GaussianFormer3D: Multi-Modal Gaussian-based Semantic Occupancy Prediction with 3D Deformable Attention

Lingjun Zhao, Sizhe Wei, James Hays and Lu Gan

***Abstract*— 3D semantic occupancy prediction is essential for achieving safe, reliable autonomous driving and robotic navigation. Compared to camera-only perception systems, multi-modal pipelines, especially LiDAR-camera fusion methods, can produce more accurate and fine-grained predictions. Although voxel-based scene representations are widely used for semantic occupancy prediction, 3D Gaussians have emerged as a continuous and significantly more compact alternative. In this work, we propose a multi-modal Gaussian-based semantic occupancy prediction framework utilizing 3D deformable attention, namely GaussianFormer3D. We introduce a voxel-to-Gaussian initialization strategy that provides 3D Gaussians with accurate geometry priors from LiDAR data, and design a LiDAR-guided 3D deformable attention mechanism to refine these Gaussians using LiDAR-camera fusion features in a lifted 3D space. Extensive experiments on real-world on-road and off-road autonomous driving datasets demonstrate that GaussianFormer3D achieves state-of-the-art prediction performance with reduced memory consumption and improved efficiency. Project website: https://lunarlab-gatech.github.io/GaussianFormer3D/.**

I. Introduction

Perception systems are essential for the development of safe, reliable and intelligent autonomous vehicles and field robots [1]. Among various perception tasks, 3D semantic occupancy prediction is particularly crucial as it enables fine-grained understanding of both geometric and semantic information of the environments [2]. For autonomous driving (AD), recent advances in vision-based occupancy prediction have shown impressive results on large-scale datasets [3], [4]. However, the sensitivity of cameras to lighting variations and their limited depth accuracy still underscore the need to incorporate additional sensing modalities for robust AD.

LiDAR sensors have been widely applied to AD for perception tasks such as 3D object detection [5], [6]. Compared to cameras, LiDAR provides more accurate depth information and finer geometric relationships of objects, making it particularly advantageous for 3D semantic occupancy prediction [7]–[12]. However, LiDAR-based pipelines often struggle to capture accurate semantics for small objects, where camera-based methods excel [13]. To balance geometric accuracy and semantic richness, multi-modal fusion algorithms have been proposed to leverage the strengths of complementary sensing modalities. Current approaches include mainly LiDAR-camera fusion [14]–[17] and camera–radar fusion [18], with LiDAR-camera fusion showing superior performance and becoming the dominant choice.

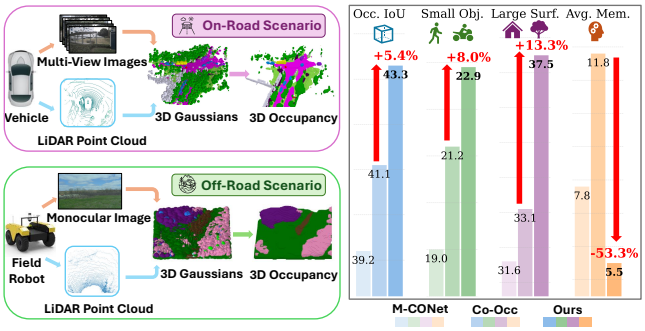


Fig. 1: **We propose a new LiDAR-camera fusion-based semantic occupancy prediction framework using 3D Gaussians.** We evaluate it on both on-road and off-road driving scenarios. Our method demonstrates superior performance on overall occupancy Intersection-of-Union (IoU), achieves substantial performance gains on small objects (*pedestrian*, *motorcycle*) and large surfaces (*man-made*, *vegetation*), and consumes less memory during inference.

Most LiDAR-camera occupancy networks employ a 3D voxel-based [14]–[17], [19] representation to model a scene as a dense grid-based structure. Despite achieving comparable performance, these methods inevitably suffer from redundant empty grids and high computational costs. Recently, inspired by the success of 3D Gaussian splatting [20], an object-centric Gaussian representation has been explored in vision-based 3D semantic occupancy prediction, achieving improved computational efficiency. GaussianFormer series [21], [22] represent a scene as a set of 3D Gaussians, each consisting of a mean, covariance and semantics. These Gaussians are refined using a 2D deformable attention [23], and then processed by an efficient Gaussian-to-voxel splatting module to predict semantic occupancy. Despite high efficiency, current Gaussian-based methods [21], [22] rely solely on 2D image to update 3D Gaussians, limiting their ability to model 3D space with accurate depth and fine-grained geometric structure. How to effectively leverage other sensor modalities, such as LiDAR, to refine and obtain a more accurate 3D Gaussian representation for efficient semantic occupancy prediction remains unexplored and challenging.

In this work, we propose **GaussianFormer3D**: a multi-modal Gaussian-based semantic occupancy prediction framework with 3D deformable attention, as shown in Fig. 1. GaussianFormer3D models a scene using 3D Gaussians initialized from LiDAR voxel features, updates Gaussians through 3D deformable attention in a LiDAR-camera unified 3D feature space, and predicts semantic occupancy via Gaussian-to-voxel splatting. To the best of our knowledge, our model is

The authors are with the Georgia Institute of Technology, Atlanta, GA 30332, USA. L. Zhao is supported by IRIM Ph.D. Fellowship at Georgia Institute of Technology. Email: {lzhao360,swei,hays,lgan}@gatech.edu.

the first multi-modal semantic occupancy network that employs an object-centric Gaussian-based scene representation. In summary, our main contributions are as follows:

- We propose a novel multi-modal Gaussian-based semantic occupancy prediction framework. By integrating LiDAR and camera data, ours significantly outperforms camera-only baselines with similar memory usage.
- We design a voxel-to-Gaussian initialization module to provide 3D Gaussians with geometry priors from LiDAR, and also develop an enhanced 3D deformable attention mechanism to update Gaussians by aggregating LiDAR-camera fusion features in a lifted 3D space.
- We conduct extensive evaluations on two on-road datasets, nuScenes-SurroundOcc [24] and nuScenes-Occ3D [25], along with an off-road dataset, RELLIS3D-WildOcc [26]. Results show that ours outperforms state-of-the-art dense grid-based methods while achieving reduced memory consumption and improved efficiency.

II. Related Work

Multi-Modal Semantic Occupancy Prediction. Multi-modal occupancy prediction methods generally outperform single-modal approaches, as different modalities provide complementary information. Among them, LiDAR-camera fusion is the top-performing configuration, combining LiDAR's accurate depth and geometry sensing with the powerful semantic understanding capability of cameras. Similar to single-modal pipelines, most LiDAR-camera occupancy prediction networks are also voxel-based [14]–[17], [27]. CONet [27] proposes a coarse-to-fine pipeline to sample 3D voxel features for refining the coarse occupancy prediction. Co-Occ [14] obtains multi-modal voxel features through a geometric and semantic-aware fusion module, and employs a NeRF-based implicit volume rendering regularization [28] to enhance the fused representation. OccGen [15] and OccMamba [16] encode multi-modal inputs to produce voxel fusion features, and then decode the features using diffusion denoising and hierarchical Mamba modules, respectively. OccFusion [17] transforms LiDAR and camera inputs into multi-modal voxel features via 2D deformable attention [23].

3D Gaussians for Autonomous Driving. Due to the inherent advantages of modeling scenes explicitly and continuously, 3D Gaussians [20] have been adopted as the scene representation over the traditional grid-based solutions in 3D semantic occupancy prediction [21], [22], [29]. 3D Gaussians also demonstrated their superiority in real-time image rendering and novel view synthesis, and thus have been adopted for driving scene reconstruction and simulation [30]–[32]. Furthermore, end-to-end autonomous driving [33] and visual pre-training [34] utilize 3D Gaussians as the driving world representation for various downstream perception and planning tasks. However, these approaches are mainly designed for camera-only autonomous driving, neglecting the potential of multi-modal data in Gaussian initialization and updating. GSPR [35] proposes a Gaussian-based multi-modal place recognition algorithm, and SplatAD [32] designs the first 3D Gaussian splatting pipeline to render both LiDAR and camera data. In this work, we explore utilizing multi-modal data, especially from LiDAR and camera sensors, to learn a fine-grained 3D Gaussian representation for more accurate and efficient semantic occupancy prediction.

III. Method

The overview of GaussianFormer3D is presented in Fig. 2.

A. Scene as 3D Gaussian Representation

Semantic occupancy prediction aims to jointly predict the semantic information and geometric structure of the scene. Given multi-view images $\mathcal{I} = \{\mathbf{I}_i\}_{i=1}^{N_c}$ and LiDAR point cloud $\mathcal{P} = \{\mathbf{P}_i\}_{i=1}^{N_p}$, $\mathbf{P}_i = (x_i, y_i, z_i, \eta_i)$ containing the 3D location and intensity of each point, the goal is to predict the semantic occupancy grid $\mathbf{O} \in \mathcal{C}^{X \times Y \times Z}$, where N_c, N_p, $\mathcal{C}$ denote the number of camera views, the number of LiDAR points, and the set of semantic classes, and $X \times Y \times Z$ is the size of the voxel grid to be predicted. Unlike uniform grids in traditional grid-based representations, 3D Gaussians can adaptively represent the regions of interest due to the universal approximation capability of Gaussian mixtures [21]. Specifically, a scene is modeled as a set of 3D Gaussians $\mathcal{G} = \{\mathbf{G}_i\}_{i=1}^{N_g}$, where N_g is the total number of Gaussians in a scene. Each Gaussian $\mathbf{G}_i$ is parameterized by its mean $\mathbf{m}_i \in \mathbb{R}^3$, rotation $\mathbf{r}_i \in \mathbb{R}^4$, scale $\mathbf{s}_i \in \mathbb{R}^3$, opacity $\sigma_i \in [0, 1]$ and semantic label $\mathbf{c}_i \in \mathbb{R}^{|\mathcal{C}|}$. The value of Gaussian $\mathbf{G}$ evaluated at location $\mathbf{x}$ can be calculated as:

$$\mathbf{g}(\mathbf{x}; \mathbf{G}) = \sigma \cdot \exp\big(-\frac{1}{2}(\mathbf{x} - \mathbf{m})^{\mathrm{T}} \mathbf{\Sigma}^{-1} (\mathbf{x} - \mathbf{m}) \big) \mathbf{c}, \quad (1)$$

$$\mathbf{\Sigma} = \mathbf{R}\mathbf{S}\mathbf{S}^{\mathrm{T}}\mathbf{R}^{\mathrm{T}}, \quad \mathbf{S} = \mathrm{diag}(\mathbf{s}), \quad \mathbf{R} = \mathrm{q2r}(\mathbf{r}), \quad (2)$$

where $\mathbf{\Sigma}$, $\mathbf{R}$ and $\mathbf{S}$ denote the covariance matrix, rotation matrix and scale matrix. $\mathrm{diag}(\cdot)$ is the diagonal matrix construction and $\mathrm{q2r}(\cdot)$ is the quaternion-to-rotation transformation. By summing the contributions of all Gaussians at location $\mathbf{x}$, the occupancy prediction can be formulated as:

$$\hat{\mathbf{o}}(\mathbf{x}; \mathcal{G}) = \sum_{i=1}^{N_g} \mathbf{g}_i(\mathbf{x}; \mathbf{m}_i, \mathbf{s}_i, \mathbf{r}_i, \sigma_i, \mathbf{c}_i). \quad (3)$$

The Gaussian-to-voxel splatting module is designed to only aggregate Gaussians within the neighborhood of a targeted voxel instead of querying all Gaussians in a scene to improve efficiency and reduce unnecessary computation and storage [21]. Thus, Eq. (3) can be furthur approximated by replacing N_g with $N_g(\mathbf{x})$, where $N_g(\mathbf{x})$ is the number of neighboring Gaussians at location $\mathbf{x}$. During training, the Gaussian-based occupancy model is trained in an end-to-end manner, supervised by the ground truth semantic occupancy label $\bar{\mathbf{O}} \in \mathcal{C}^{X \times Y \times Z}$. Both cross entropy loss L_{ce} and the lovasz-softmax loss L_{lov} are used for optimization.

B. Voxel-to-Gaussian Initialization

Two sets of 3D Gaussian features are adopted following GaussianFormer [21]. The first set consists of learnable Gaussian physical properties $\mathcal{G} = \{\mathbf{G}_i \in \mathbb{R}^d\}_{i=1}^{N_g}$ introduced in Sec. III-A, where $d = 11 + |\mathcal{C}|$, which are also our learning targets. The second set is non-learnable high-dimensional

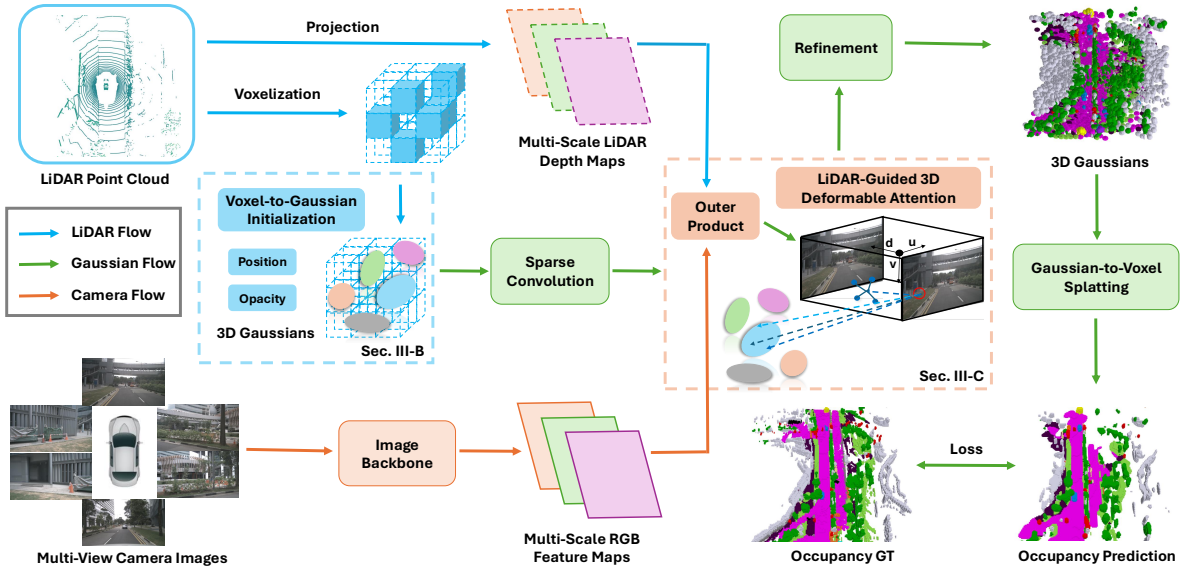


Fig. 2: **GaussianFormer3D Overview.** We first voxelize LiDAR point clouds to obtain non-empty voxel features for initializing the position and opacity of 3D Gaussians. Then LiDAR depth maps and camera feature maps are extracted respectively, and multiplied via outer product to construct a lifted 3D feature space. Gaussians are iteratively updated with sparse convolution, 3D deformable attention, and property refinement. Gaussians are eventually processed by a Gaussian-to-voxel splatting module to generate semantic occupancy.

Gaussian features $\mathcal{Q} = \{\mathbf{Q}_i \in \mathbb{R}^m\}_{i=1}^{N_g}$, where m is the feature dimension, serving as queries for the attention mechanism [36] and implicitly encoding the spatial and semantic information during the Gaussian update. Previous work [21] randomly initializes the Gaussian physical properties, and optimizes these properties iteratively through multiple refinement modules. This design constrains Gaussians to learn complex 3D geometry information solely from 2D images, which inevitably encounters inaccurate spatial modeling.

To resolve this issue, we propose a LiDAR-based voxel-to-Gaussian initialization strategy to initialize the mean and opacity of Gaussians with geometry priors from LiDAR data, as indicated in the dashed blue box in Fig. 2. Specifically, we first aggregate the most recent N_f LiDAR scans into a combined point cloud $\bar{\mathcal{P}} = \{\mathcal{P}_i\}_{i=1}^{N_f}$. Then we voxelize the combined point cloud and compute the feature of each non-empty voxel as the mean position and intensity of all points within it. These LiDAR-based voxel features are then used to initialize the position and opacity of 3D Gaussians:

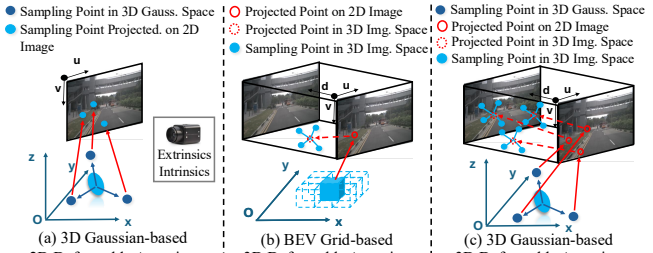
$$\mathbf{m}_i = \frac{1}{|\mathcal{P}_v|} \sum_{j \in \mathcal{P}_v} (x_j, y_j, z_j), \quad \sigma_i = \frac{1}{|\mathcal{P}_v|} \sum_{j \in \mathcal{P}_v} \eta_j, \tag{4}$$

where $i \in \{1, ..., N_g\}$ denotes the index of Gaussians to be initialized and $v \in \{1, ..., N_v\}$ denotes the index of all non-empty voxels; $\mathcal{P}_v$ is the set of LiDAR points in $\bar{\mathcal{P}}$ falling into voxel v. When $N_g < N_v$, we randomly choose a subset of N_g non-empty voxels to initialize Gaussians, otherwise, a subset of N_v Gaussians are randomly selected and initialized with non-empty voxels. After initialization, we apply a 3D sparse convolution module to the initialized 3D Gaussians for self-encoding. The features and interactions of Gaussians are efficiently extracted and aggregated through the sparse convolution for updating the Gaussian queries.

C. LiDAR-Guided 3D Deformable Attention

Lift, Splat, Shoot (LSS) [37] and 2D attention-based methods [23] are widely adopted for feature lifting which transform multi-view 2D images into a 3D space to obtain lifted features. However, LSS suffers from excessive computational costs, hindering its application to multi-scale feature maps that are important for recognizing objects of various sizes. GaussianFormer [21] utilizes a 2D deformable attention (Fig. 3(a)) to extract visual information from 2D images. Despite its efficiency, it suffers from the depth ambiguity problem. As multiple 3D reference points from different Gaussians can be projected to the same 2D position with similar sampling points in the 2D view, this leads to ineffective aggregation, i.e., aggregating the same 2D features for different 3D Gaussian queries. The underlying reason for this is the lack of accurate depth information during the feature lifting and aggregating. A 3D deformable attention operator, namely DFA3D [38], is designed to mitigate the depth ambiguity problem by first expanding 2D feature maps into 3D using estimated depth [39] and then applying an attention mechanism [36] to aggregate features from the expanded 3D feature maps. However, the operator is originally designed for BEV-based 3D object detection (Fig. 3(b)), and relies on DepthNet [39] to estimate monocular depth. Inspired by DFA3D [38], we propose a LiDAR-guided 3D deformable attention mechanism for Gaussian-based semantic occupancy prediction, as illustrated in the dashed orange box in Fig. 2. We first form a unified LiDAR-camera 3D feature space $\mathbf{F}^{\text{3D}}$ by conducting outer product between the multi-scale depth maps $\mathbf{F}^{\text{d}}$, generated from the LiDAR point cloud, and the multi-scale camera feature maps $\mathbf{F}^{\text{c}}$: $\mathbf{F}^{\text{3D}} = \mathbf{F}^{\text{d}} \otimes \mathbf{F}^{\text{c}}$. For feature sampling, we design a two-stage key point sampling method (Fig. 3(c)) to aggregate sufficient informative features for updating Gaussian queries. First, we sample a group of 3D reference points

$\mathcal{R}_G = \{\mathbf{m}_i = \mathbf{m} + \Delta\mathbf{m}_i | i = 1, ..., N_{R_1}\}$ for each Gaussian $\mathbf{G}$ by shifting its mean $\mathbf{m}$ with learned offsets $\Delta\mathbf{m}$. Then we project these 3D reference points into the fusion feature space $\mathbf{F}^{3D}$ with extrinsics $\mathcal{T}$ and intrinsics $\mathcal{K}$, where each projected reference point is positioned at $\bar{\mathbf{m}}_i = (u_i, v_i, d_i)$. After



(a) 3D Gaussian-based 2D Deformable Attention (b) BEV Grid-based 3D Deformable Attention (c) 3D Gaussian-based 3D Deformable Attention

Fig. 3: Comparison of different feature sampling methods.

projection, we further generate learnable sampling offsets $\Delta\bar{\mathbf{m}}_{ij} = (\Delta u_{ij}, \Delta v_{ij}, \Delta d_{ij})$ for each projected reference point $\bar{\mathbf{m}}_i$. The overall sampling points of a given Gaussian $\mathbf{G}$ in the fusion feature space $\mathbf{F}^{3D}$ can be formulated as:

$$\bar{\mathcal{R}}_G = \{\bar{\mathbf{m}}_{ij} = \bar{\mathbf{m}}_i + \Delta\bar{\mathbf{m}}_{ij} | i = 1, ..., N_{R_1}, j = 1, ..., N_{R_2}\}, \tag{5}$$

where N_{R_1} and N_{R_2} denote the number of sampling points for each Gaussian and for each projected 3D reference point. Finally, the Gaussian query $\mathbf{Q}$ is updated with the weighted sum of aggregated LiDAR-camera fusion features $\Delta\mathbf{Q}$:

$$\Delta\mathbf{Q} = \frac{1}{N_c}\sum_{c=1}^{N_c}\sum_{i=1}^{N_{R_1}}\sum_{j=1}^{N_{R_2}} \mathrm{DFA}(\mathbf{Q}, \boldsymbol{\pi}_c(\bar{\mathbf{m}}_{ij}; \mathcal{T}, \mathcal{K}), \mathbf{F}_c^{3D}), \tag{6}$$

where $\mathrm{DFA}(\cdot)$ and $\boldsymbol{\pi}_c(\cdot)$ represent the 3D deformable attention and the transformation from the Gaussian frame to $\mathbf{F}_c^{3D}$ frame generated from camera view c, respectively. After acquiring sufficient geometric and semantic information through sparse convolution and 3D deformable attention, the Gaussian query $\mathbf{Q}$ is passed to a multi-layer perceptron, and decoded to refine the Gaussian property $\mathbf{G}$. We iteratively optimize the properties with 4 blocks of sparse convolution, 3D deformable attention, and refinement modules.

IV. Experiments

A. Datasets

NuScenes [3] dataset provides 1000 sequences of driving scenes collected with 6 surrounding cameras, 1 LiDAR and 5 radars. Each sequence lasts 20 seconds and is annotated at a frequency of 2Hz. **SurroundOcc** [24] and **Occ3D** [25] both provide semantic occupancy annotation for nuScenes dataset, each including 700 and 150 scenes for training and validation respectively, for 18 classes (i.e., 16 semantics, 1 noise class and 1 empty). Differently, SurroundOcc partitions each scene within the range of $[-50\mathrm{m}, 50\mathrm{m}] \times [-50\mathrm{m}, 50\mathrm{m}] \times [-5\mathrm{m}, 3\mathrm{m}]$ into voxels with a resolution of $0.5\mathrm{m}$, whereas Occ3D divides a scene within $[-40\mathrm{m}, 40\mathrm{m}] \times [-40\mathrm{m}, 40\mathrm{m}] \times [-1\mathrm{m}, 5.4\mathrm{m}]$ into voxels with a resolution of $0.4\mathrm{m}$. A camera visibility mask is also provided in Occ3D.

RELLIS-3D [48] dataset is a multi-modal off-road driving dataset collected by a Clearpath Warthog robot containing RGB images, LiDAR point clouds, stereo images, GPS and IMU data. **WildOcc** [26] provides the first off-road occupancy annotation on the RELLIS-3D, which are split into 7399/1249/1399 frames for training, validation and testing respectively. The annotation is in the range of $[-20\mathrm{m}, 0\mathrm{m}] \times [-10\mathrm{m}, 10\mathrm{m}] \times [-2\mathrm{m}, 6\mathrm{m}]$, where each voxel has a resolution of $0.2\mathrm{m}$ and labeled as one of 9 classes (7 semantics, 1 other class and 1 empty). WildOcc [26] is used to evaluate the performance of our model in complex off-road environments and with a monocular-LiDAR sensor configuration.

B. Implementation and Evaluation Details

For camera branch, we set the resolution of input images as 900×1600 for nuScenes [3] and 1200×1920 for RELLIS-3D [48]. We utilize the ResNet101-DCN [49] checkpoint pretrained from FCOS3D [50] as the backbone and FPN [51] as the neck. For LiDAR branch, we aggregate and voxelize previous 10 sweeps of point clouds, and obtain the mean features through a voxel feature encoder [5]. The LiDAR depth map is generated and saved before training following [38], [52]. The number of Gaussians is set to 25,600 in our main experiments. We employ these Gaussians to only model the occupied space, and leave the empty space to one fixed large Gaussian to improve efficiency [22]. We train our model with an AdamW optimizer with a weight decay of 0.01. The learning rates are set to 1×10^{-4} for nuScenes and 3×10^{-4} for RELLIS-3D, and decay with a cosine annealing schedule. Our model is trained for 24 epochs with a batch size of 8 on nuScenes and 20 epochs with a batch size of 4 on RELLIS-3D on Nvidia A40 GPUs. We use Intersection-over-Union (IoU) and mean Intersection-over-Union (mIoU) for evaluation metrics following MonoScene [40].

C. Quantitative Results

3D semantic occupancy prediction performance. We report the performance of GaussianFormer3D on SurroundOcc [24], Occ3D [25] and WildOcc [26] in Tab. I, Tab. II and Tab. III, respectively. For on-road scenarios in Tab. I and Tab. II, our method surpasses GaussianFormer [21] extensively on all classes, leading to overall 13.5 and 8.0 increases on the IoU and mIoU respectively on SurroundOcc [24] and 10.9 increase on the mIoU on Occ3D [25]. Compared to state-of-the-art LiDAR-camera approaches [14], [17], [27], ours achieves best overall performance while showing superior performance in predicting small objects (e.g., *motorcycle*, *pedestrian*), dynamic vehicles (e.g., *car*, *construction vehicle*, *truck*) and surrounding surfaces (e.g., *manmade*, *vegetation*) which are crucial classes for autonomous driving tasks. This improvement is due to Gaussians' universal approximating ability to model objects with flexible scales and shapes. For off-road results in Tab. III, our method with single-frame image input surpasses M-OFFOcc [26] using 4 sequential images by 1.1 in IoU and performs on par in mIoU. Moreover, our method outperforms GaussianFormer [21] by 14.4 in IoU and 6.8 in mIoU on the test set, highlighting LiDAR's role in understanding the geometry of complex off-road terrains. Our method excels in predicting regions with large surfaces, such as *grass*,

TABLE I: 3D semantic occupancy prediction results on nuScenes-SurroundOcc [24] validation set.

Method	Modality	IoU ↑	mIoU ↑	barrier	bicycle	bus	car	const. veh.	motorcycle	pedestrian	traffic cone	trailer	truck	drive. surf.	other flat	sidewalk	terrain	manmade	vegetation
MonoScene [40]	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
BEVFormer [41]	C	30.5	16.8	14.2	6.6	23.5	28.3	8.7	10.8	6.6	4.1	11.2	17.8	37.3	18.0	22.9	22.2	13.8	22.2
TPVFormer [42]	C	30.9	17.1	16.0	5.3	23.9	27.3	9.8	8.7	7.1	5.2	11.0	19.2	38.9	21.3	24.3	23.2	11.7	20.8
OccFormer [43]	C	31.4	19.0	18.7	10.4	23.9	30.3	10.3	14.2	13.6	10.1	12.5	20.8	38.8	19.8	24.2	22.2	13.5	21.4
SurroundOcc [24]	C	31.5	20.3	20.6	11.7	28.1	30.9	10.7	15.1	14.1	12.1	14.4	22.3	37.3	23.7	24.5	22.8	14.9	21.9
C-CONet [27]	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
FB-Occ [44]	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
GaussianFormer [21]	C	29.8	19.1	19.5	11.3	26.1	29.8	10.5	13.8	12.6	8.7	12.7	21.6	39.6	23.3	24.5	23.0	9.6	19.1
GaussianFormer-2 [22]	C	31.7	20.8	21.4	13.4	28.5	30.8	10.9	15.8	13.6	10.5	14.0	22.9	40.6	24.4	26.1	24.3	13.8	22.0
LMSCNet [7]	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 [27]	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 [27]	L+C	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
Co-Occ [14]	L+C	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
Ours	L+C	**43.3**	**27.1**	26.9	15.8	32.7	36.1	18.6	21.7	24.1	13.0	21.3	29.0	40.6	23.7	27.3	28.2	32.6	42.3

TABLE II: 3D semantic occupancy prediction results on nuScenes-Occ3D [25] validation set. * denotes training with camera visibility mask. (xf) denotes the number of history image frames used for temporal fusion.

Method	Modality	mIoU ↑	others	barrier	bicycle	bus	car	const. veh.	motorcycle	pedestrian	traffic cone	trailer	truck	drive. surf.	other flat	sidewalk	terrain	manmade	vegetation
MonoScene [40]	C	6.1	1.8	7.2	4.3	4.9	9.4	5.7	4.0	3.0	5.9	4.5	7.2	14.9	6.3	7.9	7.4	1.0	7.7
BEVFormer [41]	C	23.7	5.0	38.8	10.0	34.4	41.1	13.2	16.5	18.2	17.8	18.7	27.7	49.0	27.7	29.1	25.4	15.4	14.5
TPVFormer [42]	C	28.3	6.7	39.2	14.2	41.5	47.0	19.2	22.6	17.9	14.5	30.2	35.5	56.2	33.7	35.7	31.6	20.0	16.1
CTF-Occ [25]	C	28.5	8.1	39.3	20.6	38.3	42.2	16.9	24.5	22.7	21.1	23.0	31.1	53.3	33.8	38.0	33.2	20.8	18.0
RenderOcc [45]	C	26.1	4.8	31.7	10.7	27.7	26.5	13.9	18.2	17.7	17.8	21.2	23.3	63.2	36.4	46.2	44.3	19.6	20.7
GaussianFormer* [21]	C	35.5	8.8	40.9	23.3	42.9	49.7	19.2	24.8	24.4	22.5	29.4	35.3	79.0	36.9	46.6	48.2	38.8	33.1
COTR* (2f) [46]	C	44.5	13.3	52.1	32.0	46.0	55.6	32.6	32.8	30.4	34.1	37.7	41.8	84.5	46.2	57.6	60.7	52.0	46.3
PanoOcc* (4f) [47]	C	42.1	11.7	50.5	29.6	49.4	55.5	23.3	33.3	30.6	31.0	34.4	42.6	83.3	44.2	54.4	56.0	45.9	40.4
FB-Occ* (16f) [44]	C	42.1	14.3	49.7	30.0	46.6	51.5	29.3	29.1	29.4	30.5	35.0	39.4	83.1	47.2	55.6	59.9	44.9	39.6
OccFusion* [17]	L+C	**48.7**	12.4	51.8	33.0	54.6	57.7	34.0	43.0	48.4	35.5	41.2	48.6	83.0	44.7	57.1	60.0	62.5	61.3
Ours*	L+C	46.4	9.8	50.0	31.3	54.0	59.4	28.1	36.2	46.2	26.7	40.2	49.7	79.1	37.3	49.0	55.0	69.1	67.6

TABLE III: 3D semantic occupancy prediction results on RELLIS3D-WildOcc [26] dataset.

Method	Mod.	IoU ↑	mIoU ↑	Grass	Tree	Bush	Puddle	Mud	Barrie	Rubble
Test Set	Class Percentage %			41.052	36.094	17.621	0.512	0.774	0.001	0.001
C-OFFOcc (4f) [26]	C	29.7	11.2	24.6	23.8	22.1	0.6	3.5	0.6	3.2
GaussianFormer [21]	C	19.5	6.3	21.8	12.1	5.2	2.7	2.3	0.0	0.0
M-OFFOcc [26]	L+C	-	12.9	-	-	-	-	-	-	-
M-OFFOcc (4f) [26]	L+C	32.8	**14.8**	28.6	33.4	27.5	0.9	6.8	1.7	4.6
Ours	L+C	**33.9**	13.1	24.0	45.4	12.9	6.6	2.8	0.0	0.0
Validation Set	Class Percentage %			31.739	42.210	18.497	0.105	0.842	2.218	3.836
GaussianFormer [21]	C	23.0	8.2	19.4	24.4	5.2	0.0	4.4	0.0	4.0
Ours	L+C	**29.5**	**13.1**	19.1	38.5	10.6	0.1	4.6	4.2	14.5

TABLE IV: Efficiency comparison and ablation on number of Gaussians. Tested on one A40 GPU with one batch during inference.

Method	Mod.	Query Form	Query Number	Lat. (ms) ↓	Mem. (GB) ↓	IoU ↑	mIoU ↑
BEVFormer [41]	C	2D BEV	200×200	310	4.5	30.5	16.8
TPVFormer [42]	C	3D TPV	200×(200+16+16)	320	5.1	30.9	17.1
SurroundOcc [24]	C	3D Voxel	200×200×16	340	5.9	**31.5**	**20.3**
GaussianFormer [21]	C	3D Gaussian	25600	**227**	4.7	28.7	16.0
			144000	370	6.1	29.8	19.1
GaussianFormer-2 [22]	C	3D Gaussian	6400	313	**3.0**	30.4	19.9
			12800	323	**3.0**	30.4	19.9
			25600	357	**3.0**	31.0	**20.3**
M-CONet [27]	L+C	3D Voxel	50×50×4	532	7.6	33.3	21.2
			100×100×8	670	7.8	39.2	24.7
Co-Occ [14]	L+C	3D Voxel	100×100×8	580	11.8	41.1	**27.1**
Ours	L+C	3D Gaussian	6400	**415**	**4.9**	39.6	21.4
			12800	462	5.0	41.4	24.2
			25600	555	5.5	**43.3**	**27.1**

tree, and *puddle*, while remaining suboptimal for subtle terrain variations like *mud*. For *barrier* and *rubble*, their low occurrence in the test set (0.001% of occupied voxels) poses a challenge due to the lack of sufficient features for reliable prediction. We further evaluate the model performance under different weather conditions in Tab. V. Ours shows a significant performance improvement over the baseline under extreme climate (rainy) and low lighting condition (night).

Evaluation of model efficiency. We evaluate and compare the latency and average memory consumption of ours with other methods during testing in Tab. IV. Ours achieves multi-modal fusion-based prediction performance while maintaining approximately the same low memory usage as camera-only methods. Compared to Co-Occ [14], ours saves about 50% average memory consumption, making it more suitable for running onboard on autonomous vehicles. In addition, our approach employs only 25,600 Gaussians with 28 channels while Co-Occ [14] requires 80,000 queries with 128 channels to achieve similar performance, demonstrating the potential of our method to enable more efficient communication for connected vehicles or multi-robot collaborations. The latency of our method is higher than that of camera-only pipelines, which is mainly due to the computation overhead introduced by 3D deformable attention. Some LiDAR-camera methods [17], [26] are not compared due to lack of open-source code. We also examine the effect of the number of Gaussians on the model performance in Tab. IV. As the number of Gaussians increases, both latency and memory consumption rise, while the IoU and mIoU metrics are steadily improved.

D. Ablation Study

We conduct extensive ablation experiments to validate our design choices. The main ablation study is conducted in Tab. VI. We observe that both the proposed voxel-to-Gaussian initialization and the LiDAR-guided 3D deformable attention modules contribute to the superior performance of our method. The voxel-to-Gaussian initialization signif-

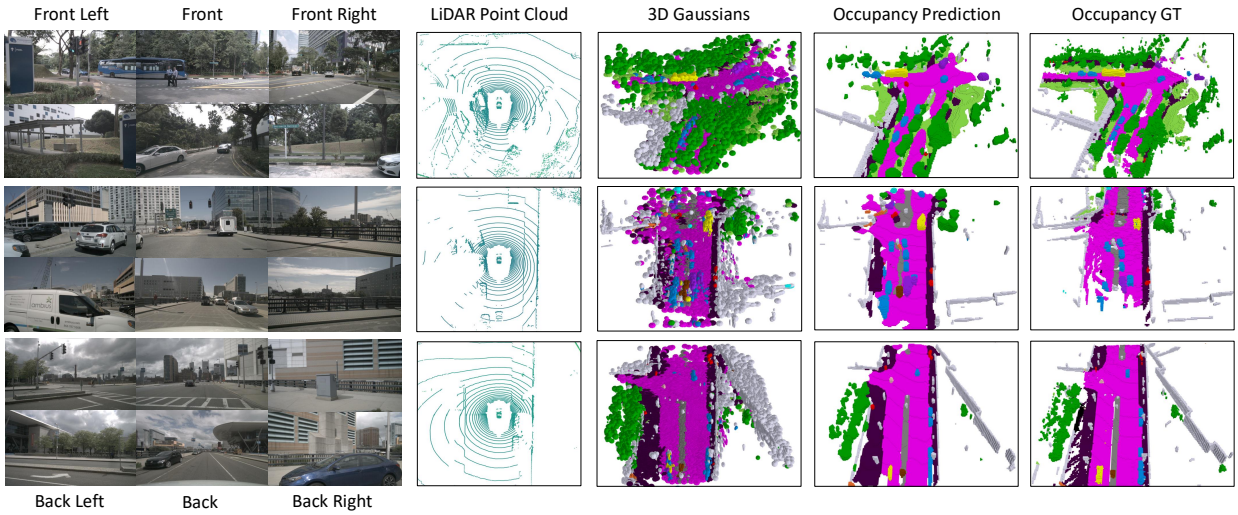


Fig. 4: Qualitative results on the on-road nuScenes-SurroundOcc [24] validation set. Our multi-modal Gaussian-based occupancy method can capture both semantics information and geometry structure of the surroundings. Color legend is given in Tab. I.

TABLE V: Performance on nuScenes-SurroundOcc [24] validation set under different weather and lighting conditions.

Method	Modality	IoU↑				mIoU↑			
		Sunny	Rainy	Day	Night	Sunny	Rainy	Day	Night
GaussianFormer [21]	C	29.6	27.5	30.3	19.5	18.9	18.0	19.2	9.3
GaussianFormer3D	L+C	**43.6 (+14.0)**	**41.6 (+14.1)**	**43.6 (+13.3)**	**40.5 (+21.0)**	**27.3 (+8.4)**	**25.2 (+7.2)**	**27.4 (+8.2)**	**15.5 (+6.2)**

TABLE VI: Ablation study of proposed modules evaluated on nuScenes-SurroundOcc [24] validation set. Voxel-to-Gaussian and LiDAR-Guided 3D Deformable Attention are abbreviated as V2G and DFA respectively.

Model	V2G	DFA	IoU ↑	mIoU ↑	barrier	bicycle	bus	car	const. veh.	motorcycle	pedestrian	traffic cone	trailer	truck	drive. surf.	other flat	sidewalk	terrain	manmade	vegetation
GaussianFormer3D			29.2	18.8	18.8	11.6	24.6	29.4	10.2	14.8	12.3	8.6	11.9	21.1	39.5	23.6	24.3	22.4	9.5	18.8
	✓		40.7	25.8	25.3	17.1	30.9	35.0	17.9	21.5	23.9	14.8	20.3	27.7	37.8	20.4	24.9	25.3	30.1	39.4
		✓	40.7	26.4	25.3	17.4	32.4	35.7	17.8	23.9	22.1	12.0	20.5	29.1	41.8	24.6	28.1	27.7	27.5	36.6
	✓	✓	43.3	27.1	26.9	15.8	32.7	36.1	18.6	21.7	24.1	13.0	21.3	29.0	40.6	23.7	27.3	28.2	32.6	42.3

TABLE VII: Ablation study of module design choices on the nuScenes-SurroundOcc [24] validation set.

(a) Ablation study of Gaussian initialization strategies. PM-Point denotes probabilistic modeling with point cloud in [22].

Module	Single-Sweep Point	PM-Point	Multi-Sweep Voxel	IoU↑	mIoU↑
V2G	✓			36.7	22.4
		✓		34.9	21.2
			✓	40.7	25.8

(b) Ablation study of LiDAR voxel size for V2G. The unit of length is m. We set the height of all the voxels as 0.2m.

Module	0.15×0.15	0.1×0.1	0.075×0.075	IoU↑	mIoU↑
V2G	✓			40.1	25.0
		✓		40.6	25.2
			✓	40.7	25.8

(c) Ablation study of offset sampling methods for DFA. We run experiments with applying learnable offset sampling before and after projecting Gaussians into the lifted 3D feature space.

Module	Sampling Before Projection	Sampling After Projection	IoU↑	mIoU↑
DFA	✓		37.7	24.5
		✓	40.1	26.1
	✓	✓	40.7	26.4

(d) Ablation study of feature lifting and aggregating methods for DFA. We concatenate LiDAR sparse and dense depth maps with RGB features respectively to conduct 2D deformable attention.

Module	2D-Sparse Depth Map	2D-Dense Depth Map	3D	IoU↑	mIoU↑
DFA	✓			36.1	22.2
		✓		36.6	22.1
			✓	40.7	26.4

icantly improves the model's ability to detect both small objects (e.g., *pedestrian*, *traffic cone*) and large surfaces (e.g., *manmade*, *vegetation*). This validates the effectiveness of multi-sweep LiDAR scans in providing Gaussians with accurate geometric information of occupied space. We also notice that LiDAR-guided 3D deformable attention mechanism enhances the model's prediction ability on dynamic vehicles (e.g., *bicycle, bus, car, motorcycle, trailer, truck*) and near-road surfaces (e.g., *drivable surface, flatten area, sidewalk, terrain*) where objects detected by LiDAR points are visible to surrounding cameras. In these regions, the LiDAR points and corresponding image pixels are associated in the lifted 3D feature space, enabling the model to retrieve aggregated fusion features of on-road and near-road objects.

Voxel-to-Gaussian Initialization. We first compare different levels of LiDAR features used for initializing Gaussian properties in Tab. VIIa. The improvement achieved with the multi-sweep voxel feature is significantly greater than that of the single-sweep point feature and the point cloud probabilistic modeling strategy used in GaussianFormer-2 [22], which validates the effectiveness of our proposed module. We further conduct an ablation study on the size

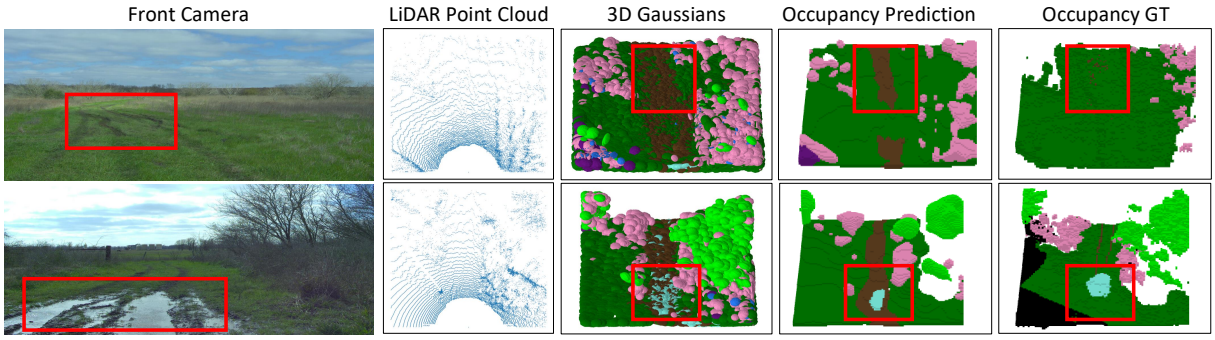


Fig. 5: Qualitative results on the off-road RELLIS3D-WildOcc [26] test set. Our method can outperform the GT (first row) at some regions and predict classes such as *puddle* that are vital for off-road autonomous driving (second row). Color legend is given in Tab. III.

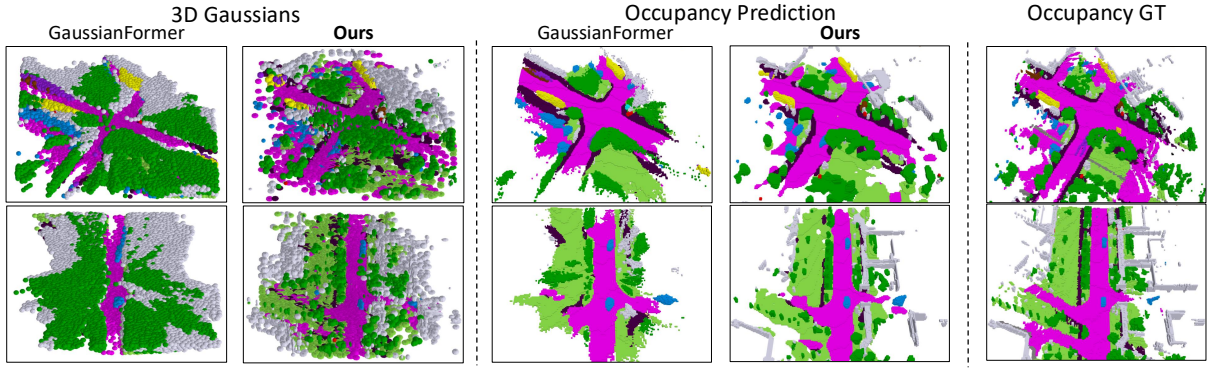


Fig. 6: Visualization comparison with GaussianFormer [21] on nuScenes-SurroundOcc [24]. By incorporating LiDAR, our method can obtain Gaussians with more adaptive scales and shapes, resulting in more accurate semantic predictions and delicate geometry details.

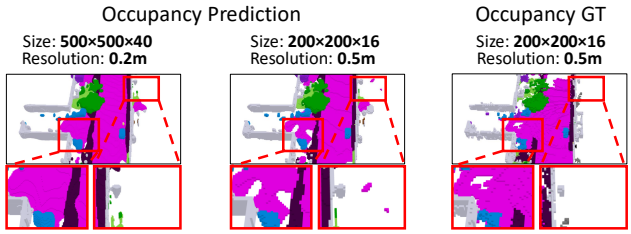


Fig. 7: Multi-resolution occupancy prediction of 3D Gaussians.

of LiDAR voxel in initialization in Tab. VIIb. As the voxel size decreases, the model performance slightly improves. We choose $0.075\text{m} \times 0.075\text{m} \times 0.2\text{m}$ as the final size.

LiDAR-Guided 3D Deformable Attention. We first study the effect of the two-stage offset sampling strategy in Tab. VIIc. We observe that applying learnable offset sampling both before and after projection achieves higher performance than single-stage sampling, which validates our two-stage sampling method can aggregate sufficient informative features for refining Gaussians. We also compare different feature aggregating methods in Tab. VIId, including 3D deformable attention, 2D deformable attention with concatenated LiDAR sparse depth map and with completed dense depth map [53]. The results validate our final choice.

E. Qualitative Results

We visualize 3D Gaussians and occupancy to qualitatively verify the effectiveness of our method for on-road scenes in Fig. 4. Our method can accurately predict both semantics and fine-grained geometry of the surrounding environments. In some cases, it even outperforms the GT by correctly completing occupancy in regions that lack semantic annotations. Qualitative results of our method on off-road scenes are given in Fig. 5. Our method is able to predict semantic occupancy for classes like *mud* and *puddle*, which are essential for achieving safe and effective off-road autonomous driving. We further compare our approach with GaussianFormer [21] in Fig. 6. The Gaussians in our method are more adaptive in scales and shapes, precisely appearing in the occupied regions of objects in both long-range and short-range areas, aided by the LiDAR sensor. Additionally, compared to voxel-based discretized approaches that train and predict at a fixed resolution, our method can predict multi-resolution semantic occupancy without additional training cost, attributed to the continuous property of Gaussians. This property enables more accurate and smoother prediction for certain areas when inferred at a higher resolution, as demonstrated in Fig. 7.

V. Conclusion

In this paper, we proposed GaussianFormer3D, a novel multi-modal semantic occupancy prediction framework that builds on 3D Gaussian scene representation. We introduced a voxel-to-Gaussian initialization strategy to endow 3D Gaussians with accurate geometry priors from LiDAR data. We also designed a LiDAR-guided 3D deformable attention mechanism to refine 3D Gaussians with LiDAR-camera fusion feature. Extensive experiments show its effectiveness in achieving accurate and fine-grained semantic occupancy prediction. However, our model is limited to fully-supervised manner, which requires densely annotated occupancy labels for training. In the future, we will explore its self-supervised variant and its application for multi-robot coordination.

REFERENCES

[1] Y. Zhang, J. Zhang, Z. Wang, J. Xu, and D. Huang, "Vision-based 3D occupancy prediction in autonomous driving: a review and outlook," *arXiv preprint arXiv:2405.02595*, 2024.

[2] S. Song, F. Yu, A. Zeng, A. X. Chang, M. Savva, and T. Funkhouser, "Semantic Scene Completion from a Single Depth Image," in *CVPR*, 2017.

[3] 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 *CVPR*, 2020.

[4] 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 *ICCV*, 2019.

[5] Y. Zhou and O. Tuzel, "VoxelNet: End-to-End Learning for Point Cloud Based 3D Object Detection," in *CVPR*, 2018.

[6] A. H. Lang, S. Vora, H. Caesar, L. Zhou, J. Yang, and O. Beijbom, "PointPillars: Fast Encoders for Object Detection from Point Clouds," in *CVPR*, 2019.

[7] L. Roldao, R. de Charette, and A. Verroust-Blondet, "LMSCNet: Lightweight Multiscale 3D Semantic Completion," in *3DV*, 2020.

[8] R. Cheng, C. Agia, Y. Ren, X. Li, and L. Bingbing, "S3CNet: A Sparse Semantic Scene Completion Network for LiDAR Point Clouds," in *CoRL*, 2021.

[9] 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 *CVPR*, 2023.

[10] J. Wilson, J. Song, Y. Fu, A. Zhang, A. Capodieci, P. Jayakumar, K. Barton, and M. Ghaffari, "MotionSC: Data Set and Network for Real-Time Semantic Mapping in Dynamic Environments," *IEEE RAL*, 2022.

[11] 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 *IROS*, 2021.

[12] 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 *IROS*, 2023.

[13] Y. Zheng, X. Li, P. Li, Y. Zheng, B. Jin, C. Zhong, X. Long, H. Zhao, and Q. Zhang, "MonoOcc: Digging into Monocular Semantic Occupancy Prediction," in *ICRA*, 2024.

[14] J. Pan, Z. Wang, and L. Wang, "Co-Occ: Coupling Explicit Feature Fusion With Volume Rendering Regularization for Multi-Modal 3D Semantic Occupancy Prediction," *IEEE RAL*, 2024.

[15] G. Wang, Z. Wang, P. Tang, J. Zheng, X. Ren, B. Feng, and C. Ma, "OccGen: Generative Multi-modal 3D Occupancy Prediction for Autonomous Driving," in *ECCV*, 2024.

[16] H. Li, Y. Hou, X. Xing, X. Sun, and Y. Zhang, "OccMamba: Semantic Occupancy Prediction with State Space Models," in *CVPR*, 2025.

[17] J. Zhang, Y. Ding, and Z. Liu, "OccFusion: Depth E stimation Free Multi-sensor Fusion for 3D Occupancy Prediction," in *ACCV*, 2024.

[18] Y. Ma, J. Mei, X. Yang, L. Wen, W. Xu, J. Zhang, X. Zuo, B. Shi, and Y. Liu, "LiCROcc: Teach Radar for Accurate Semantic Occupancy Prediction Using LiDAR and Camera," *IEEE RAL*, 2024.

[19] H. Cao and S. Behnke, "SLCF-Net: Sequential LiDAR-camera fusion for semantic scene completion using a 3D recurrent U-Net," in *ICRA*, 2024.

[20] B. Kerbl, G. Kopanas, T. Leimkühler, and G. Drettakis, "3D Gaussian Splatting for Real-time Radiance Field Rendering," *ACM TOG*, 2023.

[21] Y. Huang, W. Zheng, Y. Zhang, J. Zhou, and J. Lu, "GaussianFormer: Scene as Gaussians for Vision-based 3D Semantic Occupancy Prediction," in *ECCV*, 2024.

[22] Y. Huang, A. Thammatadatrakoon, W. Zheng, Y. Zhang, D. Du, and J. Lu, "GaussianFormer-2: Probabilistic Gaussian Superposition for Efficient 3D Occupancy Prediction," in *CVPR*, 2025.

[23] X. Zhu, W. Su, L. Lu, B. Li, X. Wang, and J. Dai, "Deformable DETR: Deformable Transformers for End-to-End Object Detection," in *ICLR*, 2021.

[24] Y. Wei, L. Zhao, W. Zheng, Z. Zhu, J. Zhou, and J. Lu, "SurroundOcc: Multi-camera 3D Occupancy Prediction for Autonomous Driving," in *ICCV*, 2023.

[25] X. Tian, T. Jiang, L. Yun, Y. Mao, H. Yang, Y. Wang, Y. Wang, and H. Zhao, "Occ3D: A Large-Scale 3D Occupancy Prediction Benchmark for Autonomous Driving," *NeurIPS*, 2024.

[26] H. Zhai, J. Mei, C. Min, L. Chen, F. Zhao, and Y. Hu, "WildOcc: A Benchmark for Off-Road 3D Semantic Occupancy Prediction," *arXiv preprint arXiv:2410.15792*, 2024.

[27] 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 *ICCV*, 2023.

[28] 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*, 2021.

[29] S. Zuo, W. Zheng, Y. Huang, J. Zhou, and J. Lu, "GaussianWorld: Gaussian World Model for Streaming 3D Occupancy Prediction," in *CVPR*, 2025.

[30] X. Zhou, Z. Lin, X. Shan, Y. Wang, D. Sun, and M.-H. Yang, "DrivingGaussian: Composite Gaussian Splatting for Surrounding Dynamic Autonomous Driving Scenes," in *CVPR*, 2024.

[31] Y. Yan, H. Lin, C. Zhou, W. Wang, H. Sun, K. Zhan, X. Lang, X. Zhou, and S. Peng, "Street Gaussians: Modeling Dynamic Urban Scenes with Gaussian Splatting," in *ECCV*, 2024.

[32] G. Hess, C. Lindström, M. Fatemi, C. Petersson, and L. Svensson, "SplatAD: Real-Time Lidar and Camera Rendering with 3D Gaussian Splatting for Autonomous Driving," in *CVPR*, 2025.

[33] W. Zheng, J. Wu, Y. Zheng, S. Zuo, Z. Xie, L. Yang, Y. Pan, Z. Hao, P. Jia, X. Lang, *et al.*, "GaussianAD: Gaussian-Centric End-to-End Autonomous Driving," *arXiv preprint arXiv:2412.10371*, 2024.

[34] S. Xu, F. Li, S. Jiang, Z. Song, L. Liu, and Z.-x. Yang, "GaussianPretrain: A Simple Unified 3D Gaussian Representation for Visual Pre-training in Autonomous Driving," *arXiv preprint arXiv:2411.12452*, 2024.

[35] Z. Qi, J. Ma, J. Xu, Z. Zhou, L. Cheng, and G. Xiong, "GSPR: Multimodal Place Recognition Using 3D Gaussian Splatting for Autonomous Driving," *arXiv preprint arXiv:2410.00299*, 2024.

[36] A. Vaswani, "Attention is all you need," in *NeurIPS*, 2017.

[37] J. Philion and S. Fidler, "Lift, Splat, Shoot: Encoding Images from Arbitrary Camera Rigs by Implicitly Unprojecting to 3D," in *ECCV*, 2020.

[38] H. Li, H. Zhang, Z. Zeng, S. Liu, F. Li, T. Ren, and L. Zhang, "DFA3D: 3D Deformable Attention For 2D-to-3D Feature Lifting," in *ICCV*, 2023.

[39] H. Fu, M. Gong, C. Wang, K. Batmanghelich, and D. Tao, "Deep Ordinal Regression Network for Monocular Depth Estimation," in *CVPR*, 2018.

[40] A.-Q. Cao and R. de Charette, "MonoScene: Monocular 3D Semantic Scene Completion," in *CVPR*, 2022.

[41] 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 *ECCV*, 2022.

[42] Y. Huang, W. Zheng, Y. Zhang, J. Zhou, and J. Lu, "Tri-Perspective View for Vision-Based 3D Semantic Occupancy Prediction," in *CVPR*, 2023.

[43] Y. Zhang, Z. Zhu, and D. Du, "OccFormer: Dual-path Transformer for Vision-based 3D Semantic Occupancy Prediction," in *ICCV*, 2023.

[44] 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.

[45] M. Pan, J. Liu, R. Zhang, P. Huang, X. Li, H. Xie, B. Wang, L. Liu, and S. Zhang, "RenderOcc: Vision-Centric 3D Occupancy Prediction with 2D Rendering Supervision," in *ICRA*, 2024.

[46] Q. Ma, X. Tan, Y. Qu, L. Ma, Z. Zhang, and Y. Xie, "COTR: Compact Occupancy Transformer for Vision-based 3D Occupancy Prediction," in *CVPR*, 2024.

[47] Y. Wang, Y. Chen, X. Liao, L. Fan, and Z. Zhang, "PanoOcc: Unified Occupancy Representation for Camera-based 3D Panoptic Segmentation," in *CVPR*, 2024.

[48] P. Jiang, P. Osteen, M. Wigness, and S. Saripalli, "RELLIS-3D Dataset: Data, Benchmarks and Analysis," in *ICRA*, 2021.

[49] K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition," in *CVPR*, 2016.

[50] T. Wang, X. Zhu, J. Pang, and D. Lin, "FCOS3D: Fully Convolutional One-Stage Monocular 3D Object Detection," in *ICCV*, 2021.

[51] T.-Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie, "Feature Pyramid Networks for Object Detection," in *CVPR*, 2017.

[52] 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," in *AAAI*, 2023.

[53] J. Ku, A. Harakeh, and S. L. Waslander, "In Defense of Classical Image Processing: Fast Depth Completion on the CPU," in *CRV*, 2018.