GUIDE: Gaussian Unified Instance Detection for Enhanced Obstacle Perception in Autonomous Driving

Chunyong Hu[1*], **Qi Luo**[1*], **Jianyun Xu**[1†], **Song Wang**[1,2], **Qiang Li**[1], **Sheng Yang**[1‡]

[1]Unmanned Vehicle Dept., CaiNiao Inc., Alibaba Group
[2]Zhejiang University

Abstract

In the realm of autonomous driving, accurately detecting surrounding obstacles is crucial for effective decision-making. Traditional methods primarily rely on 3D bounding boxes to represent these obstacles, which often fail to capture the complexity of irregularly shaped, real-world objects. To overcome these limitations, we present GUIDE, a novel framework that utilizes 3D Gaussians for instance detection and occupancy prediction. Unlike conventional occupancy prediction methods, GUIDE also offers robust tracking capabilities. Our framework employs a sparse representation strategy, using Gaussian-to-Voxel Splatting to provide fine-grained, instance-level occupancy data without the computational demands associated with dense voxel grids. Experimental validation on the nuScenes dataset demonstrates GUIDE's performance, with an instance occupancy mAP of 21.61, marking a 50% improvement over existing methods, alongside competitive tracking capabilities. GUIDE establishes a new benchmark in autonomous perception systems, effectively combining precision with computational efficiency to better address the complexities of real-world driving environments.

Code — https://github.com/CN-ADLab/GUIDE

1 Introduction

Accurate detection of surrounding obstacles is fundamental to autonomous driving systems. Despite rapid advancements in end-to-end driving technologies (Hu et al. 2023; Jiang et al. 2023a; Tong et al. 2023), obstacle instance detection continues to be a critical component, also serving as an auxiliary task to expedite model convergence. Traditional methods for obstacle detection (Huang et al. 2021; Liang et al. 2022; Li et al. 2024) typically employ 3D bounding boxes to denote the position and size of objects. While effective for standard obstacles, this simplistic representation falls short in complex environments with diverse distributions. Real-world driving scenarios, featuring irregular obstacles like barriers, debris, billboards, and complex situations such as pedestrians carrying objects or cars with open doors, challenge the effectiveness of 3D bounding boxes.

To address these issues, occupancy prediction has emerged as a promising solution, providing a more flexible approach for describing complex-shaped obstacles by predicting the occupancy and category of each voxel grid (Cao and De Charette 2022; Huang et al. 2023; Tian et al. 2023). However, traditional occupancy methods (Li et al. 2023b; Wei et al. 2023; Li et al. 2023c; Yu et al. 2023) rely on dense voxel features, resulting in memory usage that increases cubically with higher resolutions and broader perception ranges. Although some approaches (Liu et al. 2024) attempt to mitigate these limitations by using coarse-to-fine strategies, the substantial memory overhead continues to restrict the practical application of these methods. Furthermore, most traditional occupancy approaches focus solely on the semantic category of voxel occupancy. While some methods, like SparseOcc (Liu et al. 2024), venture into predicting instance-level occupancy masks, they are confined to per-frame predictions, lacking the capability to estimate instance velocity or perform temporal tracking—vital components for autonomous driving decision-making. Consequently, real-world systems often require combining the outputs of 3D object detection and tracking tasks with occupancy results, relying heavily on manual post-processing to achieve final perception outputs.

In light of these challenges, we are motivated to design a unified framework that not only offers efficient instance-level occupancy prediction but also integrates detection and tracking. Such a holistic framework provides a more comprehensive and robust perception solution for autonomous driving systems. Recent advancements in 3D Gaussian representations offer a promising pathway for this unified approach. 3D Gaussians (Kerbl et al. 2023) have gained traction in scene reconstruction due to their flexible and sparse representation abilities. Building on this foundation, the GaussianFormer series (Huang et al. 2024b,a) employ sparse Gaussians, generating occupancy predictions through Gaussian-to-Voxel Splatting, thereby reducing dependency on dense voxel features and lowering computational demands. Despite these advances, GaussianFormer largely supports semantic occupancy predictions without instance-level specificity.

Inspired by these developments, we propose GUIDE, a novel Gaussian-based Unified Instance Detection framework. GUIDE uses multiple 3D Gaussians to represent each

[*]These authors contributed equally.
[†]Project leader.
[‡]Corresponding author.

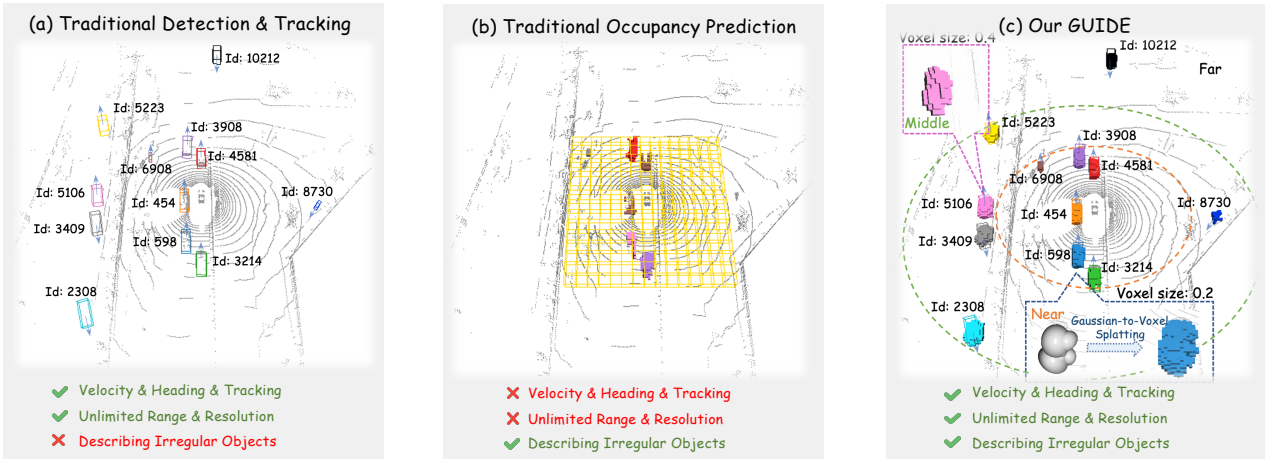

Figure 1: **Comparison of traditional detection and tracking, traditional occupancy prediction and our proposed GUIDE.** The background points are only shown for scene comprehension and are not used in model inference. (a) Traditional detection and tracking methods output 3D bounding boxes equipped with directional and velocity information, and each box is associated with a unique ID for tracking purposes. (b) Traditional occupancy prediction interacts with voxel features, producing rough occupancy predictions, yet it lacks directional and velocity information. (c) GUIDE represents each instance using multiple 3D Gaussians enriched with directional and velocity information as well as tracking IDs, and generates high-quality instance occupancy predictions via Gaussian-to-Voxel Splatting.

instance, employing Gaussian-to-Voxel Splatting for generating instance-level occupancy predictions. In addition, by leveraging Gaussian features, GUIDE outputs 3D bounding boxes and tracking IDs for each instance. This inherently sparse framework eliminates the necessity for dense voxel features, offering considerable memory efficiency. The continuous nature of Gaussian representations endows GUIDE with remarkable adaptability in modeling occupancy at varying resolutions. Specifically, the voxel size for occupancy prediction can be adjustably configured during inference, eliminating the need for retraining and facilitating efficient adaptation to diverse application requirements. We also introduce a novel mAP computation method tailored for assessing instance occupancy predictions. Experimental evaluations on the nuScenes (Caesar et al. 2020) dataset underscore our approach's effectiveness: GUIDE achieves an instance occupancy detection mAP of 21.61, showing a 50% improvement over SparseOcc, while maintaining competitive performance in detection and tracking tasks.

In summary, our primary contributions are as follows:

- We propose GUIDE, a Gaussian-based unified instance detection framework that supports instance-level occupancy prediction while simultaneously performing traditional 3D object detection and tracking.
- Our method employs a fully sparse representation for instance occupancy, greatly enhancing memory efficiency and allowing for flexible adjustment of inference resolution, thanks to the properties of Gaussian representation.
- GUIDE achieves an instance occupancy detection mAP of 21.61 on the nuScenes benchmark, reflecting a 50% improvement over SparseOcc, and delivers competitive performance in both detection and tracking tasks.

2 Related Work

2.1 Multi-view 3D Object Detection and Tracking

The task of vision-based multi-view 3D object detection is crucial for autonomous driving systems that rely on visual sensors (Park et al. 2021; Wang et al. 2021, 2022). Current methods in this field are typically categorized by whether they require dense Bird's Eye View (BEV) features. Among the methods that necessitate dense BEV features (Li et al. 2023a; Han et al. 2024), BEVDet (Huang et al. 2021) is noteworthy, utilizing the Lift-Splat-Shoot (LSS) (Philion and Fidler 2020) technique to project image features into BEV space using depth estimation. Another method generates BEV queries that are mapped into image space to sample and construct BEV features (Jiang et al. 2023b; Yang et al. 2023; Li et al. 2024). Conversely, methods that do not require dense BEV features follow distinct technical paths: the PETR series (Liu et al. 2022, 2023; Wang et al. 2023) employs implicit 3D position encoding within object queries and image features, enabling direct prediction of 3D bounding boxes through global attention mechanisms; the Sparse4D series (Lin et al. 2022, 2023a,b) utilizes explicit anchors to project and sample local features effectively for 3D object detection. There has been a notable evolution from early CNN-based detection decoders to query-based transformer decoders, and similarly, instance tracking has progressed from relying on dense features (Hu et al. 2022) to employing sparse queries (Zhang et al. 2022; Gu et al. 2023). The inherent alignment between queries and instances in transformer decoders makes them well-suited for modular, end-to-end autonomous driving frameworks like UniAD (Hu et al. 2023) and SparseDrive (Sun et al. 2024).

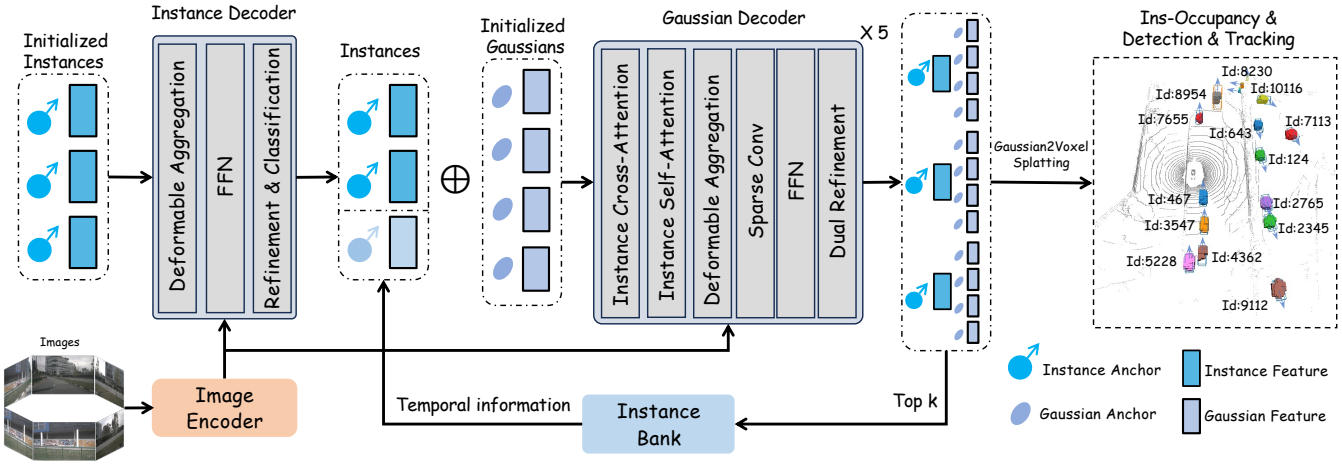

Figure 2: **Framework of our GUIDE.** Instance queries and their anchors are subsequently initialized and iteratively updated through interactions with image features using the instance decoder. The updated top-k instances are combined with those in the historical instance bank to form a new candidate instance set. Each instance is then associated with multiple 3D Gaussians, which serve as their representations. These Gaussians are refined iteratively through a 5-layer Gaussian Decoder. Subsequently, instance occupancy predictions are generated via Gaussian-to-Voxel Splatting. And aggregating Gaussian features allows reconstruction of instance-level representations to predict each instance's bounding box and category. Additionally, the top-k instances update the instance bank, adding temporal information to aid inference for later frames. Meanwhile, we assign unique IDs to instances whose confidence scores exceed a predefined threshold in the instance bank for instance tracking across frames.

2.2 Vision-based 3D Occupancy Prediction

Recent advances in vision-based 3D occupancy prediction have enhanced autonomous driving perception, particularly for long-tail corner cases involving diverse or atypical obstacles (Huang et al. 2023; Tian et al. 2023). MonoScene (Cao and De Charette 2022) pioneered end-to-end monocular 3D semantic grid prediction using a 2D-3D UNet (Ronneberger, Fischer, and Brox 2015). With the rise of Bird's Eye View (BEV) representations, subsequent methods leverage BEV features to reconstruct dense voxel occupancies (Yu et al. 2023; Li et al. 2023c; Zhang, Zhu, and Du 2023), though often at high computational cost. To improve efficiency, coarse-to-fine upsampling (Wang et al. 2024) and 2D rendering (Pan et al. 2024) have been proposed, yet they suffer from information loss and rely heavily on accurate depth estimation. While most work focuses on semantic occupancy, recent efforts explore panoramic occupancy prediction, expanding the task's scope (Liu et al. 2024; Moon et al. 2025).

2.3 Perception with 3D Gaussians

The use of 3D Gaussian splatting has gained attention in scene reconstruction for its exceptional rendering quality and speed (Kerbl et al. 2023). Recent studies highlight its potential in autonomous driving perception (Lu, Tsai, and Chen 2025). GaussianBEV (Chabot, Granger, and Lapouge 2025) achieves notable success in BEV segmentation by generating 3D Gaussians from image features and splatting them into BEV features. GaussianFormer (Huang et al. 2024b) pioneered Gaussian-to-Voxel Splatting, using 3D Gaussians for 3D semantic occupancy prediction, significantly improving efficiency. GaussianFormer-v2 (Huang et al. 2025) refines occupancy and semantic estimation within splatting, reducing Gaussian count. While the series offers memory efficiency over traditional methods via sparse representation, it only supports semantic — not instance-level — prediction. GaussianAD (Zheng et al. 2024) demonstrates 3D Gaussians' versatility across tasks like real-time mapping and scene reconstruction, further showcasing their potential in autonomous driving.

3 Proposed Approach

In this work, we introduce GUIDE, a novel unified framework designed to enhance obstacle perception in autonomous driving systems. GUIDE processes surround-view images, employing 3D Gaussian representations to efficiently produce instance-level occupancy predictions for nearby obstacles, while also offering strong detection and tracking capabilities.

3.1 Overall Architecture

As illustrated in Fig. 2, GUIDE consists of four essential components: an image encoder, an instance decoder, a Gaussian decoder, and an instance bank.

The image encoder processes multi-view images represented as $I \in \mathbb{R}^{N\times H\times W\times 3}$, where N is the number of surround-view cameras, and H and W are the image height and width, respectively. Leveraging a backbone and neck network, the image encoder extracts multi-scale features, denoted by $\mathbf{F} = \{\mathbf{F}_s\}_{s=1}^{4}$, where $\mathbf{F}_s \in \mathbb{R}^{N\times H_s\times W_s\times C}$, with s indicating the feature scales and C denoting the number of feature channels.

The instance decoder utilizes the extracted features and initialized anchors to identify individual object instances, while the Gaussian decoder employs multiple 3D Gaussian

representations for each instance to accurately model spatial occupancy. Aggregating Gaussian features enables the reconstruction of instance-level representations for category and 3D bounding box prediction. To address the adverse impact of partial object observation in single-frame images, we introduce an instance bank to enhance temporal feature fusion across sequences. Specifically, historical instance features are first concatenated with the outputs of the instance decoder, yielding a more comprehensive set of candidate instances. Next, in the Gaussian decoder, these historical features dynamically interact with current instance candidates, providing essential contextual information that improves the precision of occupancy predictions. Additionally, the instance bank assigns unique IDs to instances with confidence scores exceeding a predefined threshold (Lin et al. 2023b), facilitating instance tracking across frames.

3.2 Instance Decoder

In this module, instances are represented by a set of features $\mathbf{F}_I \in \mathbb{R}^{N_I \times C}$ and anchors $\mathbf{B}_I \in \mathbb{R}^{N_I \times 10}$. N_I denotes the number of candidate instances, and C represents the number of channels for the instance features. Each anchor comprises a 10-dimensional vector encoding the position, scale, yaw angle, and velocity. Instance features are initialized to zeros, while anchors are initialized using k-means clustering over the ground truth data from the training set. Then the information of each instance's anchor is encoded into $\mathbf{E}_I \in \mathbb{R}^{N_I \times C}$, which serves as the positional encoding for the instance features during attention operations.

The instance decoder consists of three modules: deformable aggregation, a feedforward network (FFN), and instance refinement. With deformable aggregation, each instance anchor center is projected onto the multi-view image features, generating both fixed and learnable offsets for feature sampling around the projected center. Image features aggregated from these sampling points are used to update the corresponding instance features. The updated features are further processed by the FFN and then passed to the instance refinement module, which predicts instance classes and anchor offset residuals to refine the anchors.

3.3 Gaussian Decoder

The top-k instances from the instance decoder, along with their corresponding features and anchors, are combined with those from the historical instance bank to form a new candidate instance set. In the Gaussian decoder, each instance is associated with a set of 3D Gaussians, parameterized by learnable Gaussian features $\mathbf{F}_{gs} \in \mathbb{R}^{K \times C}$ and Gaussian anchors $\mathbf{G} \in \mathbb{R}^{K \times 10}$, where K denotes the number of Gaussians per instance, and C s the feature dimension. Each Gaussian anchor encodes the offset from the instance center to the Gaussian mean, the scale, and the rotation quaternion. These initialized Gaussians are combined with candidate instances to produce instance-level Gaussian features $\mathbf{F}_{Igs} \in \mathbb{R}^{N_I \times K \times C}$ and Gaussian anchors $\mathbf{G}_I \in \mathbb{R}^{N_I \times K \times 10}$. The instance features could be obtained by averaging the Gaussian features for each instance.

The Gaussian decoder contains five layers. To enhance contextual information, each layer first performs cross-attention between instance features and those in the historical instance bank, followed by self-attention among the instance features. The updated instance features are then used to refine the corresponding Gaussian features. Each Gaussian feature, treated as a separate query, is further updated by the deformable aggregation module, which projects Gaussian anchors onto the image feature maps and samples relevant features. These aggregated features are combined with the Gaussian features, which are then processed by sparse 3D convolutions for local self-encoding.

In the dual refinement module, Gaussian and instance anchors are jointly updated using both Gaussian and instance features, where the instance features are computed by averaging their corresponding Gaussian features. The refined features and anchors are then passed to the next layer of the Gaussian decoder. In the final layer, the features and anchors of the top-k instances are used to update the historical instance memory bank.

3.4 Instance Gaussian Splatting

Inspired by the GaussianFormer series (Huang et al. 2024b, 2025), our approach employs a Gaussian-to-Voxel Splatting technique to predict 3D instance occupancy by splatting instance-level 3D Gaussians. Unlike GaussianFormer (Huang et al. 2024b), our method only performs geometric occupancy splatting, omitting semantic splatting. Instead, semantic labels for instance occupancy are directly obtained from the instance classification results. In this process, each 3D Gaussian represents a probabilistic spatial occupancy, where the probability at the center m is set to 1 and decreases with distance according to the 3D Gaussian distribution.

$$p(x; G) = \exp\left(-\frac{1}{2}(x-m)^T \Sigma^{-1}(x-m)\right) \quad (1)$$

$$\Sigma = RSS^T R^T \quad (2)$$

As illustrated in the equations above, $p(x; G)$ denotes the probability that point x is occupied as determined by 3D Gaussian G. Here, Σ is the covariance matrix, R the rotation matrix, and S the diagonal scale matrix.

In the process of splatting all Gaussians for each instance, the probability that a point is occupied by different Gaussians is considered to be independent. By applying the multiplication rule of probabilities, the aggregated probability that a point in space is occupied can be computed as follows.

$$p(x) = 1 - \prod_{i=1}^{K} (1 - p(x; G_i)) \quad (3)$$

3.5 Training Loss

During training, the supervised loss is composed of three components: regression, classification and occupancy.

$$\mathcal{L} = \mathcal{L}_{\text{reg}} + \mathcal{L}_{\text{cls}} + \mathcal{L}_{\text{occ}} \quad (4)$$

Methods	Backbone	Input Size	mAP↑	NDS↑	AMOTA↑	IDS↓	mIOU^{10}↑	mAP^{8}_{occ}↑	mAP^{10}_{occ}↑
DETR3D	R101	1333x800	30.3	37.4	-	-	-	-	-
PETR	R50	1056x384	31.3	38.1	-	-	-	-	-
BevFormer	R101	1600x900	**44.5**	**53.5**	-	-	-	-	-
ViP3D	R50	1333x800	-	-	21.7	-	-	-	-
QD-3DT	R101	1600x900	-	-	24.2	-	-	-	-
MUTR3D	R101	1600x900	-	-	29.4	3822	-	-	-
UniAD	R101	1600x900	38.0	49.8	35.9	906	-	-	-
SparseDrive-S	R50	704x256	41.8	52.5	38.6	886	-	-	-
RenderOcc	Swin-B	1408×512	-	-	-	-	17.30	-	-
OccFormer	R50	704x256	-	-	-	-	21.38	-	-
CTF-Occ	R101	960x640	-	-	-	-	27.97	-	-
GaussianFormer	R101	1600x864	-	-	-	-	**29.93**	-	-
SparseOCC	R50	704x256	-	-	-	-	25.31	14.40	-
GUIDE	R50	704x256	41.0	51.8	**39.4**	**516**	25.39	**21.61**	22.81

Table 1: **Performance comparison on the nuScenes val set.** The metric mIOU^{10} represents the mean Intersection over Union for all 10 foreground categories in the 3D semantic occupancy evaluation. The mAP^{8}_{occ} metric indicates the average instance occupancy mean Average Precision across thresholds $\{0.1, 0.2, 0.3\}$ for 8 categories, excluding the general obstacle category. Meanwhile, mAP^{10}_{occ} reflects the same evaluation performed across all 10 foreground categories.

We adopt the Hungarian matching algorithm to associate predicted instances with ground truth instances. Specifically, the regression tasks, which predict instance center positions, scales, and velocities, are supervised using the L1 loss function. For the instance classification tasks, we incorporate a multi-class focal loss. For occupancy, we also employ focal loss (Lin et al. 2017) to address the class imbalance between occupied and unoccupied voxels in the ground truth.

4 Experiments

4.1 Datasets

Our experiments are conducted on the nuScenes (Caesar et al. 2020) dataset, which contains 1000 sequences officially split into 750 for training, 150 for validation, and 150 for testing. Each sequence spans 20 seconds and provides keyframe detections and tracking ground truth at 2 Hz, along with six surround-view images.

To obtain occupancy ground truth, we use the annotations provided by Occ3D (Tian et al. 2023), which supplies occupancy labels for the nuScenes dataset in the ego vehicle coordinate system. The annotations cover a range of $[-40\,\text{m}, -40\,\text{m}, -1\,\text{m},\ 40\,\text{m},\ 40\,\text{m},\ 5.4\,\text{m}]$ with a voxel resolution of $[0.4\,\text{m},\ 0.4\,\text{m},\ 0.4\,\text{m}]$. The semantic labels in Occ3D are consistent with those used in nuScenes LiDAR segmentation tasks. To support foreground instance detection in our study, we further utilize tools from SparseOcc (Liu et al. 2024) to generate instance-level occupancy ground truth for foreground objects with IDs.

4.2 Evaluation Metrics

To our knowledge, GUIDE is the first method to address foreground instance occupancy prediction. For evaluation, we adopt a metric inspired by the mean Average Precision (mAP) commonly used in object detection. Specifically, we use the Intersection over Union (IoU) between predicted and ground-truth instance occupancies as the thresholding criterion, enabling finer-grained measurement. To comprehensively assess both the precision and recall of instance occupancy predictions, we compute mAP at multiple IoU thresholds $\{0.1, 0.2, 0.3\}$, and report the average as our final evaluation metric.

$$mAP_{occ} = Average(mAP_{0.1}, mAP_{0.2}, mAP_{0.3}) \quad (5)$$

Additionally, we employ the commonly used mean Intersection over Union (mIoU) metric from the 3D semantic occupancy task as a supplementary evaluation measure, which allows us to indirectly assess the semantic occupancy prediction capability of our method. Furthermore, we utilize the standard mAP and NDS metrics on the nuScenes dataset for detection performance evaluation, as well as AMOTA and IDS for tracking evaluation.

4.3 Main Results

In this section, we evaluate the performance of the proposed GUIDE on the nuScenes validation set. To the best of our knowledge, GUIDE is the first framework to address instance occupancy prediction focusing exclusively on the foreground categories in nuScenes. The most closely related prior work is SparseOcc, which employs a mask transformer for panoramic occupancy prediction on this dataset. However, there are key distinctions between the two approaches. SparseOcc performs instance-level predictions only for eight foreground categories, such as cars and pedestrians, and provides semantic occupancy predictions for the remaining classes. In contrast, our method extends instance occupancy prediction to additional obstacle categories, specifically barriers and traffic cones in nuScenes. This extension enables unified instance detection for all foreground obstacles.

To evaluate the performance of the proposed GUIDE on instance occupancy prediction, we conduct a comparative analysis with SparseOcc using the proposed instance occupancy mAP metric. This evaluation focuses on the eight

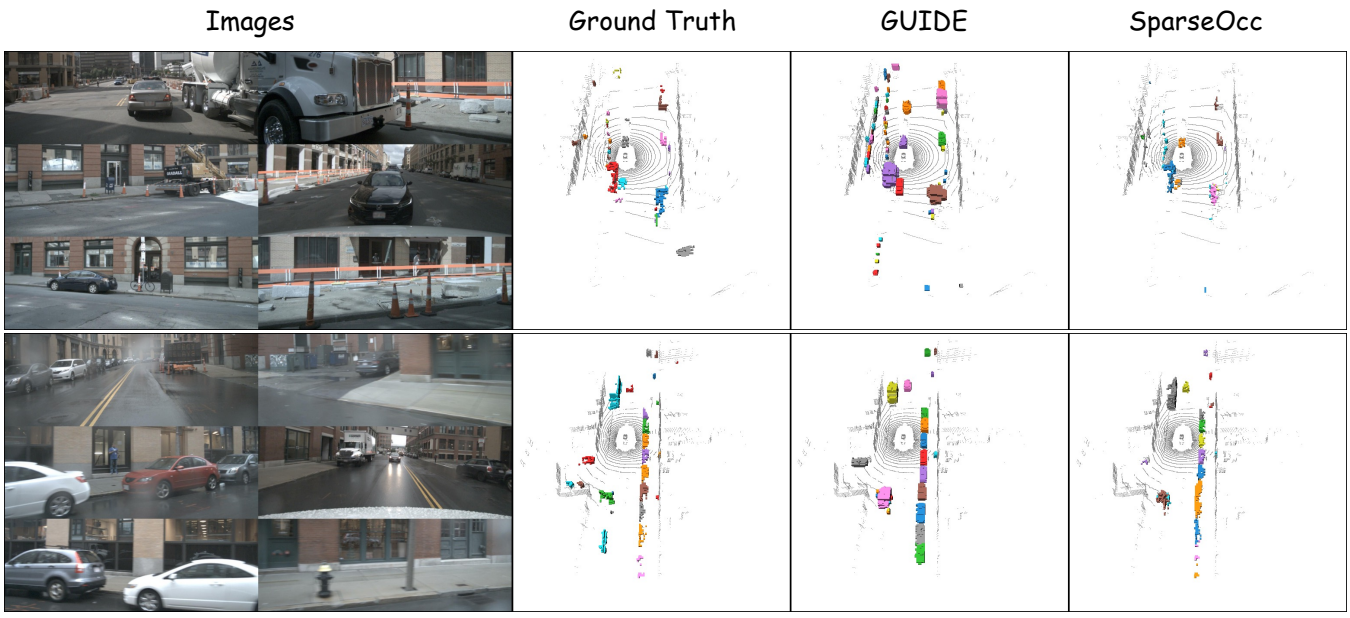

Figure 3: **Visualization results for 3D instance occupancy prediction on nuScenes.** The different colors are used to represent the occupancy predictions for various instances. The background points are included to aid scene comprehension and are not utilized during the model inference process.

Methods	barrier	traffic cone	mIOU↑	mAP_{occ}↑
SparseOCC	33.59	**24.39**	28.99	-
GUIDE	**39.14**	22.78	**30.96**	**27.79**

Table 2: **Performance for general obstacles on the nuScenes val set.** The mIOU metric represents the average mean IOU for the barrier and traffic cone categories, based on the 3D semantic occupancy evaluation. Similarly, the mAP metric is evaluated exclusively for them.

Level_{crowd}↑	Low($\leq$ 20)	Medium(20-40)	High($\geq$ 40)
GUIDE	32.35	28.16	31.92

Table 3: Performance of mAP_{occ} at IoU threshold 0.1 **under different Crowding Levels.** The level is linked to the instance number per scene.

foreground categories, including cars and pedestrians. As shown in Tab. 1, GUIDE achieves an average instance occupancy mAP of 21.61 for these categories, representing a 50% improvement over SparseOcc. For all foreground categories, including general obstacles, GUIDE attains a mean instance occupancy mAP of 22.81. These results indicate that GUIDE demonstrates exceptional accuracy and recall in instance occupancy prediction. We attribute this favorable performance to the instance-centric design of our method, which progressively refines spatial information at the instance level and thus achieves higher boundary precision in occupancy prediction.

Furthermore, to indirectly assess the semantic occupancy prediction capability of our method, we aggregate instance occupancy predictions according to their semantic categories. Since our method focuses on foreground categories, we recalculate the mIoU of the baselines for foreground classes alone to ensure a fair comparison. As shown in Tab. 1, our method achieves competitive performance even with a smaller backbone and lower-resolution inputs. Moreover, with a conservative allocation of only 48 Gaussians per instance, GUIDE outperforms our primary baseline, SparseOcc, in semantic occupancy accuracy, despite both being instance occupancy approaches.

Methods	Memory	mIOU^{10}↑	mAP^{8}_{occ}↑
SparseOcc	5424 MB	25.31	14.40
GUIDE	**3434 MB**	**25.39**	**21.61**

Table 4: **Comparison with SparseOcc** in terms of computation and memory.

Distinctively, GUIDE overcomes the limitations of prior methods that are restricted solely to occupancy prediction, as the intrinsic properties of 3D Gaussian representations naturally endow our framework with integrated detection and tracking capabilities. As shown in Tab. 1, GUIDE achieves competitive detection performance even with a relatively small network scale and low input resolution. Notably, GUIDE also exhibits improved multi-object tracking performance, particularly in terms of IDS, which primarily assesses the consistency of object tracking across frames. This notable gain in tracking stability may be attributed to the richer geometric and spatial information inherently encoded in the Gaussian representation, which facilitates a more stable and robust matching process during tracking.

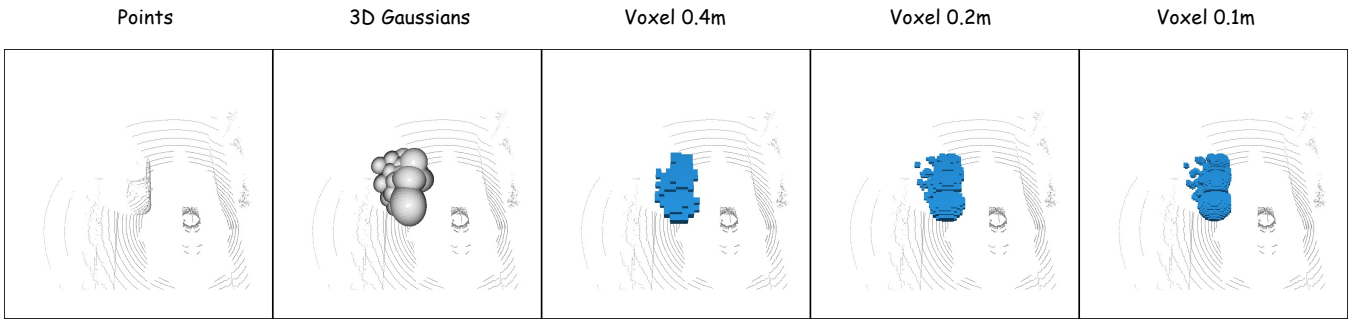

Figure 4: Visualization for occupancy **with different voxel sizes** when processing the Gaussian-to-Voxel Splatting.

In addition, Tab. 2 showcases the performance of GUIDE in predicting the semantic and instance-level occupancy of general obstacles. For semantic occupancy prediction, GUIDE achieves an mIoU of 30.96 for barriers and traffic cones, representing an improvement of 6.8% over SparseOcc. This result underscores GUIDE's ability to accurately segment these common obstacle categories. Furthermore, for instance-level occupancy prediction of general obstacles, GUIDE achieves an mAP of 27.79, highlighting its capability to precisely identify and predict the occupancy of such instances.

We further evaluate the instance discrimination ability of GUIDE under varying crowding levels. As shown in Tab. 3, performance remains stable even in dense scenarios, confirming its robustness to high instance density.

4.4 Visualizations

As illustrated in Fig. 3, we present the instance occupancy prediction results of GUIDE across various scenes, alongside a comparison with SparseOcc. The figure shows that, compared to SparseOcc, GUIDE produces significantly denser instance occupancy predictions. Moreover, GUIDE demonstrates superior instance recall capability. Even in scenarios with numerous targets, GUIDE maintains high accuracy while achieving outstanding recall performance.

Moreover, we visualize Gaussian-to-Voxel results at different resolutions. Leveraging the continuous nature of the Gaussian representation, GUIDE enables flexible adjustment of the predicted instance occupancy resolution at inference time without retraining. As illustrated in Fig. 4, reducing the voxel size leads to enhanced detail and precision in the occupancy prediction results. This demonstrates that our method offers strong flexibility in resolution. In practical autonomous driving systems, the splatting resolution for instance occupancy during inference can be dynamically adjusted as needed. For example, high resolution such as 0.1 m can be used for nearby instances requiring precise localization, while lower resolutions such as 0.4 m can be applied to distant targets.

4.5 Memory Efficiency

We further compare the memory efficiency of GUIDE with SparseOcc. The experiments are conducted on a single NVIDIA 3090 GPU with a batch size of 1. As shown in Tab. 4, GUIDE achieves a 36.7% reduction in GPU memory consumption during inference compared to SparseOcc. This result demonstrates the advantage of our sparse Gaussian representation in terms of memory efficiency.

Gaussian Number	Latency↓	mIOU↑	mAP_{occ}↑
16	**205ms**	22.79	18.94
32	291ms	23.31	20.21
48	373ms	**25.39**	**22.81**

Table 5: **Ablation on the number of Gaussians for each instance.** Both the mIOU and mAP metrics reflect evaluations conducted across all 10 foreground categories.

4.6 Ablation Study

In this section, we investigate the effect of varying the number of Gaussians used to model each instance on GUIDE's performance. As reported in Tab. 5, increasing the number of Gaussians per instance consistently improves instance occupancy accuracy, as it enables the model to represent more complex spatial distributions with higher fidelity. However, this gain comes at the cost of increased computational overhead, and this trade-off informs subsequent design choices by appropriately balancing higher prediction accuracy and computational efficiency.

5 Conclusion

In this work, we propose GUIDE, a novel Gaussian-based unified instance detection framework that employs 3D Gaussians as flexible and efficient intermediate representations for autonomous vehicle perception. By replacing dense voxel grids with Gaussian-to-Voxel Splatting, GUIDE substantially reduces computational and memory demands while enhancing the accuracy of instance occupancy prediction. Furthermore, GUIDE also supports both detection and tracking tasks, and achieves competitive performance. Comprehensive experiments on the nuScenes benchmark demonstrate that GUIDE outperforms the previous instance occupancy method, SparseOcc, in both memory efficiency and occupancy detection accuracy. These findings illustrate the promise of GUIDE as a robust and scalable solution for perception in end-to-end autonomous driving systems.

Acknowledgments

This research was supported by the Zhejiang Provincial Natural Science Foundation of China under Grant No. LD24F030001.

References

Caesar, H.; Bankiti, V.; Lang, A. H.; Vora, S.; Liong, V. E.; Xu, Q.; Krishnan, A.; Pan, Y.; Baldan, G.; and Beijbom, O. 2020. nuscenes: A multimodal dataset for autonomous driving. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 11621–11631.

Cao, A.-Q.; and De Charette, R. 2022. Monoscene: Monocular 3d semantic scene completion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 3991–4001.

Chabot, F.; Granger, N.; and Lapouge, G. 2025. Gaussianbev: 3d gaussian representation meets perception models for bev segmentation. In *2025 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, 2250–2259. IEEE.

Gu, J.; Hu, C.; Zhang, T.; Chen, X.; Wang, Y.; Wang, Y.; and Zhao, H. 2023. Vip3d: End-to-end visual trajectory prediction via 3d agent queries. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 5496–5506.

Han, C.; Yang, J.; Sun, J.; Ge, Z.; Dong, R.; Zhou, H.; Mao, W.; Peng, Y.; and Zhang, X. 2024. Exploring recurrent long-term temporal fusion for multi-view 3d perception. *IEEE Robotics and Automation Letters*.

Hu, H.-N.; Yang, Y.-H.; Fischer, T.; Darrell, T.; Yu, F.; and Sun, M. 2022. Monocular quasi-dense 3d object tracking. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(2): 1992–2008.

Hu, Y.; Yang, J.; Chen, L.; Li, K.; Sima, C.; Zhu, X.; Chai, S.; Du, S.; Lin, T.; Wang, W.; et al. 2023. Planning-oriented autonomous driving. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 17853–17862.

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

Huang, Y.; Thammatadatrakoon, A.; Zheng, W.; Zhang, Y.; Du, D.; and Lu, J. 2024a. Probabilistic Gaussian Superposition for Efficient 3D Occupancy Prediction. *arXiv preprint arXiv:2412.04384*.

Huang, Y.; Thammatadatrakoon, A.; Zheng, W.; Zhang, Y.; Du, D.; and Lu, J. 2025. Gaussianformer-2: Probabilistic gaussian superposition for efficient 3d occupancy prediction. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, 27477–27486.

Huang, Y.; Zheng, W.; Zhang, Y.; Zhou, J.; and Lu, J. 2023. Tri-perspective view for vision-based 3d semantic occupancy prediction. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 9223–9232.

Huang, Y.; Zheng, W.; Zhang, Y.; Zhou, J.; and Lu, J. 2024b. Gaussianformer: Scene as gaussians for vision-based 3d semantic occupancy prediction. In *European Conference on Computer Vision*, 376–393. Springer.

Jiang, B.; Chen, S.; Xu, Q.; Liao, B.; Chen, J.; Zhou, H.; Zhang, Q.; Liu, W.; Huang, C.; and Wang, X. 2023a. Vad: Vectorized scene representation for efficient autonomous driving. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 8340–8350.

Jiang, Y.; Zhang, L.; Miao, Z.; Zhu, X.; Gao, J.; Hu, W.; and Jiang, Y.-G. 2023b. Polarformer: Multi-camera 3d object detection with polar transformer. In *Proceedings of the AAAI conference on Artificial Intelligence*, volume 37, 1042–1050.

Kerbl, B.; Kopanas, G.; Leimkühler, T.; and Drettakis, G. 2023. 3d gaussian splatting for real-time radiance field rendering. *ACM Trans. Graph.*, 42(4): 139–1.

Li, Y.; Ge, Z.; Yu, G.; Yang, J.; Wang, Z.; Shi, Y.; Sun, J.; and Li, Z. 2023a. Bevdepth: Acquisition of reliable depth for multi-view 3d object detection. In *Proceedings of the AAAI conference on artificial intelligence*, volume 37, 1477–1485.

Li, Y.; Yu, Z.; Choy, C.; Xiao, C.; Alvarez, J. M.; Fidler, S.; Feng, C.; and Anandkumar, A. 2023b. Voxformer: Sparse voxel transformer for camera-based 3d semantic scene completion. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 9087–9098.

Li, Z.; Wang, W.; Li, H.; Xie, E.; Sima, C.; Lu, T.; Yu, Q.; and Dai, J. 2024. Bevformer: learning bird's-eye-view representation from lidar-camera via spatiotemporal transformers. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.

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

Liang, T.; Xie, H.; Yu, K.; Xia, Z.; Lin, Z.; Wang, Y.; Tang, T.; Wang, B.; and Tang, Z. 2022. Bevfusion: A simple and robust lidar-camera fusion framework. *Advances in Neural Information Processing Systems*, 35: 10421–10434.

Lin, T.-Y.; Goyal, P.; Girshick, R.; He, K.; and Dollár, P. 2017. Focal loss for dense object detection. In *Proceedings of the IEEE international conference on computer vision*, 2980–2988.

Lin, X.; Lin, T.; Pei, Z.; Huang, L.; and Su, Z. 2022. Sparse4d: Multi-view 3d object detection with sparse spatial-temporal fusion. *arXiv preprint arXiv:2211.10581*.

Lin, X.; Lin, T.; Pei, Z.; Huang, L.; and Su, Z. 2023a. Sparse4d v2: Recurrent temporal fusion with sparse model. *arXiv preprint arXiv:2305.14018*.

Lin, X.; Pei, Z.; Lin, T.; Huang, L.; and Su, Z. 2023b. Sparse4d v3: Advancing end-to-end 3d detection and tracking. *arXiv preprint arXiv:2311.11722*.

Liu, H.; Chen, Y.; Wang, H.; Yang, Z.; Li, T.; Zeng, J.; Chen, L.; Li, H.; and Wang, L. 2024. Fully sparse 3d occupancy prediction. In *European Conference on Computer Vision*, 54–71. Springer.

Liu, Y.; Wang, T.; Zhang, X.; and Sun, J. 2022. Petr: Position embedding transformation for multi-view 3d object detection. In *European conference on computer vision*, 531–548. Springer.

Liu, Y.; Yan, J.; Jia, F.; Li, S.; Gao, A.; Wang, T.; and Zhang, X. 2023. Petrv2: A unified framework for 3d perception from multi-camera images. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 3262–3272.

Lu, S.-W.; Tsai, Y.-H.; and Chen, Y.-T. 2025. Toward Real-world BEV Perception: Depth Uncertainty Estimation via Gaussian Splatting. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, 17124–17133.

Moon, S.; Baek, J.; Kim, G.; Kim, J.; and Choi, S. 2025. Mitigating trade-off: Stream and query-guided aggregation for efficient and effective 3d occupancy prediction. *arXiv preprint arXiv:2503.22087*.

Pan, M.; Liu, J.; Zhang, R.; Huang, P.; Li, X.; Xie, H.; Wang, B.; Liu, L.; and Zhang, S. 2024. Renderocc: Vision-centric 3d occupancy prediction with 2d rendering supervision. In *2024 IEEE International Conference on Robotics and Automation (ICRA)*, 12404–12411. IEEE.

Park, D.; Ambrus, R.; Guizilini, V.; Li, J.; and Gaidon, A. 2021. Is pseudo-lidar needed for monocular 3d object detection? In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 3142–3152.

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

Ronneberger, O.; Fischer, P.; and Brox, T. 2015. U-net: Convolutional networks for biomedical image segmentation. In *Medical image computing and computer-assisted intervention–MICCAI 2015: 18th international conference, Munich, Germany, October 5-9, 2015, proceedings, part III 18*, 234–241. Springer.

Sun, W.; Lin, X.; Shi, Y.; Zhang, C.; Wu, H.; and Zheng, S. 2024. Sparsedrive: End-to-end autonomous driving via sparse scene representation. *arXiv preprint arXiv:2405.19620*.

Tian, X.; Jiang, T.; Yun, L.; Mao, Y.; Yang, H.; Wang, Y.; Wang, Y.; and Zhao, H. 2023. Occ3d: A large-scale 3d occupancy prediction benchmark for autonomous driving. *Advances in Neural Information Processing Systems*, 36: 64318–64330.

Tong, W.; Sima, C.; Wang, T.; Chen, L.; Wu, S.; Deng, H.; Gu, Y.; Lu, L.; Luo, P.; Lin, D.; et al. 2023. Scene as occupancy. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 8406–8415.

Wang, S.; Liu, Y.; Wang, T.; Li, Y.; and Zhang, X. 2023. Exploring object-centric temporal modeling for efficient multi-view 3d object detection. In *Proceedings of the IEEE/CVF international conference on computer vision*, 3621–3631.

Wang, T.; Zhu, X.; Pang, J.; and Lin, D. 2021. Fcos3d: Fully convolutional one-stage monocular 3d object detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 913–922.

Wang, Y.; Chen, Y.; Liao, X.; Fan, L.; and Zhang, Z. 2024. Panoocc: Unified occupancy representation for camera-based 3d panoptic segmentation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 17158–17168.

Wang, Y.; Guizilini, V. C.; Zhang, T.; Wang, Y.; Zhao, H.; and Solomon, J. 2022. Detr3d: 3d object detection from multi-view images via 3d-to-2d queries. In *Conference on Robot Learning*, 180–191. PMLR.

Wei, Y.; Zhao, L.; Zheng, W.; Zhu, Z.; Zhou, J.; and Lu, J. 2023. Surroundocc: Multi-camera 3d occupancy prediction for autonomous driving. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 21729–21740.

Yang, C.; Chen, Y.; Tian, H.; Tao, C.; Zhu, X.; Zhang, Z.; Huang, G.; Li, H.; Qiao, Y.; Lu, L.; et al. 2023. Bevformer v2: Adapting modern image backbones to bird's-eye-view recognition via perspective supervision. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 17830–17839.

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

Zhang, T.; Chen, X.; Wang, Y.; Wang, Y.; and Zhao, H. 2022. Mutr3d: A multi-camera tracking framework via 3d-to-2d queries. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 4537–4546.

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

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