CHALMERS
UNIVERSITY OF TECHNOLOGY

Guided Gaussians: Enhancing 3D Occupancy Estimation with Sparse Sensor Priors

Guided Gaussians: Enhancing 3D Occupancy Estimation with Sparse Sensor Priors

Amer Mustajbasic[a,b,*,1], **Han Fu**[c,1], **Jialu Xu**[c,1], **Shuangshuang Chen**[d], **Erik Stenborg**[b] **and Selpi**[a]

[a]Chalmers University of Technology and University of Gothenburg
[b]Zenseact
[c]Lund University
[d]Volvo Car Corporation

Abstract. We introduce a new initialization method for 3D Gaussians used in 3D occupancy estimation, a key task in autonomous driving that involves identifying semantic elements in a vehicle's surroundings and accurately locating them in space. Our approach leverages distance sensor data, such as from lidar or radar, to place 3D Gaussians using farthest point sampling, ensuring coverage of meaningful scene areas while avoiding redundant representation of empty space. Unlike prior work that either densely voxelizes the scene or spreads 3D Gaussians uniformly, our method uses real sensor signals to drive object-centric placement, resulting in a more efficient and precise representation of the environment. We further enhance performance through a multimodal attention mechanism between 3D Gaussian features and distance sensor inputs, improving the integration of geometry and semantics. Our results show that this strategy consistently achieves state-of-the-art performance in 3D occupancy estimation. This contributes to a scalable solution for real-world deployment in autonomous vehicle perception systems, highlighting the potential of sensor-informed initialization for spatial reasoning in dynamic environments.

1 Introduction

In autonomous driving, robust scene understanding is critical for safe and reliable operation. To achieve this, modern vehicles are equipped with a diverse set of multimodal sensors such as cameras, lidar, and radar strategically placed to provide a 360-degree view of the surrounding environment. These sensors capture complementary types of information: visual cues from cameras, depth and surface data from lidar, and velocity or material-specific reflections from radar. However, integrating these distinct sensor modalities into a unified and consistent perception of the environment presents a significant challenge. The perception system must not only reconcile differences in resolution, range, and signal characteristics, but also fuse them to predict both the semantic content (e.g., object categories) and geometric structure (e.g., shape and location) of the scene.

A key task in this context is 3D occupancy prediction, which aims to represent both what is in the scene and where it is located in space. Traditional approaches [2, 3] often use dense voxel grids, where each voxel encodes a learned feature vector regardless of whether the voxel corresponds to an occupied or empty region. While effective to some extent, this method is computationally expensive and inefficient, particularly in sparse outdoor driving environments.

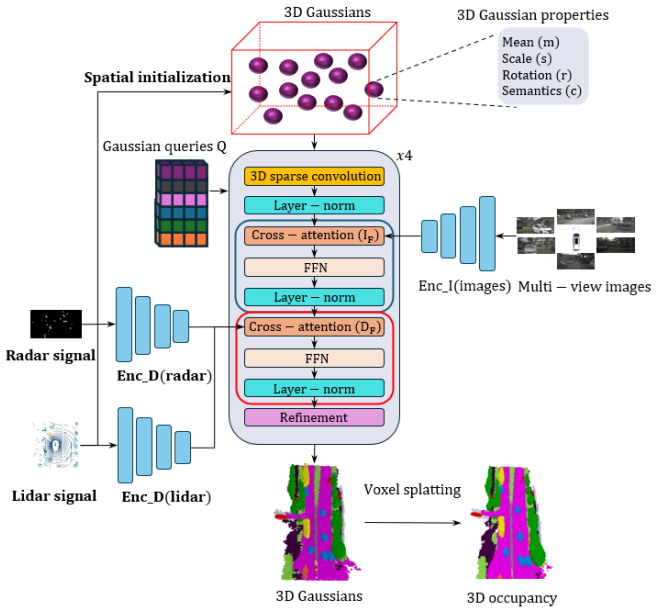

Figure 1: Architecture of Guided Gaussians ($\mathbf{G^2}$). Distance signals from lidar and/or radar are used to initialize the spatial means of the 3D Gaussians. Gaussian queries $\mathbf{Q}$ are employed to iteratively update the properties of the 3D Gaussians. Encoded distance features are integrated via cross-attention with the Gaussian queries $\mathbf{Q}$ and the image features using the cross-attention mechanism from the GaussianFormer [1] baseline.

To address this issue, recent works [1, 4], inspired by advances in 3D Gaussian splatting [5], use an alternative object-centric representation. These methods propose a sparse set of semantic 3D Gaussians to model scenes more efficiently, capturing the relevant structure without densely populating empty space. This change allows for more scalable and expressive modeling of complex real-world driving scenes. These methods rely on the initialization of the spatial positions of the 3D Gaussians as a critical component.

GaussianFormer [1] learns positions during training, while GaussianFormer2 [4] samples them based on a learned occupancy probability distribution. Each method conduct iterative refinement of the

* Corresponding Author. Email: amer.mustajbasic@chalmers.se.
[1] Equal contribution.

Gaussian mean through learned Gaussian queries. Although both works improve the performance of scene modeling, they overlook the raw distance signals from sensors such as lidar and radar, which give a valuable source of spatial information and are usually available in autonomous driving datasets. These signals inherently encode the scene structure of the non-empty areas and can serve as an informative prior for semantic 3D Gaussian placement.

In this work, we introduce Guided Gaussians ($\mathbf{G^2}$) (see Figure 1), a method that builds on the GaussianFormer framework [1] by explicitly leveraging multimodal distance signals to improve initialization and fusion. First, we propose a novel initialization strategy that uses Farthest Point Sampling [6] on lidar and/or radar distance returns as a prior to guide the placement of 3D Gaussians. This helps the model focus on the regions that are more likely to contain meaningful scene objects. Second, we incorporate the multimodal cross-attention mechanism [7] that further facilitates the interaction between Gaussian queries and features extracted from distance signals, allowing more effective multimodal fusion.

Through extensive experiments, we demonstrate that this sensor-guided initialization strategy, when combined with multimodal attention, significantly boosts performance on the 3D occupancy prediction task. Our approach not only improves upon the baseline GaussianFormer [1] but also outperforms other multimodal fusion methods for 3D Occupancy Prediction in terms of accuracy.

Our contributions are as follows:

- We propose a distance-signal-based initialization method for 3D Gaussians, improving their spatial alignment with occupied regions.
- We introduce a multimodal attention mechanism between Gaussian queries $\mathbf{Q}$ and distance sensor signals to enhance semantic and geometric fusion.
- We utilize less than ten percent of 3D Gaussians compared to GaussianFormer [1] baseline while achieving higher performance.
- We achieve state-of-the-art performance on 3D occupancy prediction benchmarks, compared to existing unimodal and multimodal approaches.

2 Related work

Semantic occupancy prediction has emerged as a core component of scene understanding in autonomous driving, enabling systems to reason about both free space and semantic content in the 3D environment. Early approaches often relied on dense voxel grids [2, 3], which discretize the scene into regularly spaced cells and learn features for each voxel. While effective in representing detailed structure, these methods are computationally expensive and suffer from poor scalability, particularly in large-scale or outdoor scenes where most of the space is empty.

To improve efficiency, more recent works have shifted toward sparse representations. Inspired by 3D Gaussian splatting [5], GaussianFormer [1] and its extension GaussianFormer2 [4] represent the scene using a set of semantic 3D Gaussians that can be rendered onto voxel grids via splatting. These models can reduce unnecessary computation in empty space by concentrating representation on regions of actual occupancy. However, their initialization strategies either learn Gaussian positions during training or sample from learned occupancy distributions, which do not take full advantage of spatial priors readily available from onboard sensors.

A separate line of work explores the integration of multimodal sensor data, particularly cameras, lidar, and radar, in order to build more robust scene representations. Fusion strategies range from early fusion, where raw signals are jointly processed, to late fusion techniques like BEVFusion [8] that align image and lidar features in bird's-eye view. BEVFusion4D [9] improves this by using lidar-guided view transformations but assumes all modalities are always available.

Other works have looked into radar-centric or radar-enhanced perception. CRN [10] uses radar view transformations and cross-attention to enhance visual features, and SimpleBEV [11] rasterizes sensor signals into a unified grid before lifting them into 3D. However, methods that rely heavily on convolutional backbones often struggle with capturing long-range dependencies and global context. To overcome this, DeepInteraction [12] introduces attention-based modality interaction but suffers from scalability issues as the number of modalities increases.

Multimodal robustness has become a key research direction. OccFusion [13] is a recent example that explicitly fuses camera, lidar, and radar data to improve performance under adverse conditions. Similarly, OpenOccupancy [14] establishes a benchmark for multimodal occupancy prediction, underscoring the value of integrating multiple sensing sources.

Generative approaches are also gaining traction. OccGen [15] formulates occupancy prediction as a generative task using a denoising diffusion model to recover fine-grained structure. Such methods offer not only strong performance but also uncertainty estimates, which are particularly valuable for safety-critical applications like autonomous driving.

Our method Guided Gaussians ($\mathbf{G^2}$) brings a new perspective by unifying sensor integration, and reasoning within a single 3D Gaussian-based framework. By directly initializing 3D Gaussians using raw sensor signals and refining them through iterative transformer updates, our approach tightly couples sensor observations with geometric structure from the very first layer. This enables efficient attention, compact representations, and interpretability without sacrificing performance, highlighting a path forward for structured and scalable 3D perception.

3 Method

We introduce Guided Gaussian ($\mathbf{G^2}$), a method that effectively leverages geometric information from distance signals, lidar and/or radar, to guide the placement of 3D Gaussian towards spatial regions with a high likelihood of occupancy. Furthermore, $\mathbf{G^2}$ additionally exploits the semantic content of these distance signals through a cross-attention mechanism with Gaussian queries $\mathbf{Q}$ and fuses them with features of multiview images to enhance the semantic understanding of scenes (see Figure 1).

3.1 3D Gaussian Prior

The 3D Gaussian is described as a set of properties $\mathbf{m} \in \mathbb{R}^3$, $\mathbf{s} \in \mathbb{R}^3$, $\mathbf{r} \in \mathbb{R}^4$, $\mathbf{c} \in \mathbb{R}^{|C|}$ where $\mathbf{m}$ is a mean i.e. spatial position, $\mathbf{s}$ is scale, $\mathbf{r}$ is rotation and $\mathbf{c}$ is semantic embedding. To initialize the spatial means of the 3D Gaussians in a way that aligns with the occupied regions of the scene, we adopt a two-stage strategy that combines guided sampling and random initialization and it is performed only once, prior to the start of the transformer block iterations.

The guided component leverages raw distance signals from lidar and/or radar to select a subset of meaningful spatial locations using Farthest Point Sampling (FPS) [6], while the remaining 3D Gaussian means are initialized randomly to preserve diversity. We choose FPS to represent the distribution of points more evenly across the

occupied space, ensuring that the selected points are spread out and well-distributed, especially in areas with varying density.

We denote the set of 3D points obtained from the distance signal as $\mathcal{P} = \{\mathbf{p}_1, \ldots, \mathbf{p}_v\} \subset \mathbb{R}^{3V}$, and the total number of 3D Gaussians to initialize as T, of which is the number of guided samples L ($L < T$) selected through FPS.

FPS algorithm to initiate Gaussian means begins by randomly selecting a seed point $p_r \in \mathcal{P}$, which forms the initial element of the guided set $\mathcal{M}_{\text{guided}}$. The selected point is removed from $\mathcal{P}$, and the Euclidean distances from all remaining points to p_r are computed. In each iteration, the algorithm updates the distance D_i for every remaining point in $\mathcal{P}$, setting it to the minimum of its current value and the distance to the most recently selected point. This ensures that each point in $\mathcal{P}$ always reflects its closest distance to any of the selected guided means. The point with the greatest minimum distance that is farthest from the current set of guided samples is then chosen and added to the set $\mathcal{M}_{\text{guided}}$, and removed from $\mathcal{P}$.

Once L such guided samples are chosen, the remaining $T - L$ 3D Gaussian means are initialized by randomly sampling spatial positions from a predefined spatial domain. The guided and random samples are concatenated to form the full set of initialized 3D Gaussian means, i.e., $\mathcal{M} = \mathcal{M}_{\text{guided}} \cup \mathcal{M}_{\text{random}}$.

This hybrid initialization approach ensures that a portion of the 3D Gaussians are placed in the informative regions of the scene, while still retaining flexibility to capture unobserved or ambiguous regions during training. The complete 3D Gaussian initialization is summarized in Algorithm 1.

Algorithm 1 Guided Initialization of 3D Gaussian Means

Require: Distance signal point set $\mathcal{P} = \{\mathbf{p}_1, \ldots, \mathbf{p}_V\} \subset \mathbb{R}^{3V}$, total number of Gaussians T, number of guided samples L ($L < T$)
Ensure: Initialized Gaussian means $\mathcal{M} = \{\mathbf{m}_1, \ldots, \mathbf{m}_T\} \subset \mathbb{R}^{3T}$

```
Randomly select p_v ∈ P and set M_guided ← {p_v}
Remove p_v from P
Compute D_i = ||p_i − p_v||_2 for all p_i ∈ P
for l = 2 to L do
    for each i = 1 to |P| do
        D_i ← min(D_i, ||p_i − m_{l−1}||_2)
    end for
    j ← arg max_i D_i
    m_l ← p_j
    M_guided ← M_guided ∪ {m_l}
    Remove p_j from P
end for
Initialize M_random with (T − L) random points in 3D space
Concatenate: M ← M_guided ∪ M_random
return M
```

3.2 Feature Encoding

We incorporate both image data from multi-view placed cameras and point cloud data from lidar and/or radar. Image data are denoted as $\mathbf{I} \in \mathbb{R}^{N \times C \times H \times W}$, where N, C, H, W stand for the number of views, channels, height and width respectively. Distance signals from lidar and radar are noted as $\mathbf{D}^b \in \mathbb{R}^{S \times K \times C_D}$, where b is signal type (lidar or radar), S is number of sweeps, K is the number of distance signal signatures and C_D is the number of distance signal features.

Spatial feature representations are extracted by the image encoder $\text{Enc}_I(\cdot)$ and denoted as $\mathbf{I}_F = Enc_I(\mathbf{I}) \in \mathbb{R}^{N \times L_F \times C_F \times H_F \times W_F}$ where L_F and C_F are the number of image feature levels and feature channel while W_F and H_F are width and height at each image feature level. Similar to [16], the distance signal encoder $\text{Enc}_D(\cdot)$ consists of three components: Voxelization, Backbone and FPN [17]. It generates distance signal features $\mathbf{D}_F^b = Enc_{D^b}(\mathbf{D}^b) \in \mathbb{R}^{C_F \times H_F \times W_F}$ where C_F is feature channel while W_F and H_F are spatial width and height of the distance signal feature map represented in Bird's Eye View (BEV). The distance signals are thereby reduced in the vertical dimension and represented as a 2D BEV feature grid.

3.3 Multimodal Cross-attention

Building upon the GaussianFormer [1], we follow its image cross-attention module to extract semantic information from images via Gaussian queries $\mathbf{Q}$ and image features $\mathbf{I}_F$ from image encoder $\text{Enc}_I(\cdot)$, and update 3D Gaussian properties.

To incorporate the GaussianFormer framework with additional distance signal, we introduce a multimodal cross-attention mechanism that allows the Gaussian queries $\mathbf{Q}$ to attend towards the distance signal features $\mathbf{D}_F^b$. Motivated by [7], the multimodal attention module adopts deformable attention [18], where the 3D Gaussians' mean $\mathbf{m}$ serves as reference points $\mathbf{m}_q$. Since the distance features are represented in the BEV space, the vertical dimension of the Gaussian mean $\mathbf{m}$ is ignored. Gaussian queries $\mathbf{Q}$ are used as attention queries $\mathbf{z}_q$. From the attention queries $\mathbf{z}_q$, we generate multimodal attention weights $\mathbf{A}_{hbqk}$ and reference point offsets $\Delta\mathbf{m}_{hbqk}$ as in [18]. Here, h stands for attention heads, b distance signals, q for query and k sampling coordinates index. Sampling coordinates are defined as $\mathbf{m}_s = \mathbf{m}_q + \Delta\mathbf{m}_{hbqk}$ and normalized $\mathbf{m}_s \in [0, 1]^2$ for bilinear sampling operator. Distance features $\mathbf{D}_F^b$ are sampled and attended using sampling coordinates $\mathbf{m}_s$ and attention weights $\mathbf{A}_{hbqk}$, aggregated over the spatial features space, distance signals and different attention heads. The procedure is shown in the Equation 1. $\mathbf{W}_{hb}$ represents parameter matrix applied to sampled signals and attention head.

$$\mathbf{CA}(\mathbf{z}_q, \mathbf{m}_q, \mathbf{D}_F^b) = \sum_{h=1}^{H} \sum_{b=1}^{B} \sum_{k=1}^{K} \mathbf{A}_{hbqk} \mathbf{W}_{hb} \mathbf{D}_F^b(\mathbf{m}_q + \Delta\mathbf{m}_{hbqk}), \quad (1)$$

We integrate this cross-attention module for the distance signals with the transformer of GaussianFormer [1], and it is followed by a feed-forward network and layer normalization [19] as in [1], thereby enhancing the model to fuse and interpret multimodal information.

3.4 Refinement

Following GaussianFormer [1], transformer block updates the 3D Gaussian means using Gaussian queries $\mathbf{Q}$ at each iteration of the refinement steps. Gaussian queries $\mathbf{Q}$ encode 3D information through 3D sparse convolution, image cross-attention and multimodal cross-attention when using distance signals. A multi-layer perceptron (MLP) decodes each query into updated Gaussian properties by adding to a residual $\hat{\mathbf{m}}$ to the Gaussian $\mathbf{m}$ mean and replacing the scale $\mathbf{s}$, rotation $\mathbf{r}$, and semantic features $\mathbf{c}$. This residual refinement ensures consistent mean $\mathbf{m}$ updates across transformer blocks, while substitution of other properties avoids vanishing gradient issues that could be caused by sigmoid and softmax activation functions.

3.5 Gaussian-to-Voxel splatting

We follow GaussianFormer [1] to perform Gaussian-to-Voxel splatting in the same way. The 3D Gaussians are first embedded into a

voxel grid based on their mean $\mathbf{m}$ i.e., spatial positions, and each Gaussian's voxel neighborhood is determined by its scale $\mathbf{s}$ property. The indices of Gaussians and their affected voxels are stored as pairs and sorted by voxel index to identify which Gaussians influence each voxel. This allows the model to efficiently approximate occupancy predictions using only the neighboring Gaussians for each voxel.

4 Experiment setup

We use Guided Gaussians ($\mathbf{G^2}$) with 3D semantic Gaussians to represent the scene in a compact and continuous manner. Each 3D Gaussian encodes semantic and spatial information, allowing for flexible fusion of multimodal signals such as images and distance-based measurements. Our experiments focus on the 3D occupancy prediction task, where the goal is to infer which regions of space are occupied by physical objects. We evaluate our method on two widely used multimodal autonomous driving benchmarks: NuScenes [20] and SemanticKITTI [21], both of which provide synchronized image and lidar data along with high-quality semantic annotations. These datasets enable us to assess the effectiveness of $\mathbf{G^2}$ in leveraging both geometry and semantics for accurate scene understanding.

4.1 Data

Dataset. We conduct training and evaluation on the NuScenes [20] and SemanticKITTI [21] datasets. For NuScenes [20], we utilize images from all six onboard cameras and aggregate data from six lidar and/or radar sweeps. The lidar input comprises five channels: three representing distance measurements, one for intensity, and one for the timestamp. Radar input includes all 16 metadata channels along with three positional channels, without applying the built-in outlier filtering provided by the NuScenes API [20]. NuScenes [20] ground truth for semantic voxels is generated from SurroundOcc [2].

For SemanticKITTI [21], we use images from one front-facing RGB camera and single lidar sweep. The lidar data includes distance measurements and intensity values associated with each return. SematicKITTI ground truth [21] is provided by the dataset.

Data representation. For the NuScenes [20] dataset, we follow the setup from the GaussianFormer baseline [1], defining the 3D occupancy space as a voxel grid of size $200 \times 200 \times 16$ along the X, Y, and Z axes, respectively, with a voxel resolution of 0.5 meters. This configuration corresponds to a physical space of $100\,\text{m} \times 100\,\text{m} \times 8\,\text{m}$. The coordinate system adheres to the NuScenes lidar convention, where the X-axis points to the right, the Y-axis forward, and the Z-axis upward.

For the SemanticKITTI [21] dataset, we adopt the configuration used in OccFormer [3], defining the 3D occupancy space as a voxel grid of size $256 \times 256 \times 32$ with a voxel resolution of 0.2 meters. This yields a coverage of $51.2\,\text{m} \times 51.2\,\text{m} \times 6.4\,\text{m}$. The coordinate system is aligned with SemanticKITTI conventions, where the X-axis points forward, the Y-axis to the left, and the Z-axis upward.

We use the same total number of Gaussian queries as the number of 3D semantic Gaussians, though they are not explicitly paired. Both the Gaussian queries and voxelized distance signal features have a channel dimension of 128.

Augmentations. We follow baseline [1] and apply random flip and photometric distortions.

4.2 Occupancy task

The 3D occupancy task is formulated as the prediction of semantic labels within a voxelized 3D space, where evaluation is performed by comparing predictions against ground-truth annotations for each labeled voxel. Following the baseline [1] approach, we apply voxel splatting to render the 3D scene representation encoded by the 3D semantic Gaussians onto the voxel grid used for occupancy prediction.

For the NuScenes dataset [20], we utilize voxel-wise semantic labels provided by SurroundOcc [2] for both training and evaluation. Each voxel is assigned one of 18 classes, comprising 16 semantic categories, along with one empty and one unknown class.

For the SemanticKITTI dataset [21], we take the voxel labels provided by the dataset as ground truth for training and evaluation purposes. Specifically, sequences 1 - 7, 9, and 10 are designated as the training set, while sequence 8 is used as the validation set. We make use of the learning map provided by the dataset to map the original labels into the index range from 0 to 20. Additionally, a 21st class is added to represent the empty category.

4.3 Training and evaluation setup

For the NuScenes dataset [20], we use input images with a resolution of 1600×900. For SemanticKITTI [21], we use the resolution 1241×376. To construct a dense representation of the scene, for NuScenes [20] we aggregate six consecutive lidar and/or radar sweeps. As the image encoder, we adopt ResNet101-DCN [22] pretrained on FCOS3D [23]. For SemanticKITTI [21], we use single lidar sweep and image from a single camera.

To extract multi-scale image features, we use a Feature Pyramid Network (FPN) [17], producing features at four downsampling scales: $4\times$, $8\times$, $16\times$, and $32\times$. For Farthest Point Sampling (FPS), we use a CUDA optimized algorithm [24]. For initializing 3D semantic Gaussians, we adopt a hybrid strategy: 70% of the Gaussians are initialized using FPS on lidar signals, 20% using FPS on radar signals (when available), and the remaining 10% are randomly initialized. A similar ratio (70/30 or 20/80) is applied even when only one of the distance modalities is used.

We use 12800 3D semantic Gaussians and queries during training, and report results for 6400 Gaussians for efficiency analysis. Scene representations are refined over four transformer blocks, with distance signal initialization of the 3D Gaussians applied only at the first refinement step.

Training follows the baseline setup from [1], using the AdamW optimizer [25] with a weight decay of 0.01. The learning rate is linearly warmed up to 2×10^{-4} over the first 500 iterations, then decayed according to a cosine schedule. We apply early stopping, typically halting training before reaching 12 epochs. Our loss is the same as in the GaussianFormer [1] with cross entropy L_{ce} and lovasz-softmax [26] loss L_{lov} with final loss being $L = \sum_{i=1}^{B} \left(L_{\text{ce}}^{i} + L_{\text{lov}}^{i}\right)$.

For evaluation, we follow [27] and use mean Intersection-over-Union (mIoU) and Intersection-over-Union (IoU) defined as:

$$\text{IoU} = \frac{TP_c}{TP_c + FP_c + FN_c} \qquad \text{mIoU} \quad = \frac{1}{A} \sum_{c=1}^{A} \frac{TP_c}{TP_c + FP_c + FN_c}$$

where c denote non-empty classes, A number of classes while TP_c, FP_c, FN_c are the number of true positive, false positive and false negative predictions respectively.

All trainings and evaluations are performed using batch size 4 and A100 GPUs. The runtime measured on the same single GPU for the Guided Gaussians ($\mathbf{G^2}$) model using 6400 Gaussians is 0.26s without initialization. The runtime for FPS initialization using 6 lidar sweeps is 0.4s, making the total runtime for the model $0.4 + 0.26 = 0.66s$. The comparison runtime of the baseline [1] using the least number of Gaussians (25600) is 0.32s.

Table 1: **3D semantic occupancy prediction results on NuScenes [20] validation set.** Guided Gaussian ($\mathbf{G}^2$) is trained using a Gaussian prior, where 70% of the Gaussians are initialized with lidar signals and the remaining 30% are randomly initialized. No cross-attention with distance signals is employed. $\mathbf{G}^2$ outperforms existing state-of-the-art methods. The original TPVFormer [28] is trained using lidar-based segmentation labels, while TPVFormer* is supervised with dense occupancy annotations. Values for individual semantic classes are reported as mIoU. Color coding: red indicates the top-1 score, green the top-2 score, and blue the top-3 score.

Method	IoU	mIoU	barrier	bicycle	bus	car	const. veh.	motorcycle	pedestrian	traffic cone	trailer	truck	drive. suf.	other flat	sidewalk	terrain	manmade	vegetation
MonoScene [27]	23.96	7.31	4.03	0.35	8.00	8.04	2.90	0.28	1.16	0.67	4.01	4.35	27.72	5.20	15.13	11.29	9.03	14.86
Atlas [29]	28.66	15.00	10.64	5.68	19.66	24.94	8.90	8.84	6.47	3.28	10.42	16.21	34.86	15.46	21.89	20.95	11.21	20.54
BEVFormer [30]	30.50	16.75	14.22	6.58	23.46	28.28	8.66	10.77	6.64	4.05	11.20	17.78	37.28	18.00	22.88	22.17	13.80	**22.21**
TPVFormer [28]	11.51	11.66	16.14	7.17	22.63	17.13	8.83	11.39	10.46	8.23	9.43	17.02	8.07	13.64	13.85	10.34	4.90	7.37
TPVFormer* [28]	30.86	17.10	15.96	5.31	23.86	27.32	9.79	8.74	7.09	5.20	10.97	19.22	38.87	21.25	24.26	23.15	11.73	20.81
OccFormer [3]	31.39	19.03	18.65	10.41	23.92	30.29	10.31	14.19	13.59	10.13	12.49	20.77	38.78	19.79	24.19	22.21	13.48	21.35
SurroundOcc [2]	**31.49**	**20.30**	**20.59**	11.68	**28.06**	**30.86**	10.70	15.14	**14.09**	**12.06**	**14.38**	**22.26**	37.29	**23.70**	24.49	22.77	**14.89**	21.86
GF1(144000 gaussians) [1]	29.83	19.10	19.52	11.26	26.11	29.78	10.47	13.83	12.58	8.67	12.74	21.57	39.63	23.28	24.46	22.99	9.59	19.12
GF2(25600 gaussians) [4]	30.56	20.02	20.15	**12.99**	27.61	30.23	**11.19**	**15.31**	12.64	9.63	13.31	**22.26**	**39.68**	**23.47**	**25.62**	**23.20**	12.25	20.73
$\mathbf{G}^2$(6400 gaussians, 70% initialized by Lidar)	**44.97**	**27.06**	**27.47**	**18.43**	**31.30**	**34.89**	**19.64**	**20.83**	**20.67**	**16.33**	**22.26**	**28.73**	**41.22**	21.99	**26.18**	**26.02**	**34.03**	**43.01**
$\mathbf{G}^2$(12800 gaussians, 70% initialized by Lidar)	**46.31**	**28.54**	**29.82**	**17.95**	**31.51**	**36.65**	**20.94**	**23.08**	**22.16**	**17.80**	**22.59**	**29.38**	**42.64**	**25.38**	**28.05**	**27.75**	**36.31**	**44.62**

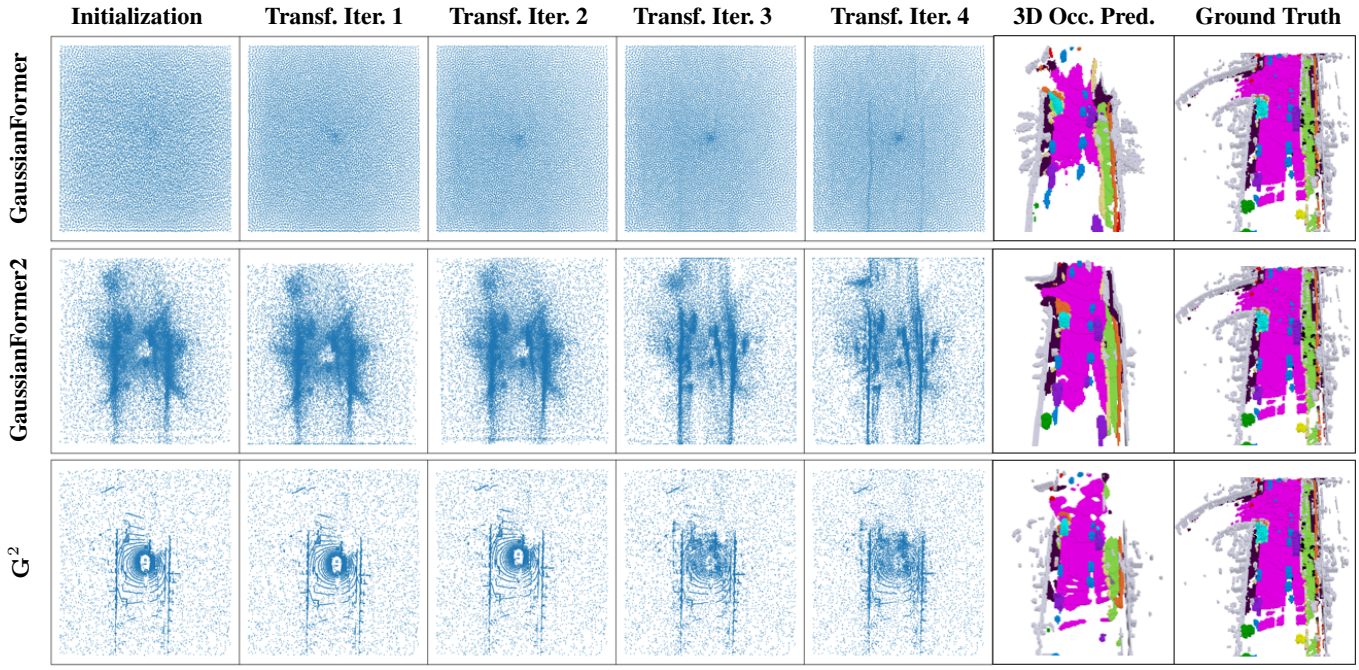

Figure 2: **Comparison of Gaussian movements and occupancy predictions across models.** From initialization through the final transformer iteration, $\mathbf{G}^2$ positions Gaussians more effectively to minimize empty space and concentrate on occupied voxels. The first three rows show outputs from GaussianFormer [1], GaussianFormer2 [4], and Guided Gaussians ($\mathbf{G}^2$), respectively, while the last column presents the ground truth. Columns: Initialization, Transformer Iterations 1–4, 3D Occupancy Prediction, and Ground Truth.

5 Results and Analysis

We demonstrate the effectiveness of the Gaussian mean intitialization and cross-attention with distance signals, lidar and/or radar.

5.1 Effects of the 3D Gaussian mean initialization

We first evaluate the impact of introduced 3D Gaussian prior, using only lidar distance signals. As shown in Table 1, our proposed $\mathbf{G}^2$ method consistently outperforms all camera-only methods, including GaussianFormer [1] and GaussianFormer2 [4], across both mIoU and IoU metrics, as well as on majority of a per-class basis.

Even with a reduced number of Gaussians, our method demonstrates strong performance: using only 6400 Gaussians, we achieve an IoU of 44.97 and mIoU of 27.06, compared to 31.49 and 20.30 which are achieved by state-of-the-art SurroundOcc [2]. When increasing the number of Gaussians to 12800, the accuracy of the prediction increases further to an IoU of 46.31 and mIoU of 28.54, illustrating the scalability and robustness of our approach.

Furthermore, in Table 5, we provide a direct comparison between our $\mathbf{G}^2$ method and the baseline approaches, which are also reported in their ablation study when utilizing lidar signals for 3D Gaussian mean initialization. Notably, our method achieves superior results de-

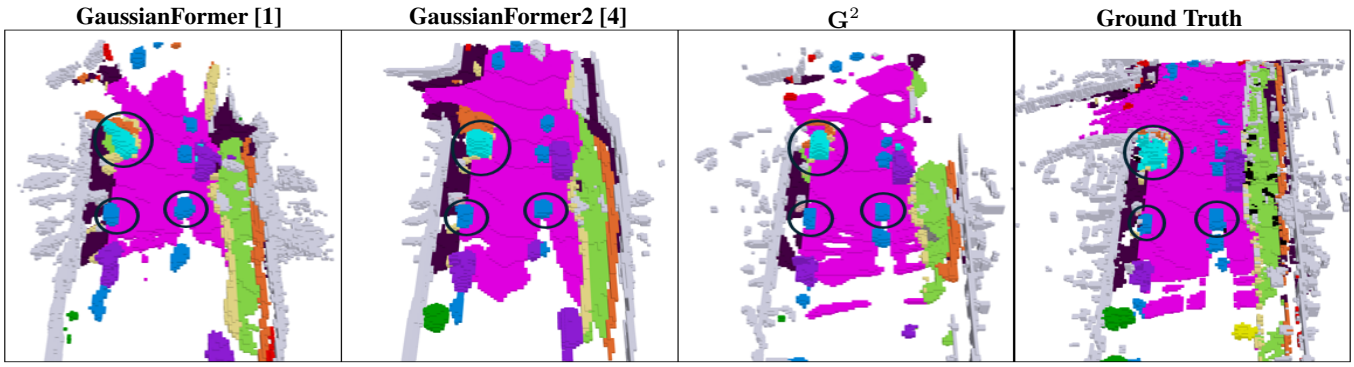

Figure 3: **Comparison of the accuracy of G^2 versus GaussianFormer [1] and GaussianFormer2 [4].** Although the ground truth contains noisy labels, G^2 assigns semantic labels to voxels more accurately than the baseline methods.

Table 2: **3D semantic occupancy prediction results on NuScenes [20] validation set.** Compared with other multimodal methods, Guided Gaussians **G²** outperforms state-of-the-art methods. Notations: Camera(C), Lidar(L), Radar(R), initialization(init), cross-attention (CA) and Voxelization (VX). Values for individual semantic classes are reported as mIoU. Color coding: red indicates the top-1 score, green the top-2 score, and blue the top-3 score.

Method	Backbone	Modality	IoU	mIoU	barrier	bicycle	bus	car	const. veh.	motorcycle	pedestrian	traffic cone	trailer	truck	drive. suf.	other flat	sidewalk	terrain	manmade	vegetation
OccFusion [13]	R101+VoxelNet	C+L	44.35	26.87	26.67	18.38	**32.97**	35.81	19.39	**22.17**	**24.48**	**17.77**	21.46	29.67	39.01	21.94	24.90	26.76	28.53	40.03
Co-Occ [31]	R101+VX	C+L	41.10	27.10	**28.10**	16.10	**34.00**	**37.20**	17.00	21.60	20.80	15.90	**21.90**	28.70	**42.30**	**25.40**	**29.10**	**28.60**	28.20	38.00
M-CONet [14, 31]	-	C+L	39.20	24.70	24.80	13.00	31.60	34.80	14.60	18.00	20.00	14.70	20.00	26.60	39.20	22.80	26.10	26.00	26.00	37.10
G²(12800 gaussians)	R101-DCN +VX	C+L (CA, L init)	**46.93**	**28.52**	**29.06**	**19.43**	31.46	**36.35**	**21.02**	21.92	21.44	**17.26**	**23.34**	**29.68**	**42.67**	**24.53**	**27.86**	**29.04**	**36.58**	**44.73**
OccFusion [13]	R101+VoxelNet	C+L+R	44.66	27.30	27.09	**19.56**	**33.68**	36.23	**21.66**	**24.84**	**25.29**	16.33	**21.81**	**30.01**	39.53	19.94	24.94	26.45	28.93	40.41
G²(12800 gaussians)	R101-DCN +VX	C+L (CA, L+R init)	**46.41**	**28.74**	**28.73**	**20.07**	32.35	**36.76**	**21.62**	**22.42**	**22.11**	**18.34**	21.77	**29.92**	**42.96**	**24.36**	**28.20**	**28.76**	**36.41**	**45.00**

spite using significantly fewer 3D Gaussians, highlighting the effectiveness of our initialization and representation strategy.

In Figure 2, we present visual examples of the refinement processes applied to the Gaussian prior among GaussianFormer [1], GaussianFormer2 [4] and our method **G²**. The Gaussian prior of **G²** using distance signals effectively locates the Gaussians toward regions in the scene that are more likely to be occupied. This improves the alignment of Gaussians with scene structures, making subsequent cross-attention operations, whether with image features, distance-based signals, or both, more semantically informative. From the same figure, it is evident that neither GaussianFormer [1] nor GaussianFormer2 [4] is capable of accurately placing Gaussians prior to refinement.

Figure 3 demonstrates that the **G²** method achieves higher accuracy in assigning semantic labels to voxels compared to GaussianFormer [1] and GaussianFormer2 [4], even with the presence of noisy and imprecise ground truth annotations.

5.2 *Effect of multimodal fusion on the 3D occupancy prediction*

We further compare our **G²** method in a multimodal configuration with other state-of-the-art fusion methods designed for 3D occupancy prediction using the same voxel-wise semantic labels. As shown in Table 2, our approach outperforms all existing methods of fusing image and lidar signals, achieving an IoU of 46.93 and an mIoU of 28.52. When adding radar to the 3D Gaussian initialization, the mIoU improves slightly from 28.52 to 28.74.

Table 3 presents the results of the **G²** method on the SemanticKITTI [21] dataset, compared against other multimodal fusion approaches. An analysis of class occurrence in the SemanticKITTI [21] dataset reveals that sample representation is very low for some classes like motorcycle (0.03%), bicycle (0.02%) and truck (0.16%), which likely contributes to our model's poor performance on those classes. In contrast, our model performs relatively well on the classes trunk (0.51%), building (14.41%) and vegetation (39.34%), which contributes to the overall superior performance in terms of IoU and mIoU.

5.3 *Ablation Studies*

In Table 4, we analyze how different combinations of initialization and cross-attention with distance signals influence 3D occupancy prediction. All experiments with **G²** use a fixed set of 12800 Gaussians.

Comparing row 1 in Table 4 with row 8 in Table 1, even without any explicit initialization, the **G²** model with cross-attention to lidar and randomly initialized, learnable Gaussian means, already achieves strong performance, despite using fewer than 10% of the Gaussians compared to GaussianFormer [1] (144000 Gaussians).

Using only radar as distance signal for cross-attention and initialization (row 3, Table 4) yields a better result than the case where only lidar is used as distance signal, solely for cross-attention (row 1, Table 4). This suggests that the radar is an informative prior, although the signal is noisy and sparse, serving as a strong baseline (compared to some state-of-the-art methods in Table 1) without any distance signal.

Lidar, being a precise and dense signal, proves especially effective: the model that uses lidar for both cross-attention and initialization (row 2, Table 4) achieves the best IoU. When radar is added to lidar-based initialization (row 4, Table 4), the model reaches the highest mIoU, indicating that combining distance signals for initialization can further enhance semantic reasoning.

Table 3: **3D semantic scene completion performance on SemanticKITTI validation set.** Notation of Modality: Camera(C), Lidar(L), Radar(R), initialization(init) and cross-attention (CA). Values for individual semantic classes are reported as mIoU. Highlighted values indicate the best scores.

Method	Modality	IoU	mIoU	road	sidewalk	parking	other-ground	building	car	truck	bicycle	motorcycle	other-vehicle	vegetation	trunk	terrain	person	bicyclist	motorcyclist	fence	pole	traf. sign
MonoScene [27]	C	37.12	11.50	57.47	27.05	15.72	**0.87**	14.24	23.55	7.83	0.20	0.77	3.59	18.12	2.57	30.76	1.79	1.03	0.00	6.39	4.11	2.48
TPVFormer [28]	C	35.61	11.36	56.50	25.87	**20.60**	0.85	13.88	23.81	8.08	0.36	0.05	4.35	16.92	2.26	30.38	0.51	0.89	0.00	5.94	3.14	1.52
OccFormer [3]	C	36.50	13.46	58.85	26.88	19.61	0.31	14.40	25.09	**25.53**	0.81	1.19	8.52	19.63	3.93	**32.62**	2.78	2.82	0.00	5.61	4.26	2.86
OccGen [15]	C+L	36.87	13.74	**61.28**	**28.30**	20.42	0.43	14.49	26.83	15.49	**1.60**	**2.53**	**12.83**	20.04	3.94	32.44	**3.20**	**3.37**	0.00	6.94	4.11	2.77
G²	C+L(CA, L init)	**49.53**	**14.11**	54.39	27.09	8.65	0.01	**27.12**	**30.74**	0.25	0.00	0.77	0.11	**33.49**	**15.78**	31.84	0.04	0.00	0.00	**10.03**	**21.40**	**6.40**

Table 4: **3D semantic occupancy prediction results on NuScenes [20] validation set.** Ablation study on various combinations of distance signals used for cross-attention and 3D Gaussian initialization. All experiments use 12800 Gaussians, a ResNet101-DCN image backbone, and voxelized distance signals. Initializing Gaussians with distance signals yields better performance than relying solely on cross-attention. Among distance sensor setups, lidar outperforms radar. Notion of the used modality: Camera(C), Lidar(L) and Radar(R). Values for individual semantic classes are reported as mIoU. Color coding: red indicates the top-1 score, green the top-2 score, and blue the top-3 score.

Cross-Attention	Initialization	IoU	mIoU	barrier	bicycle	bus	car	const. veh.	motorcycle	pedestrian	traffic cone	trailer	truck	drive. suf.	other flat	sidewalk	terrain	manmade	vegetation
C+L	-	30.25	19.16	18.04	12.98	26.27	28.98	10.81	14,94	12.47	9.73	13.02	20.60	39.19	21.89	24.28	22.80	10.22	20.37
C+L	L	**46.93**	**28.52**	**29.06**	**19.43**	**31.46**	**36.35**	**21.02**	**21.92**	**21.44**	**17.26**	**23.34**	**29.68**	**42.67**	**24.53**	**27.86**	**29.04**	**36.58**	**44.73**
C+R	R	31.28	19.53	18.73	13.19	24.85	29.35	12.45	15.41	12.38	10.28	12.19	21.82	38.72	21.41	23.65	21.68	15.51	20.89
C+L	L+R	**46.41**	**28.74**	**28.73**	**20.07**	**32.35**	**36.76**	**21.62**	**22.42**	**22.11**	**18.34**	**21.77**	**29.92**	**42.96**	**24.36**	**28.20**	**28.76**	**36.41**	**45.00**
C+R	L+R	45.52	27.06	27.02	**17.09**	**30.30**	**34.67**	19.31	**20.79**	20.85	15.29	**21.81**	27.66	41.49	23.44	26.29	27.57	35.53	43.90
C+L+R	L+R	**46.46**	**27.48**	**27.67**	16.99	29.76	34.26	**20.50**	20.38	**20.97**	**16.97**	21.00	**28.94**	**41.98**	**24.00**	**27.49**	**28.52**	**35.99**	**44.31**

Table 5: **3D semantic occupancy prediction results on NuScenes [20] validation set.** Compared with similar setup on the initialization and reported metrics in the baseline GaussianFormer [1] and GaussianFormer2 [4], Guided Gaussians **G²** perform better with smaller number of 3D Gaussians when 70% of Gaussians are initialized by lidar.

Method	Gaussian Number	IoU	mIoU
GaussianFormer [1]	51200	41.81	26.78
GaussianFormer2 [4]	25600	34.91	21.17
G²	6400	44.97	27.06
G²	12800	**46.31**	**28.54**

However, in the full multimodal setup (row 6, Table 4), which uses both lidar and radar for cross-attention and initialization, IoU drops slightly to 46.46 and mIoU to 27.48. It suggests that combining distance signals from radar and lidar for both cross-attention and initialization may slightly degrade overall performance. This effect is attributed to the inherent properties of the radar signals in the dataset. Since no radar signal filtering is applied, the radar input contains noise, and attending to radar alongside lidar adds the additional challenge of mitigating the radar-induced noise.

Next, we explore how proportions of distance signal usage affect Gaussian mean initialization. As shown in Table 6, initializing all Gaussians using distance information performs worse than initializing only 70% of them. This suggests that leaving a portion of the Gaussians uninitialized allows the model to adapt more flexibly, especially in dynamic or unobserved regions of the scene where the lidar signal may be sparse or absent. Allowing some Gaussians to move freely enables a better overall coverage and semantic representation.

Table 6: **Ablation study on different ratios of Gaussian mean initialization using lidar, radar and random placement with varying number of Gaussians.** Ratio represents how many Gaussians are initialized by different distance signals and randomly. The ratio values (in percent) indicate the proportion of Gaussians initialized by each source. For example, 30% radar means 30% of the Gaussians are initialized using radar signal. The optimal ratio of lidar initialization appears to be around 70% of Gaussians. Notion: Lidar(L), Radar(R) and Random (Rdm)

No. Gaussians	L (%)	R (%)	Rdm (%)	IoU	mIoU
12800	40	0	60	46.24	27.78
12800	70	0	30	46.32	27.92
12800	100	0	0	44.14	24.06
6400	30	20	50	38.81	23.10
6400	50	20	30	41.68	25.11
6400	70	20	10	43.81	25.85

6 Conclusion

We presented **G²**, a 3D occupancy prediction method based on 3D semantic Gaussians with guided initialization of the 3D Gaussian means using structured distance signals from lidar and radar to better capture likely occupied regions. On NuScenes [20] and SemanticKITTI [21], **G²** outperforms existing unimodal and multimodal methods in both IoU and mIoU metrics while using far fewer 3D Gaussians. **G²** shows strong results with lidar-only initialization and competitive performance in full multimodal setups, highlighting the method's flexibility. Furthermore, naive fusion of radar and lidar can reduce performance, requiring careful multimodal integration. Despite computational overhead from FPS sampling, **G²** provides a scalable framework for structured 3D scene understanding, with future work targeting temporally consistent occupancy prediction.

Acknowledgements

This work was conducted as part of the project "Deep Multimodal Learning for Automotive Applications", funded by Sweden's Innovation Agency Vinnova, grant no. 2023-00763. The computations were enabled by the Zenseact and resources provided by the National Academic Infrastructure for Supercomputing in Sweden (NAISS), partially funded by the Swedish Research Council through grant agreement no. 2022-06725.

References

[1] Yuanhui Huang, Wenzhao Zheng, Yunpeng Zhang, Jie Zhou, and Jiwen Lu. Gaussianformer: Scene as gaussians for vision-based 3d semantic occupancy prediction. In *European Conference on Computer Vision*, pages 376–393. Springer, 2024.

[2] Yi Wei, Linqing Zhao, Wenzhao Zheng, Zheng Zhu, Jie Zhou, and Jiwen Lu. Surroundocc: Multi-camera 3d occupancy prediction for autonomous driving. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 21729–21740, 2023.

[3] Yunpeng Zhang, Zheng Zhu, and Dalong Du. Occformer: Dual-path transformer for vision-based 3d semantic occupancy prediction. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9433–9443, 2023.

[4] Yuanhui Huang, Amonnut Thammatadatrakoon, Wenzhao Zheng, Yunpeng Zhang, Dalong Du, and Jiwen Lu. Probabilistic gaussian superposition for efficient 3d occupancy prediction. *arXiv preprint arXiv:2412.04384*, 2024.

[5] Bernhard Kerbl, Georgios Kopanas, Thomas Leimkühler, and George Drettakis. 3d gaussian splatting for real-time radiance field rendering. *ACM Trans. Graph.*, 42(4):139–1, 2023.

[6] Yuval Eldar, Michael Lindenbaum, Moshe Porat, and Yehoshua Y Zeevi. The farthest point strategy for progressive image sampling. *IEEE transactions on image processing*, 6(9):1305–1315, 1997.

[7] Amer Mustajbasic, Shuangshuang Chen, Erik Stenborg, and Selpi. Smab: Simple multimodal attention for effective bev fusion. In *2025 IEEE Intelligent Vehicles Symposium (IV)*, pages 1766–1772, 2025. doi: 10.1109/IV64158.2025.11097770.

[8] Zhijian Liu, Haotian Tang, Alexander Amini, Xinyu Yang, Huizi Mao, Daniela L. Rus, and Song Han. Bevfusion: Multi-task multi-sensor fusion with unified bird's-eye view representation. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pages 2774–2781, 2023. doi: 10.1109/ICRA48891.2023.10160968.

[9] Hongxiang Cai, Zeyuan Zhang, Zhenyu Zhou, Ziyin Li, Wenbo Ding, and Jiuhua Zhao. Bevfusion4d: Learning lidar-camera fusion under bird's-eye-view via cross-modality guidance and temporal aggregation, 2023. URL https://arxiv.org/abs/2303.17099.

[10] Youngseok Kim, Juyeb Shin, Sanmin Kim, In-Jae Lee, Jun Won Choi, and Dongsuk Kum. Crn: Camera radar net for accurate, robust, efficient 3d perception, 2023. URL https://arxiv.org/abs/2304.00670.

[11] Adam W. Harley, Zhaoyuan Fang, Jie Li, Rares Ambrus, and Katerina Fragkiadaki. Simple-bev: What really matters for multi-sensor bev perception? In *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pages 2759–2765, 2023. doi: 10.1109/ICRA48891.2023.10160831.

[12] Zeyu Yang, Jiaqi Chen, Zhenwei Miao, Wei Li, Xiatian Zhu, and Li Zhang. Deepinteraction: 3d object detection via modality interaction. In S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh, editors, *Advances in Neural Information Processing Systems*, volume 35, pages 1992–2005. Curran Associates, Inc., 2022. URL https://proceedings.neurips.cc/paper_files/paper/2022/file/0d18ab3b5fabfa6fe47c62e711af02f0-Paper-Conference.pdf.

[13] Zhenxing Ming, Julie Stephany Berrio, Mao Shan, and Stewart Worrall. Occfusion: Multi-sensor fusion framework for 3d semantic occupancy prediction. *IEEE Transactions on Intelligent Vehicles*, 2024.

[14] Xiaofeng Wang, Zheng Zhu, Wenbo Xu, Yunpeng Zhang, Yi Wei, Xu Chi, Yun Ye, Dalong Du, Jiwen Lu, and Xingang Wang. Openoccupancy: A large scale benchmark for surrounding semantic occupancy perception. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 17850–17859, 2023.

[15] Guoqing Wang, Zhongdao Wang, Pin Tang, Jilai Zheng, Xiangxuan Ren, Bailan Feng, and Chao Ma. Occgen: Generative multi-modal 3d occupancy prediction for autonomous driving, 2024. URL https://arxiv.org/abs/2404.15014.

[16] Ben Agro, Quinlan Sykora, Sergio Casas, Thomas Gilles, and Raquel Urtasun. Uno: Unsupervised occupancy fields for perception and forecasting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14487–14496, 2024.

[17] Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. Feature pyramid networks for object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2117–2125, 2017.

[18] Xizhou Zhu, Weijie Su, Lewei Lu, Bin Li, Xiaogang Wang, and Jifeng Dai. Deformable detr: Deformable transformers for end-to-end object detection. *arXiv preprint arXiv:2010.04159*, 2020.

[19] Jimmy Lei Ba, Jamie Ryan Kiros, and Geoffrey E Hinton. Layer normalization. *arXiv preprint arXiv:1607.06450*, 2016.

[20] Holger Caesar, Varun Bankiti, Alex H Lang, Sourabh Vora, Venice Erin Liong, Qiang Xu, Anush Krishnan, Yu Pan, Giancarlo Baldan, and Oscar Beijbom. nuscenes: A multimodal dataset for autonomous driving. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11621–11631, 2020.

[21] J. Behley, M. Garbade, A. Milioto, J. Quenzel, S. Behnke, C. Stachniss, and J. Gall. SemanticKITTI: A Dataset for Semantic Scene Understanding of LiDAR Sequences. In *Proc. of the IEEE/CVF International Conf. on Computer Vision (ICCV)*, 2019.

[22] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[23] Tai Wang, Xinge Zhu, Jiangmiao Pang, and Dahua Lin. Fcos3d: Fully convolutional one-stage monocular 3d object detection. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 913–922, 2021.

[24] Hengshuang Zhao, Li Jiang, Jiaya Jia, Philip HS Torr, and Vladlen Koltun. Point transformer. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 16259–16268, 2021.

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

[26] Maxim Berman, Amal Rannen Triki, and Matthew B Blaschko. The lovász-softmax loss: A tractable surrogate for the optimization of the intersection-over-union measure in neural networks. In *IEEE conf. on computer vision and pattern recognition*, pages 4413–4421, 2018.

[27] Anh-Quan Cao and Raoul De Charette. Monoscene: Monocular 3d semantic scene completion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3991–4001, 2022.

[28] Yuanhui Huang, Wenzhao Zheng, Yunpeng Zhang, Jie Zhou, and Jiwen Lu. Tri-perspective view for vision-based 3d semantic occupancy prediction, 2023. URL https://arxiv.org/abs/2302.07817.

[29] Zak Murez, Tarrence van As, James Bartolozzi, Ayan Sinha, Vijay Badrinarayanan, and Andrew Rabinovich. Atlas: End-to-end 3d scene reconstruction from posed images, 2020. URL https://arxiv.org/abs/2003.10432.

[30] Zhiqi Li, Wenhai Wang, Hongyang Li, Enze Xie, Chonghao Sima, Tong Lu, Qiao Yu, and Jifeng Dai. Bevformer: Learning bird's-eye-view representation from multi-camera images via spatiotemporal transformers, 2022. URL https://arxiv.org/abs/2203.17270.

[31] Jingyi Pan, Zipeng Wang, and Lin Wang. Co-occ: Coupling explicit feature fusion with volume rendering regularization for multi-modal 3d semantic occupancy prediction. *IEEE Robotics and Automation Letters*, 2024.