeps. Extensive experiments confirmed the effectiveness of our proposed method.

REFERENCES

[1] J. Varley, C. DeChant, A. Richardson, J. Ruales, and P. Allen, "Shape completion enabled robotic grasping," in *2017 IEEE/RSJ international conference on intelligent robots and systems (IROS)*. IEEE, 2017, pp. 2442–2447.

[2] L. Wang, H. Ye, Q. Wang, Y. Gao, C. Xu, and F. Gao, "Learning-based 3d occupancy prediction for autonomous navigation in occluded environments," in *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2021, pp. 4509–4516.

[3] X. Yan, J. Gao, J. Li, R. Zhang, Z. Li, R. Huang, and S. Cui, "Sparse single sweep lidar point cloud segmentation via learning contextual shape priors from scene completion," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 35, no. 4, 2021, pp. 3101–3109.

[4] J. Mei, Y. Yang, M. Wang, T. Huang, X. Yang, and Y. Liu, "Ssc-rs: Elevate lidar semantic scene completion with representation separation and bev fusion," in *2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2023, pp. 1–8.

[5] 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 *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 11 621–11 631.

[6] 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 *Proceedings of the IEEE/CVF international conference on computer vision*, 2019, pp. 9297–9307.

[7] Z. Liu, H. Tang, A. Amini, X. Yang, H. Mao, D. L. Rus, and S. Han, "Bevfusion: Multi-task multi-sensor fusion with unified bird's-eye view representation," in *2023 IEEE international conference on robotics and automation (ICRA)*. IEEE, 2023, pp. 2774–2781.

[8] Y. Li, Y. Chen, X. Qi, Z. Li, J. Sun, and J. Jia, "Unifying voxel-based representation with transformer for 3d object detection," *Advances in Neural Information Processing Systems*, vol. 35, pp. 18 442–18 455, 2022.

[9] J. Li, H. Dai, H. Han, and Y. Ding, "Mseg3d: Multi-modal 3d semantic segmentation for autonomous driving," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 21 694–21 704.

[10] Y. Qin, C. Wang, Z. Kang, N. Ma, Z. Li, and R. Zhang, "Supfusion: Supervised lidar-camera fusion for 3d object detection," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023, pp. 22 014–22 024.

[11] T. Huang, Z. Liu, X. Chen, and X. Bai, "Epnet: Enhancing point features with image semantics for 3d object detection," in *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XV 16*. Springer, 2020, pp. 35–52.

[12] 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 *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023, pp. 17 850–17 859.

[13] S. Chen, J. Liu, X. Liang, S. Zhang, J. Hyyppä, and R. Chen, "A novel calibration method between a camera and a 3d lidar with infrared images," in *2020 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2020, pp. 4963–4969.

[14] J. Philion and S. Fidler, "Lift, splat, shoot: Encoding images from arbitrary camera rigs by implicitly unprojecting to 3d," in *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XIV 16*. Springer, 2020, pp. 194–210.

[15] 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," *Communications of the ACM*, vol. 65, no. 1, pp. 99–106, 2021.

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

[17] S. Song, F. Yu, A. Zeng, A. X. Chang, M. Savva, and T. Funkhouser, "Semantic scene completion from a single depth image," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 1746–1754.

[18] J. Zhang, H. Zhao, A. Yao, Y. Chen, L. Zhang, and H. Liao, "Efficient semantic scene completion network with spatial group convolution," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 733–749.

[19] S. Liu, Y. Hu, Y. Zeng, Q. Tang, B. Jin, Y. Han, and X. Li, "See and think: Disentangling semantic scene completion," *Advances in Neural Information Processing Systems*, vol. 31, pp. 263–274, 2018.

[20] R. Cheng, C. Agia, Y. Ren, X. Li, and L. Bingbing, "S3cnet: A sparse semantic scene completion network for lidar point clouds," in *Conference on Robot Learning*. PMLR, 2021, pp. 2148–2161.

[21] X. Yang, H. Zou, X. Kong, T. Huang, Y. Liu, W. Li, F. Wen, and H. Zhang, "Semantic segmentation-assisted scene completion for lidar point clouds," in *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2021, pp. 3555–3562.

[22] A.-Q. Cao and R. de Charette, "Monoscene: Monocular 3d semantic scene completion," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 3991–4001.

[23] Z. Xia, Y. Liu, X. Li, X. Zhu, Y. Ma, Y. Li, Y. Hou, and Y. Qiao, "Scpnet: Semantic scene completion on point cloud," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 17 642–17 651.

[24] Y. Zhang, Z. Zhu, and D. Du, "Occformer: Dual-path transformer for vision-based 3d semantic occupancy prediction," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023, pp. 9433–9443.

[25] M. Pan, L. Liu, J. Liu, P. Huang, L. Wang, S. Zhang, S. Xu, Z. Lai, and K. Yang, "Uniocc: Unifying vision-centric 3d occupancy prediction with geometric and semantic rendering," *arXiv preprint arXiv:2306.09117*, 2023.

[26] 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," *arXiv preprint arXiv:2309.09502*, 2023.

[27] Y. Li, A. W. Yu, T. Meng, B. Caine, J. Ngiam, D. Peng, J. Shen, Y. Lu, D. Zhou, Q. V. Le, *et al.*, "Deepfusion: Lidar-camera deep fusion for multi-modal 3d object detection," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 17 182–17 191.

[28] Z. Zhou and S. Tulsiani, "Sparsefusion: Distilling view-conditioned diffusion for 3d reconstruction," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 12 588–12 597.

[29] Z. Zhang, Z. Zhang, Q. Yu, R. Yi, Y. Xie, and L. Ma, "Lidar-camera panoptic segmentation via geometry-consistent and semantic-aware alignment," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023, pp. 3662–3671.

[30] Z. Wang and K. Jia, "Frustum convnet: Sliding frustums to aggregate local point-wise features for amodal 3d object detection," in *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2019, pp. 1742–1749.

[31] X. Bai, Z. Hu, X. Zhu, Q. Huang, Y. Chen, H. Fu, and C.-L. Tai, "Transfusion: Robust lidar-camera fusion for 3d object detection with transformers," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2022, pp. 1090–1099.

[32] A. Kundu, K. Genova, X. Yin, A. Fathi, C. Pantofaru, L. J. Guibas, A. Tagliasacchi, F. Dellaert, and T. Funkhouser, "Panoptic neural fields: A semantic object-aware neural scene representation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 12 871–12 881.

[33] Z. Wu, T. Liu, L. Luo, Z. Zhong, J. Chen, H. Xiao, C. Hou, H. Lou, Y. Chen, R. Yang, *et al.*, "Mars: An instance-aware, modular and realistic simulator for autonomous driving," in *CAAI International Conference on Artificial Intelligence*. Springer, 2023, pp. 3–15.

[34] C. Zhang, J. Yan, Y. Wei, J. Li, L. Liu, Y. Tang, Y. Duan, and J. Lu, "Occnerf: Self-supervised multi-camera occupancy prediction with neural radiance fields," *arXiv preprint arXiv:2312.09243*, 2023.

[35] Z. Song, H. Wei, L. Bai, L. Yang, and C. Jia, "Graphalign: Enhancing accurate feature alignment by graph matching for multi-modal 3d object detection," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023, pp. 3358–3369.

[36] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[37] Z. Liu, Y. Lin, Y. Cao, H. Hu, Y. Wei, Z. Zhang, S. Lin, and B. Guo, "Swin transformer: Hierarchical vision transformer using shifted windows," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2021, pp. 10 012–10 022.

[38] T.-Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie, "Feature pyramid networks for object detection," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 2117–2125.

[39] Y. Yan, Y. Mao, and B. Li, "Second: Sparsely embedded convolutional detection," *Sensors*, vol. 18, no. 10, p. 3337, 2018.

[40] Y. Siddiqui, L. Porzi, S. R. Bulò, N. Müller, M. Nießner, A. Dai, and P. Kontschieder, "Panoptic lifting for 3d scene understanding with neural fields," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 9043–9052.

[41] J. Xu, L. Peng, H. Cheng, H. Li, W. Qian, K. Li, W. Wang, and D. Cai, "Mononerd: Nerf-like representations for monocular 3d object detection," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023, pp. 6814–6824.

[42] Z. Li, W. Wang, H. Li, E. Xie, C. Sima, T. Lu, Y. Qiao, and J. Dai, "Bevformer: Learning bird's-eye-view representation from multi-camera images via spatiotemporal transformers," in *European conference on computer vision*. Springer, 2022, pp. 1–18.

[43] 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.

[44] L. Roldao, R. de Charette, and A. Verroust-Blondet, "Lmscnet: Lightweight multiscale 3d semantic completion," in *2020 International Conference on 3D Vision (3DV)*. IEEE, 2020, pp. 111–119.

[45] I. Loshchilov and F. Hutter, "Decoupled weight decay regularization," in *Proceedings of the 7th International Conference on Learning Representations*, 2019, pp. 1–18.