QuadricFormer: Scene as Superquadrics for 3D Semantic Occupancy Prediction

Sicheng Zuo[1,*] **Wenzhao Zheng**[1,*,†] **Xiaoyong Han**[1,*]
Longchao Yang[2] **Yong Pan**[2] **Jiwen Lu**[1,3]

[1]Tsinghua University [2]Li Auto Inc
[3]Beijing National Research Center for Information Science and Technology
Project Page: https://zuosc19.github.io/QuadricFormer/

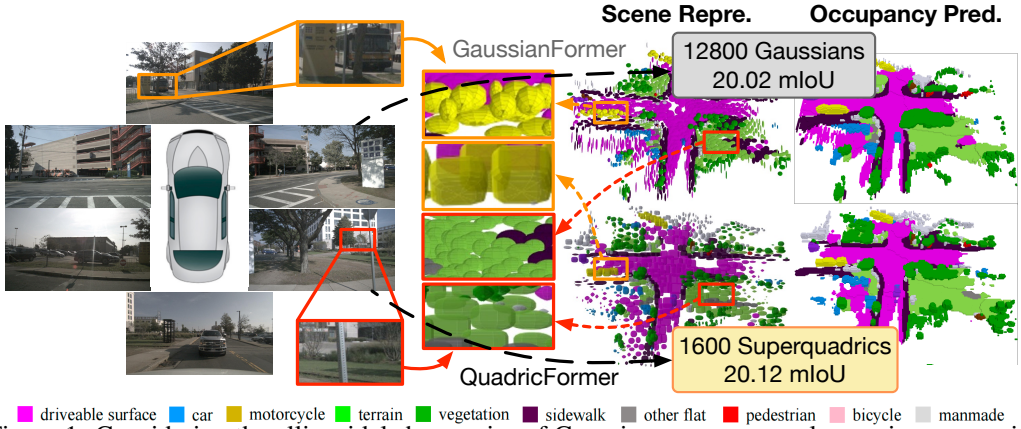


Figure 1: Considering the ellipsoidal shape prior of Gaussians, we propose leveraging expressive superquadrics to build an efficient and powerful object-centric representation. Our QuadricFormer achieves state-of-the-art performance with superior efficiency for 3D occupancy prediction.

Abstract

3D occupancy prediction is crucial for robust autonomous driving systems as it enables comprehensive perception of environmental structures and semantics. Most existing methods employ dense voxel-based scene representations, ignoring the sparsity of driving scenes and resulting in inefficiency. Recent works explore object-centric representations based on sparse Gaussians, but their ellipsoidal shape prior limits the modeling of diverse structures. In real-world driving scenes, objects exhibit rich geometries (e.g., cuboids, cylinders, and irregular shapes), necessitating excessive ellipsoidal Gaussians densely packed for accurate modeling, which leads to inefficient representations. To address this, we propose to use geometrically expressive superquadrics as scene primitives, enabling efficient representation of complex structures with fewer primitives through their inherent shape diversity. We develop a probabilistic superquadric mixture model, which interprets each superquadric as an occupancy probability distribution with a corresponding geometry prior, and calculates semantics through probabilistic mixture. Building on this, we present QuadricFormer, a superquadric-based model for efficient 3D occupancy prediction, and introduce a pruning-and-splitting module to further enhance modeling efficiency by concentrating superquadrics in occupied regions. Extensive experiments on the nuScenes and KITTI-360 datasets demonstrate that QuadricFormer achieves state-of-the-art performance while maintaining superior efficiency. Code is available at https://github.com/zuosc19/QuadricFormer.

[*]Equal contribution. [†]Corresponding author.

39th Conference on Neural Information Processing Systems (NeurIPS 2025).

1 Introduction

Vision-centric autonomous driving systems have gained much attention for their cost-effectiveness over LiDAR-based solutions [4, 17, 50, 26, 23]. However, they struggle to perceive irregularly shaped obstacles due to visual ambiguity, which compromises driving safety. Recent advances in 3D semantic occupancy prediction address this by estimating voxel-level occupancy status and semantic labels in 3D scenes [43, 44, 38, 39]. This provides a full understanding of scene structures and semantics, which enables applications including self-supervised 3D scene understanding [6, 16], 4D occupancy forecasting [51, 31, 41, 46], and end-to-end autonomous driving [14, 52].

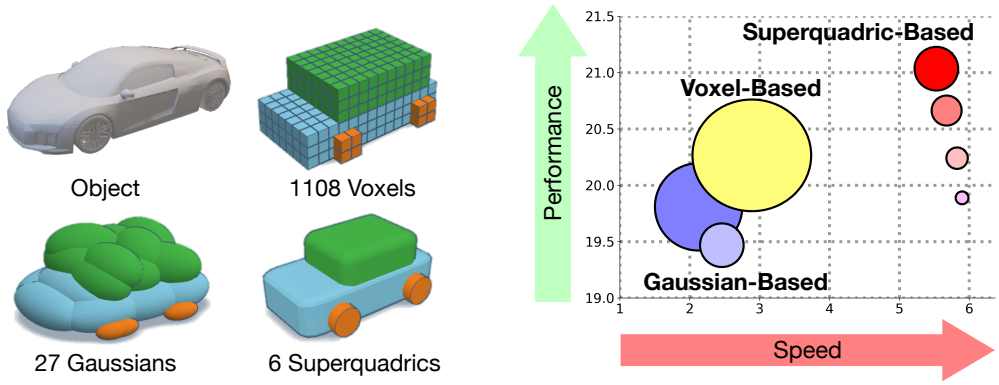
Despite promising applications, 3D semantic occupancy prediction faces efficiency challenges due to its dense 3D predictions [4, 38]. An efficient and expressive 3D representation is therefore essential. While voxel-based methods [23, 44] use dense 3D grids to capture fine details, they ignore the sparsity of driving scenes and suffer from high computational costs. Recent advances introduce object-centric representations using 3D Gaussians [18, 55] to describe scenes sparsely. Each Gaussian models the occupancy probability distribution of its local region via learnable attributes including position, covariance, opacity, and semantics. However, Gaussian representations are fundamentally limited. By their mathematical formulation, Gaussians describe the spatial occupancy probability with an ellipsoidal decay pattern. This imposes a strong ellipsoidal shape prior to Gaussians and severely constrains their capacity to model diverse geometries. Real-world driving scenarios contain objects with rich structural variations, which cannot be accurately represented by a few ellipsoidal Gaussian. Consequently, Gaussian-based models must aggregate numerous densely packed Gaussians to approximate target shapes, causing significant efficiency degradation.

In this paper, we propose an efficient and expressive object-centric 3D representation using superquadrics [1] as scene primitives. Superquadrics are a family of parameterized shapes with high geometric expressiveness and compact shape parameters, offering great flexibility in modeling diverse geometries. This allows superquadrics to model complex structures with sparse packing, enabling an efficient and powerful representation [12]. We represent scenes with a set of learnable superquadrics, each characterized by attributes including position, scale, rotation, opacity, semantics, and shape exponents. For occupancy prediction, we adopt a probabilistic superquadric mixture model that interprets each superquadric as a local occupancy probability distribution, and calculates semantics through probabilistic mixture. Building on this representation, we introduce QuadricFormer, a superquadric-based framework for efficient 3D semantic occupancy prediction. Moreover, we design a pruning-and-splitting module that concentrates superquadrics on occupied regions to further enhance modeling efficiency. Extensive experiments on the nuScenes and KITTI-360 dataset demonstrate that our QuadricFormer achieves state-of-the-art performance with superior efficiency.

2 Related Work

3D Semantic Occupancy Prediction. 3D semantic occupancy prediction reconstructs fine-grained 3D scenes by labeling each voxel with geometric and semantic information, which is critical for autonomous driving [4, 17, 38, 50, 51]. LiDAR and cameras are the two most commonly used sensors. While LiDAR-based methods excel in depth accuracy [8, 7, 10, 20, 21, 28, 33, 36, 45, 47, 48, 53, 54], their limitations in adverse weather and long-range detection motivate the vision-centric approaches, which reconstruct scenes from multi-view visual input [26, 44, 49, 4, 17]. Early approaches lifted image features directly into dense voxel grids for 3D occupancy prediction [9, 23, 44, 30]. However, given the sparsity of occupied voxels in driving scenes, subsequent works prioritized efficiency through alternative representations. Planar representations like BEV [25] and TPV [17] compress 3D data into 2D feature maps for efficient processing, but sacrifice geometric fidelity. Object-centric modeling preserves geometric fidelity by focusing computation on salient regions [18, 15, 29, 37, 55], alleviating both the redundancy of uniform voxel grids and the information loss from planar compression. However, these methods still struggle to balance efficiency and modeling capacity due to the complexity of real-world structures. To address this, we propose a superquadric-based model that achieves efficient and accurate representation of complex geometries.

Object-centric Scene Representation. Existing 3D scene representations primarily use voxel-based frameworks for fine-grained volumetric modeling [44, 23], excelling in semantic prediction tasks. However, their uniform processing of all voxels introduces spatial redundancy, particularly in sparse environments. To address this, recent works explore object-centric representations [37, 29, 18, 15, 55].



(a) Scene Representations. (b) Performance Comparison.

Figure 2: **Comparisons between different representations.** (a) Quadric-based method represents the same object with a smaller number of primitives and greater shape expressiveness. (b) Quadric-based representation outperforms existing methods in both accuracy and speed with far fewer primitives.

One line of methods partitions dense grids into localized regions, preserving only detected object areas [37, 29]. While efficient, non-empty regions may be falsely pruned, leading to irreversible loss of critical geometry. Alternatively, point-based methods use sparse points as queries for iterative refinement [34, 40]. However, points inherently lack spatial extent, limiting their ability to capture contextual geometry. Recent advances adopt 3D semantic Gaussians [18, 15, 55], where probability densities radiate from Gaussian centers to enable adaptive spatial coverage. While Gaussians mitigate point rigidity through probability spread, complex geometries often require multiple densely packed primitives, particularly for fine structures, leading to inefficient representations. In this paper, we propose geometrically expressive superquadrics as compact scene primitives. Unlike conventional object-centric methods, superquadrics natively parameterize diverse geometries (e.g., cuboids, cylinders) without dense packing, achieving superior reconstruction fidelity using fewer primitives.

Superquarics. Superquadrics are parametric geometric primitives introduced by Barr et al. [1] to model diverse shapes with compact parameterizations. A canonical superquadric is defined by five parameters: three scale parameters along each of its semi-axes and two exponents that determine its shape [19]. The scale and shape parameters of superquadrics allow for smooth interpolation between different geometric shapes, such as cuboids, cylinders, and spheres. When combined with six pose parameters for translation and rotation, a superquadric can represent a complete 3D object using only 11 parameters. Recent works employed superquadrics to decompose complex environments into compact geometric primitives [12]. These methods demonstrate compelling reconstruction capability and editing flexibility, while maintaining model efficiency. However, existing approaches operate exclusively on point clouds and are limited to object-level reconstructions. Differently, we present the first superquadric-based framework for holistic scene reconstruction directly from multi-view images, delivering state-of-the-art performance with superior efficiency.

3 Proposed Approach

In this section, we present our method based on the superquadric representation for efficient 3D semantic occupancy prediction. We first review the Gaussian-based object-centric representation and analyze its limitations (Sec 3.1). We then introduce our superquadric representation and probabilistic modeling approach for efficient occupancy prediction (Sec 3.2). Finally, we describe the overall architecture of QuadricFormer for vision-centric 3D occupancy prediction.(Sec 3.3).

3.1 Object-Centric Representation

Vision-centric 3D semantic occupancy prediction aims to estimate the occupancy status and semantic label of each voxel in 3D space based on visual inputs. Formally, given input images $\mathcal{I} = \{\mathbf{I}_i\}_{i=1}^{N}$

from N views, the model aims to predict voxel-level semantic labels $\mathbf{O} \in \mathcal{C}^{X \times Y \times Z}$ of the 3D scene, where $\mathcal{C}$ denotes the semantic classes and $X \times Y \times Z$ represents the spatial shape of occupancy.

To achieve this, voxel-based methods [44, 50] adopt dense voxel features to model 3D scenes, resulting in extremely high computational complexity of $\mathcal{O}(XYZ)$. This inefficiency stems from their uniform processing of all voxels in space, which ignores the inherent sparsity of real-world scenes. Considering this, recent works [18, 15] explore object-centric representations based on 3D Gaussians to focus computational resources on salient regions for efficient scene modeling. Gaussian-based method [15] typically employs a set of P semantic 3D Gaussian primitives $\mathcal{G} = \{\mathbf{G}_i\}_{i=1}^{P}$ to represent 3D scenes sparsely. Each Gaussian $\mathbf{G}_i$ models a flexible local region with its explicit mean $\mathbf{m}_i$, scale $\mathbf{s}_i$, rotation $\mathbf{r}_i$, opacity a_i, and semantic probability $\mathbf{c}_i$. For a point $\mathbf{x}$ in 3D space, its geometric occupancy probability associated with the Gaussian $\mathbf{G}$ is:

$$\alpha(\mathbf{x}; \mathbf{G}) = \exp\big(-\frac{1}{2}(\mathbf{x}-\mathbf{m})^{\mathrm{T}} \mathbf{\Sigma}^{-1} (\mathbf{x}-\mathbf{m}) \big), \tag{1}$$

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

where $\mathbf{x}$ denotes the point position, and $\mathbf{\Sigma}$, $\mathbf{R}$, $\mathbf{S}$ represent the covariance matrix, the rotation matrix constructed from the quaternion $\mathbf{r}$, and the diagonal scale matrix from the scale $\mathbf{s}$. Furthermore, a probabilistic Gaussian mixture model is used to aggregate multiple Gaussians for predicting the structure and semantics of the scene. As each Gaussian represents a flexible region of the scene, the Gaussian-based representation enables adaptive allocation of resources and efficient modeling.

Although 3D Gaussian representation is more efficient than dense voxels (e.g., 6400 Gaussians vs. $200 \times 200 \times 16$ voxels per scene), it still exhibits limitations that prevent an optimal efficiency-performance balance. *Our key insight is that Gaussians inherently impose an ellipsoidal shape prior, which limits their ability to model diverse structures*. As shown in Eq. 1, the occupancy probability distribution of the Gaussian $\mathbf{G}$ can be viewed as a set of iso-probability surfaces defined by:

$$g(\mathbf{x}) = -\frac{1}{2}\big((\frac{x}{s_x})^2 + (\frac{y}{s_y})^2 + (\frac{z}{s_z})^2\big) = k, \tag{3}$$

where $\mathbf{x} = (x, y, z)^T$ denotes the point position, k denotes the hyperparameter of the surface family, and $\mathbf{s} = (s_x, s_y, s_z)^T$ represents the Gaussian's scales along three axes. The rotation and mean of the Gaussian are omitted for simplicity in Eq. 3, which describes a standard ellipsoid. Each Gaussian then models occupancy probability with an ellipsoidal decay in 3D space. But real-world objects often have diverse shapes, such as cuboids, cylinders, and irregular shapes, which cannot be accurately represented by a few ellipsoidal Gaussians. This forces the model to use numerous densely packed Gaussians to approximate complex structures, leading to inefficient scene representations. In contrast, our method employs expressive superquadrics as scene primitives, enabling efficient and compact modeling of complex structures with only a few sparsely packed superquadrics.

3.2 Scene as Superquadrics

We introduce an object-centric scene representation leveraging superquadric primitives for their efficiency and expressive power. Superquadrics are a parametric shape family with strong geometric expressiveness, defined as follows:

$$f(\mathbf{x}) = \left(\left(\frac{x}{s_x} \right)^{\frac{2}{\epsilon_2}} + \left(\frac{y}{s_y} \right)^{\frac{2}{\epsilon_2}} \right)^{\frac{\epsilon_2}{\epsilon_1}} + \left(\frac{z}{s_z} \right)^{\frac{2}{\epsilon_2}} = k, \tag{4}$$

where $\mathbf{x} = (x, y, z)^T$ denotes the point position, and k denotes the hyperparameter of the surface family. Compared to the ellipsoids in Eq. 3, superquadrics introduce only two additional shape-defining exponents ϵ_1, ϵ_2 yet can represent a much wider variety of shapes. As shown in Fig. 2a, superquadrics allow for continuous and diverse shape variations as the shape parameters change. This inherent parameter efficiency and geometric expressiveness enable superquadrics to model diverse shapes without being densely packed. Consequently, only a small number of superquadrics are needed to represent complex scene structures, achieving an efficient yet powerful scene representation.

We thus utilize a set of P parameterized superquadrics $\mathcal{Q} = \{\mathbf{Q}_i\}_{i=1}^{P}$ to represent the 3D scene. Each superquadric is characterized by its scale $\mathbf{s}$ and shape exponents ϵ_1, ϵ_2 to define its geometry. To extend the representation to the global coordinate system, each primitive is also assigned a

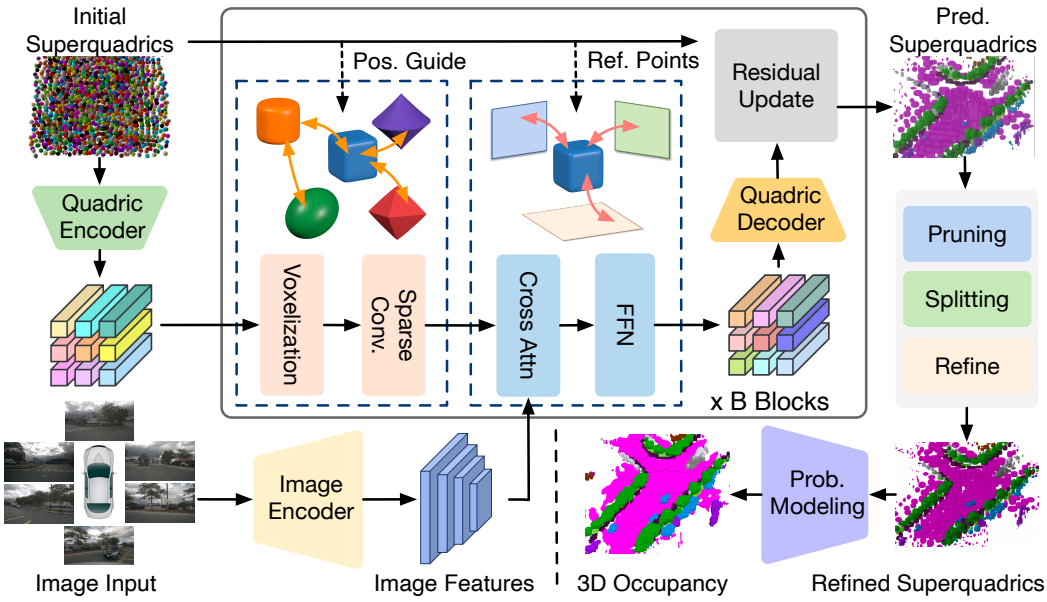


Figure 3: **Overall Framework of QuadricFormer.** We use several quadric-encoder blocks to update superquadrics, and employ a pruning-and-splitting module to further enhance modeling efficiency.

position $\mathbf{x}$ and rotation $\mathbf{r}$. Beyond geometric attributes, each superquadric is further equipped with an opacity a and a semantic probability $\mathbf{c}$ to incorporate semantic information. In summary, our superquadric-based representation can be formulated as:

$$\mathcal{Q} = \{\mathbf{Q}_i\}_{i=1}^{P} = \{(\mathbf{x}_i, \mathbf{s}_i, \mathbf{r}_i, a_i, \epsilon_{1;i}, \epsilon_{2;i}, \mathbf{c}_i,)\}_{i=1}^{P}. \tag{5}$$

We now explore obtaining 3D occupancy prediction from the superquadric representation. Existing methods [12] typically treat superquadrics as deterministic surfaces, fitting them to object parts for point cloud reconstruction. However, these surface-based approaches face key limitations in vision-centric occupancy prediction. A primary challenge is supervision. While point cloud reconstruction directly optimizes the distance between points and the superquadric surfaces, occupancy prediction requires fine-grained scene understanding, which lacks clear surface-based constraints. Furthermore, surface-based methods rely on the explicit structure from point cloud inputs, whereas visual inputs introduce structural uncertainty, making deterministic modeling unstable. Lastly, surface-based methods focus on object-level reconstruction with simple spatial relationships. But real-world driving scenes involve far more complex surface interactions, posing significant modeling difficulties.

To achieve robust 3D semantic occupancy prediction, we design a probabilistic modeling mechanism that converts superquadrics into occupancy probabilities. Inspired by GaussianFormer-2 [15], we adopt a probabilistic superquadric mixture model, where each superquadric defines the occupancy probability distribution in its local neighborhood. To compute the probability of a 3D point $\mathbf{x}$ being occupied by the superquadric $\mathbf{Q}$, we first transform $\mathbf{x}$ into $\mathbf{Q}$'s local coordinate system, defined by its position $\mathbf{m}$ and rotation $\mathbf{r}$:

$$\mathbf{x_Q} = \mathbf{R}(\mathbf{x} - \mathbf{m}), \tag{6}$$

where $\mathbf{x_Q}$ denotes the local coordinate of $\mathbf{x}$, and $\mathbf{R}$ denotes the rotation matrix constructed from the rotation $\mathbf{r}$. The occupancy probability of $\mathbf{x}$ associated with $\mathbf{Q}$ is then computed as:

$$p_{\mathbf{o}}(\mathbf{x}; \mathbf{G}) = \exp\big(-\mathrm{f}(\mathbf{x_Q})\big) = \exp\left(\left(\left(\frac{\mathrm{x_Q}}{\mathrm{s_x}}\right)^{\frac{2}{\epsilon_2}} + \left(\frac{\mathrm{y_Q}}{\mathrm{s_y}}\right)^{\frac{2}{\epsilon_2}}\right)^{\frac{\epsilon_2}{\epsilon_1}} + \left(\frac{\mathrm{z_Q}}{\mathrm{s_z}}\right)^{\frac{2}{\epsilon_2}}\right), \tag{7}$$

where $\mathbf{x_Q} = (x_\mathbf{Q}, y_\mathbf{Q}, z_\mathbf{Q})^T$ and $\mathbf{s} = (s_x, y_x, z_s)^T$ are the position and scale parameters, respectively, and ϵ_1, ϵ_2 are the shape exponents of the superquadric $\mathbf{Q}$. Assuming conditional independence of occupancy among different superquadrics, the final occupancy probability at $\mathbf{x}$ is computed as:

$$p_{\mathbf{o}}(\mathbf{x}) = 1 - \prod_{i=1}^{P} \big(1 - p_{\mathbf{o}}(\mathbf{x}; \mathbf{Q}_i)\big). \tag{8}$$

Semantic predictions are subsequently inferred by a weighted aggregation of semantic probabilities from all contributing superquadrics, where weights correspond to their occupancy influence at $\mathbf{x}$:

$$p_{\mathbf{c}}(\mathbf{x}) = \frac{\sum_{i=1}^{P} p_{\mathbf{o}}(\mathbf{x}|\mathbf{Q}_i) a_i \mathbf{c}_i}{\sum_{j=1}^{P} p_{\mathbf{o}}(\mathbf{x}|\mathbf{Q}_j) a_j}, \tag{9}$$

The key to this probabilistic modeling is incorporating the superquadric geometry as shape priors within the probability distribution, realized as iso-probability surfaces conforming to its geometry in Eq. 4. Leveraging the geometrically expressive power of superquadrics, our model can efficiently represent complex 3D structures using a sparse set of primitives without dense packing, achieving an efficient yet powerful scene representation. Moreover, this probabilistic framework effectively models structural uncertainties arising from visual ambiguities, significantly improving the model's robustness and generalization capabilities.

3.3 QuadricFormer

We present the overall framework of QuadricFormer in Fig. 3. Starting from the image inputs of N views $\mathcal{I} = \{\mathbf{I}_i\}_{i=1}^{N}$, we first employ an image backbone $E_{\mathbf{I}}$ to extract multi-scale image features $\mathbf{F}_{\mathbf{I}}$:

$$\mathbf{F}_{\mathbf{I}} = E_{\mathbf{I}}(\mathcal{I}), \tag{10}$$

Due to the lack of any structural prior of the scene, we randomly initialize a few superquadrics $\mathbf{Q}_{init}$ in 3D space, and use B quadric-encoder blocks $E_{\mathbf{B}}$ to predict the final superquadrics from images. In each block, we first encode current superquadrics $\mathbf{Q}_i$ into features $\mathbf{F}_{\mathbf{Q}}$ via a quadric encoder $E_{\mathbf{Q}}$:

$$\mathbf{F}_{\mathbf{Q}} = E_{\mathbf{Q}}(\mathbf{Q}_i). \tag{11}$$

We then use 3D sparse convolution E_{conv} for superquadric feature self-encoding and deformable attention E_{attn} for interaction between superquadric and image features:

$$\mathbf{F}_{\mathbf{Q}} = E_{conv}(\mathbf{F}_{\mathbf{Q}}, \mathbf{x}_{\mathbf{Q}}), \mathbf{F}_{\mathbf{Q}} = E_{attn}(\mathbf{F}_{\mathbf{Q}}, \mathbf{x}_{\mathbf{Q}}, \mathbf{F}_{\mathbf{I}}), \tag{12}$$

where $\mathbf{x}_{\mathbf{Q}}$ denotes the explicit position of the superquadric $\mathbf{Q}$, serving as auxiliary information to guide feature encoding. Finally, a quadric decoder $D_{\mathbf{Q}}$ is used to predict the update of superquadric attributes $\Delta\mathbf{Q}$, which are combined with the original attributes $\mathbf{Q}$ via residual addition:

$$\Delta\mathbf{Q} = D_{\mathbf{Q}}(\mathbf{F}_{\mathbf{Q}}), \mathbf{Q}_{i+1} = \mathbf{Q}_i + \Delta\mathbf{Q}. \tag{13}$$

After B blocks update, we get the final superquadric prediction $\mathbf{Q}$, and the 3D semantic occupancy prediction $\mathbf{O} \in \mathcal{C}^{X \times Y \times Z}$ can be inferred through the probabilistic modeling mechanism:

$$\mathbf{O} = Prob(\mathbf{Q}). \tag{14}$$

For optimization, we adopt the cross entropy loss and the lovaszsoftmax [2] loss for training.

Due to the lack of structural priors, superquadrics are uniformly initialized in 3D space. As a result, some superquadrics in empty regions are optimized to small scales and contribute little to scene modeling, which leads to inefficiency. To address this, we introduce a pruning-splitting module after initial training. Small-scale superquadrics (likely in empty regions) are pruned, while large-scale ones (likely in occupied regions) are split for finer modeling. We keep the number of superquadrics unchanged and use two additional blocks to further refine their properties. Notably, this lightweight module improves superquadric utilization for more efficient scene representation without introducing significant computational overhead.

4 Experiments

4.1 Datasets and Metrics

nuScenes [3] comprises 1,000 urban driving sequences collected in Boston and Singapore. The dataset is officially split into 700 sequences for training, 150 for validation, and 150 for testing. Each sequence spans a duration of 20 seconds with RGB images captured by 6 surrounding cameras, and the key frames are annotated at a 2 Hz frequency. For supervision and evaluation, we leverage the dense semantic occupancy annotations from SurroundOcc. The annotated voxel grid extends from

Table 1: **3D semantic occupancy prediction results on nuScenes.** * means supervised by dense occupancy annotations as opposed to original LiDAR segmentation labels. Sq. denotes the number of Superquadrics in our model. Our method achieves state-of-the-art performance.

Method	IoU	mIoU	barrier	bicycle	bus	car	const. veh.	motorcycle	pedestrian	traffic cone	trailer	truck	drive. suf.	other flat	sidewalk	terrain	manmade	vegetation
MonoScene [5]	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 [32]	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 [25]	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 [17]	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* [17]	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 [50]	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 [44]	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
GaussianFormer [18]	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
GaussianFormer-2 [15]	31.74	20.82	**21.39**	**13.44**	**28.49**	30.82	10.92	15.84	13.55	10.53	14.04	**22.92**	40.61	24.36	26.08	24.27	13.83	21.98
QuadricFormer (1600 Sq.)	31.22	20.12	19.58	13.11	27.27	29.64	11.25	16.26	12.65	9.15	12.51	21.24	40.20	24.34	25.69	24.24	12.95	21.86
QuadricFormer (12800 Sq.)	**32.13**	**21.11**	21.38	13.41	28.40	**31.01**	**11.32**	**17.10**	13.94	11.28	**14.75**	22.66	**40.81**	**24.71**	**26.51**	**25.22**	13.54	21.78

Table 2: **Monocular 3D semantic occupancy prediction results on SSCBench-KITTI-360.** Num of Prims. denotes the number of primitives in the model. Our method achieves comparable performance to GaussianFormer-2 [15] with much fewer primitives.

Method	Input	Num of Prims.	IoU	mIoU	car	bicycle	motorcycle	truck	other-veh.	person	road	parking	sidewalk	other-grnd	building	fence	vegetation	terrain	pole	traf.-sign	other-struct.	other-object
LMSCNet [33]	L	-	47.53	13.65	20.91	0	0	0.26	0	0	62.95	13.51	33.51	0.2	43.67	0.33	40.01	26.80	0	0	3.63	0
SSCNet [35]	L	-	53.58	16.95	31.95	0	0.17	10.29	0.58	0.07	65.7	17.33	41.24	3.22	44.41	6.77	43.72	28.87	0.78	0.75	8.60	0.67
MonoScene [5]	C	262144	37.87	12.31	19.34	0.43	0.58	8.02	2.03	0.86	48.35	11.38	28.13	3.22	32.89	3.53	26.15	16.75	6.92	5.67	4.20	3.09
Voxformer [24]	C	262144	38.76	11.91	17.84	1.16	0.89	4.56	2.06	1.63	47.01	9.67	27.21	2.89	31.18	4.97	28.99	14.69	6.51	6.92	3.79	2.43
TPVFormer [17]	C	81920	40.22	13.64	21.56	1.09	1.37	8.06	2.57	2.38	52.99	11.99	31.07	3.78	34.83	4.80	30.08	17.51	7.46	5.86	5.48	2.70
OccFormer [50]	C	262144	**40.27**	13.81	**22.58**	0.66	0.26	9.89	3.82	2.77	**54.30**	13.44	31.53	3.55	**36.42**	4.80	31.00	**19.51**	**7.77**	**8.51**	6.95	4.60
GaussianFormer [18]	C	38400	35.38	12.92	18.93	1.02	**4.62**	**18.07**	7.59	3.35	45.47	10.89	25.03	**5.32**	28.44	5.68	29.54	8.62	2.99	2.32	9.51	**5.14**
GaussianFormer-2 [15]	C	38400	38.37	**13.90**	21.08	**2.55**	4.21	12.41	5.73	1.59	54.12	11.04	**32.31**	3.34	32.01	4.98	28.94	17.33	3.57	5.48	5.88	3.54
QuadricFormer	C	**12800**	38.89	13.63	18.80	1.31	4.43	16.57	**8.57**	**3.44**	45.49	**13.77**	25.92	5.25	29.73	**6.73**	**31.95**	9.13	4.47	4.02	**10.88**	4.88

-50m to 50m along both the X and Y axes, and from -5m to 3m along the Z axis, with a spatial resolution of $200 \times 200 \times 16$. Each voxel is classified into one of the 18 categories(16 semantics, 1 empty and 1 unknown).

KITTI-360 [27] comprises over 320k images collected in suburban driving scenes with comprehensive 360° sensory coverage, including two perspective cameras, two fisheye cameras, a Velodyne LiDAR, and a laser scanner. In our experiments, we use the RGB images from the left perspective camera of the ego vehicle as model input. For supervision and evaluation, we adopt the semantic occupancy annotations provided by SSCBench-KITTI-360 [22]. The official split contains 7 sequences for training, 1 for validation, and 1 for testing, corresponding to 8487, 1812, and 2566 key frames, respectively. The annotated voxel grid covers a region of $51.2 \times 51.2 \times 6.4\ \mathrm{m}^3$ in front of the ego vehicle, with a spatial resolution of $256 \times 256 \times 32$. Each voxel is categorized into one of 19 classes (18 semantic classes and 1 empty).

The evaluation metrics adhere to common practice, namely mean Intersection-over-Union (mIoU) and Intersection-over-Union (IoU):

$$\mathbf{mIoU} = \frac{1}{|\mathcal{C}'|} \sum_{i \in \mathcal{C}'} \frac{TP_i}{TP_i + FP_i + FN_i}, \tag{15}$$

$$\mathbf{IoU} = \frac{TP_{\neq c_0}}{TP_{\neq c_0} + FP_{\neq c_0} + FN_{\neq c_0}}, \tag{16}$$

Where $\mathcal{C}'$, c_0, TP, FP, and FN represent the non-empty classes, the empty class, and the number of true positive, false positive, and false negative predictions, respectively.

4.2 Implementation Details

The input images are at resolutions of 900×1600 for nuScenes and 376×1408 for KITTI-360 [27] with random flipping and photometric distortion augmentations. We employ ResNet101-DCN [13] with FCOS3D checkpoint [42] for nuScenes [3], and ResNet50 [13] pretrained on ImageNet [11]

Table 3: **Performance and efficiency comparison with Gaussian-based methods.** The latency and memory are tested on an NVIDIA 4090 GPU with batch size one during inference, in accordance with Gaussian-based methods [18, 15]. Our method achieves better performance-efficiency trade-off.

Method	Number of Primitives	Latency (ms)	Memory (MB)	mIoU	IoU
GaussianFormer [18]	25600	227	4850	16.00	28.72
	144000	372	6229	19.10	29.83
GaussianFormer-2 [15] (Depth Initialized)	1600	341	3075	18.73	28.99
	3200	355	3076	18.75	29.64
	6400	395	3652	19.55	30.37
	12800	451	4535	19.69	30.43
QuadricFormer (Ours)	1600	**162**	**2554**	20.04	30.71
	3200	164	2556	20.35	31.62
	6400	165	2560	20.79	31.89
	12800	179	2563	**21.11**	**32.13**

for KITTI-360 [27]. The numbers of Superquarics are set to 1600 in our main results for nuScenes and KITTI-360. For optimization, we train our model using AdamW with weight decay of 0.01, and maximum learning rate of 4×10^{-4}, which decays with a cosine schedule. We train our model for 20 epochs on nuScenes and KITTI-360 with a batch of 8.

4.3 Main Results

3D Semantic Occupancy Prediction. We report the performance of our QuadricFormer on nuScenes dataset [3] in Table 1. Compared to other methods, our approach achieves state-of-the-art performance. Specifically, QuadricFormer outperforms other methods on categories such as bicycle, motorcycle, truck and various ground-related classes (drivable surface, sidewalk, terrain, etc.), demonstrating superior capability in modeling both small and structural objects. Moreover, our method significantly surpasses GaussianFormer-2 [15] while using substantially fewer superquadrics (1600 vs. 12800), further validating its efficiency and effectiveness. Furthermore, We report the results for monocular 3D semantic occupancy prediction on SSCBench-KITTI-360 [22] in Table 2. Our method achieves comparable mIoU performance to GaussianFormer-2 [15], demonstrating the effectiveness of our approach for monocular 3D semantic occupancy prediction.

Performance and Efficiency Comparison with Gaussian-based Methods. We report the performance and efficiency comparison for QuadricFormer with Gaussian-based methods on nuScenes in Table 3. QuadricFormer consistently outperforms prior methods in both 3D semantic occupancy prediction and computational efficiency. Specifically, our method achieves the highest mIoU (up to 21.11) and IoU (up to 32.13), surpassing all Gaussian-based approaches. In terms of efficiency, QuadricFormer significantly reduces both latency and memory usage. For similar or even fewer primitives (e.g., 1600 or 3200), our method achieves a latency as low as 162 ms and 2554 MB memory consumption, which are substantially lower than others. Notably, even when increasing the number of primitives in QuadricFormer to 12800, both latency and memory usage remain lower than those of Gaussian-based methods using only 1600 primitives. This further highlights the superior efficiency of our approach for the complex structures in real-world applications.

4.4 Ablation Study

Effect of the ϵ Range. We conduct ablation study on the range of the superquadric exponent parameters ϵ in Eq. 4, as reported in Table 4. We set the number of superquadrics to 12800 for these experiments. The table explores the effect of different ϵ ranges on 3D semantic occupancy prediction performance. We observe that setting the range of $(0.1, 2)$ yields the best results, achieving the highest mIoU (20.51) and IoU (31.25).

Effect of the Pruning-splitting Module. We conduct ablation studies on the effect of the pruning-splitting module, as shown in Table 5. The results demonstrate that increasing the crop & split number consistently improves performance. This confirms that reallocating primitives from low to high occupancy regions effectively enhances the accuracy and efficiency of our 3D scene representation.

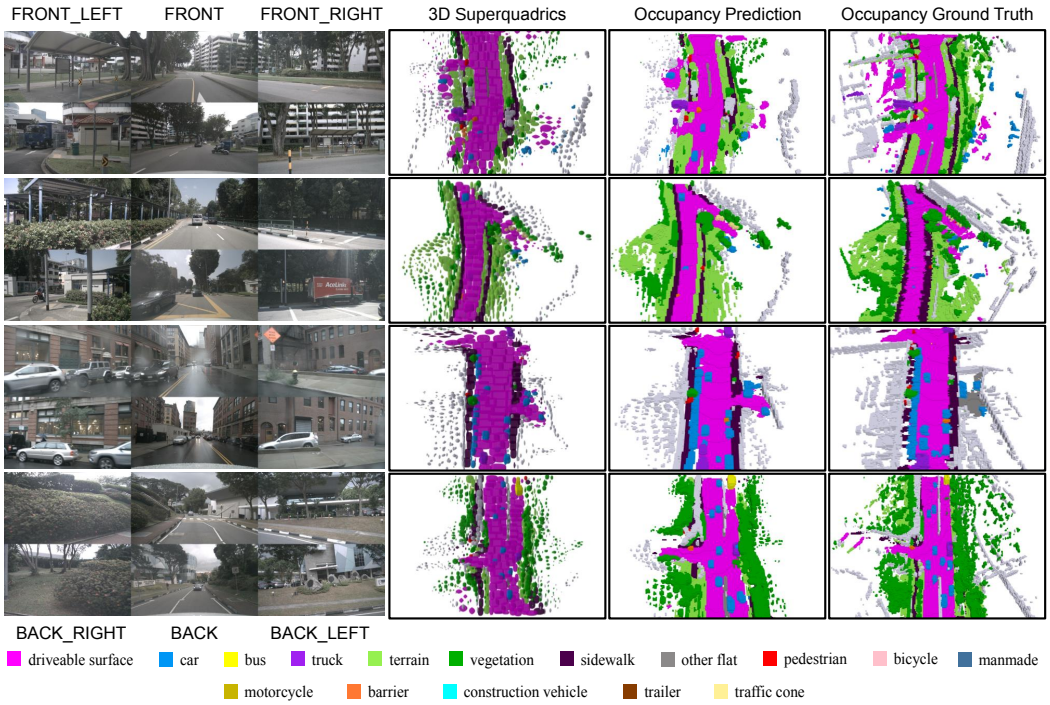


Figure 4: **3D Superquadrics and occupancy visualizations on nuScenes.** Our model is able to predict high-fidelity shapes and achieves comprehensive occupancy results.

Table 4: Effect of the ϵ range.

Range of ϵ	mIoU	IoU
$(0.01, 2)$	20.39	31.13
$(0.01, 5)$	20.25	30.63
$(0.1, 2)$	**20.51**	**31.25**
$(0.1, 5)$	19.86	30.65

Table 5: Effect of the pruning-splitting module.

Crop & Split Number	mIoU	IoU
0	19.41	39.77
200	19.65	30.35
400	19.90	30.67
800	**20.12**	**31.22**

4.5 Visualizations

We present visualizations of the predicted superquadrics and occupancy results in Figure 4. Our model is able to predict high-fidelity shapes using superquadrics and achieves comprehensive occupancy results. Further, we compare our method against GaussianFormer-2 [15] in Figure 5, showing that our predicted superquadrics offer more adaptive shapes than Gaussians. Moreover, our method achieves high-quality performance using only 1600 superquadrics, compared to 6400 Gaussians. Figure 6 shows a sample for 3D semantic occupancy prediction on the nuScenes [3] validation set. Compared to GaussianFormer-2 [15], our QuadricFormer exhibits enhanced modeling capability for complex objects and road surfaces.

5 Conclusion

In this paper, we have proposed a superquadric-based object-centric representation for efficient 3D semantic occupancy prediction. Specifically, we leverage the geometric expressiveness of superquadrics to model complex structures with far fewer sparsely packed primitives. We formulate a probabilistic superquadric mixture model, where each superquadric encodes an occupancy probability distribution with a corresponding geometry prior, and semantics are inferred via probabilistic mixture. Furthermore, we introduce a pruning-and-splitting module that adaptively concentrates superquadrics in occupied regions to further enhance modeling efficiency. Our proposed QuadricFormer demonstrates state-of-the-art performance and superior efficiency on the nuScenes benchmark, providing an effective and compact solution for scene understanding in vision-centric autonomous driving systems.

Limitations. With random initialization, QuadricFormer cannot fully learn accurate superquadric positions, leaving some superquadrics in empty regions and reducing representation efficiency.

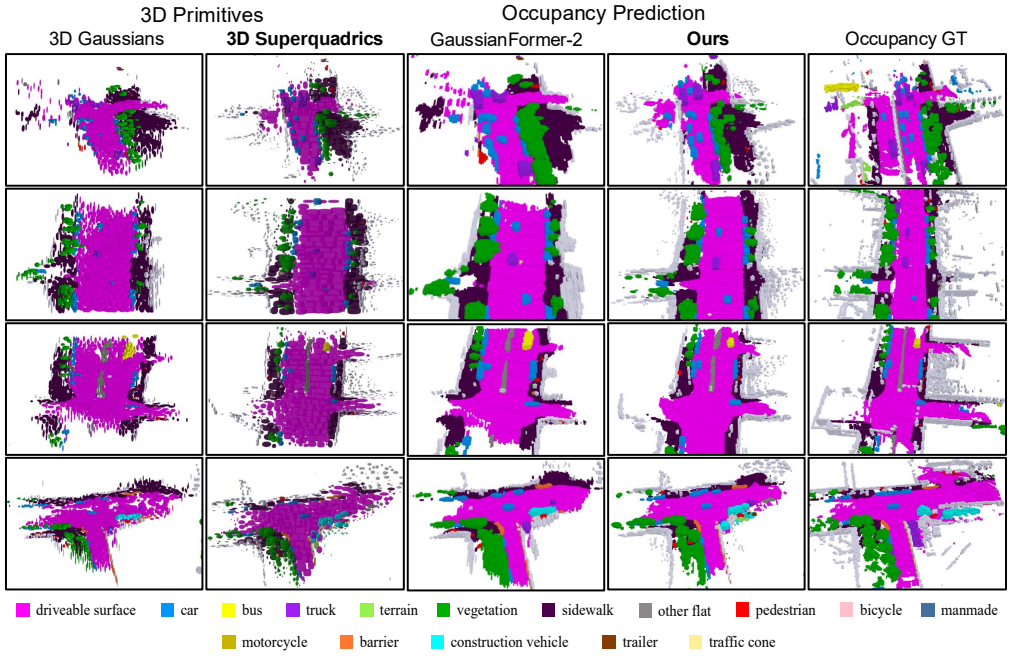


Figure 5: **Qualitative comparisons.** QuadricFormer predicts more flexible and adaptive shapes.

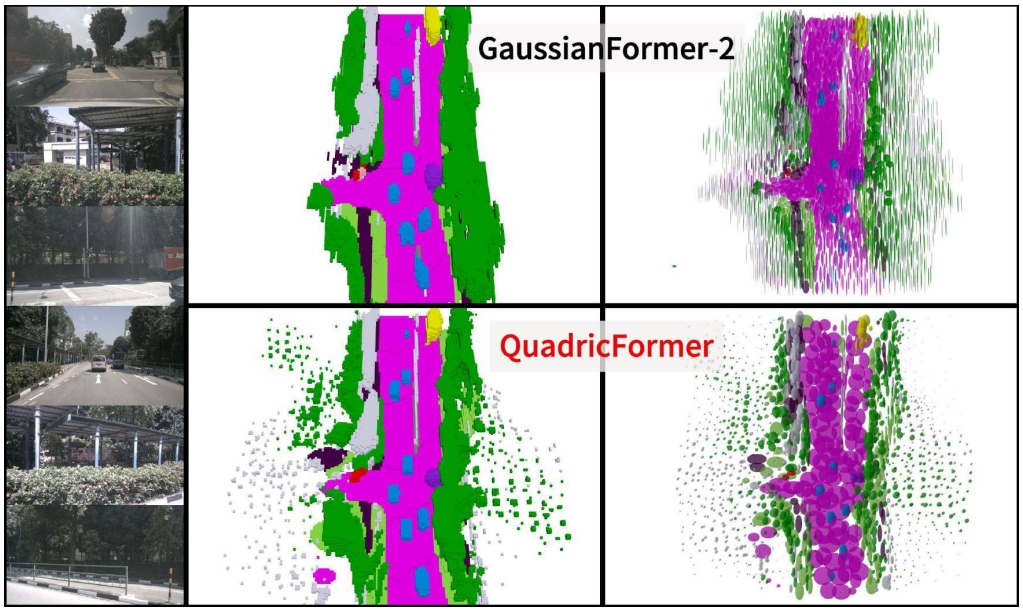


Figure 6: **Visualizations of the proposed QuadricFormer compared to GaussianFormer-2 [15] for 3D semantic occupancy prediction on the nuScenes [3] validation set.** We visualize the six surrounding camera inputs, the corresponding occupancy prediction results, and the primitive representations. The upper row shows the predicted occupancy (left) and the primitive representation (right) by GaussianFormer-2. The lower row shows the prediction results of QuadricFormer.

Broader impact. Our work on autonomous driving has the potential to improve traffic efficiency in the future, but it may also contribute to job displacement for drivers.

Acknowledgements

This work was supported in part by the National Natural Science Foundation of China under Grant 62576188, Grant 62336004, Grant 62321005, and Grant 62125603, in part by the Beijing Natural Science Foundation under Grant L247009, and in part by the Beijing National Research Center for Information Science and Technology.

References

[1] Alan H Barr. Superquadrics and angle-preserving transformations. *IEEE Computer graphics and Applications*, 1(01):11–23, 1981. 2, 3

[2] 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 *CVPR*, pages 4413–4421, 2018. 6

[3] 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 *CVPR*, 2020. 6, 7, 8, 9, 10

[4] Anh-Quan Cao and Raoul de Charette. Monoscene: Monocular 3d semantic scene completion. In *CVPR*, pages 3991–4001, 2022. 2

[5] 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. 7

[6] Anh-Quan Cao and Raoul de Charette. Scenerf: Self-supervised monocular 3d scene reconstruction with radiance fields. In *ICCV*, pages 9387–9398, 2023. 2

[7] Xiaokang Chen, Kwan-Yee Lin, Chen Qian, Gang Zeng, and Hongsheng Li. 3d sketch-aware semantic scene completion via semi-supervised structure prior. In *CVPR*, pages 4193–4202, 2020. 2

[8] Xiaozhi Chen, Huimin Ma, Ji Wan, Bo Li, and Tian Xia. Multi-view 3d object detection network for autonomous driving. In *CVPR*, 2017. 2

[9] Yueh-Tung Chen, Martin Garbade, and Juergen Gall. 3d semantic scene completion from a single depth image using adversarial training. In *2019 IEEE International Conference on Image Processing (ICIP)*, pages 1835–1839. IEEE, 2019. 2

[10] Ran Cheng, Ryan Razani, Ehsan Taghavi, Enxu Li, and Bingbing Liu. 2-s3net: Attentive feature fusion with adaptive feature selection for sparse semantic segmentation network. In *CVPR*, pages 12547–12556, 2021. 2

[11] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009. 7

[12] Elisabetta Fedele, Boyang Sun, Leonidas Guibas, Marc Pollefeys, and Francis Engelmann. Superdec: 3d scene decomposition with superquadric primitives. *arXiv preprint arXiv:2504.00992*, 2025. 2, 3, 5

[13] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, pages 770–778, 2016. 7

[14] Yihan Hu, Jiazhi Yang, Li Chen, Keyu Li, Chonghao Sima, Xizhou Zhu, Siqi Chai, Senyao Du, Tianwei Lin, Wenhai Wang, et al. Planning-oriented autonomous driving. In *CVPR*, pages 17853–17862, 2023. 2

[15] 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. 2, 3, 4, 5, 7, 8, 9, 10

[16] Yuanhui Huang, Wenzhao Zheng, Borui Zhang, Jie Zhou, and Jiwen Lu. Selfocc: Self-supervised vision-based 3d occupancy prediction. In *CVPR*, pages 19946–19956, 2024. 2

[17] Yuanhui Huang, Wenzhao Zheng, Yunpeng Zhang, Jie Zhou, and Jiwen Lu. Tri-perspective view for vision-based 3d semantic occupancy prediction. In *CVPR*, pages 9223–9232, 2023. 2, 7

[18] Yuanhui Huang, Wenzhao Zheng, Yunpeng Zhang, Jie Zhou, and Jiwen Lu. Gaussianformer: Scene as gaussians for vision-based 3d semantic occupancy prediction. *arXiv preprint arXiv:2405.17429*, 2024. 2, 3, 4, 7, 8

[19] Ales Jaklic, Ales Leonardis, and Franc Solina. *Segmentation and recovery of superquadrics*, volume 20. Springer Science & Business Media, 2000. 3

[20] Alex H Lang, Sourabh Vora, Holger Caesar, Lubing Zhou, Jiong Yang, and Oscar Beijbom. Pointpillars: Fast encoders for object detection from point clouds. In *CVPR*, 2019. 2

[21] Jie Li, Kai Han, Peng Wang, Yu Liu, and Xia Yuan. Anisotropic convolutional networks for 3d semantic scene completion. In *CVPR*, pages 3351–3359, 2020. 2

[22] Yiming Li, Sihang Li, Xinhao Liu, Moonjun Gong, Kenan Li, Nuo Chen, Zijun Wang, Zhiheng Li, Tao Jiang, Fisher Yu, et al. Sscbench: A large-scale 3d semantic scene completion benchmark for autonomous driving. In *2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 13333–13340. IEEE, 2024. 7, 8

[23] Yiming Li, Zhiding Yu, Christopher Choy, Chaowei Xiao, Jose M Alvarez, Sanja Fidler, Chen Feng, and Anima Anandkumar. Voxformer: Sparse voxel transformer for camera-based 3d semantic scene completion. In *CVPR*, pages 9087–9098, 2023. 2

[24] Yiming Li, Zhiding Yu, Christopher Choy, Chaowei Xiao, Jose M Alvarez, Sanja Fidler, Chen Feng, and Anima Anandkumar. Voxformer: Sparse voxel transformer for camera-based 3d semantic scene completion. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9087–9098, 2023. 7

[25] 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. In *ECCV*, 2022. 2, 7

[26] Zhiqi Li, Zhiding Yu, David Austin, Mingsheng Fang, Shiyi Lan, Jan Kautz, and Jose M Alvarez. Fb-occ: 3d occupancy prediction based on forward-backward view transformation. *arXiv preprint arXiv:2307.01492*, 2023. 2

[27] Yiyi Liao, Jun Xie, and Andreas Geiger. Kitti-360: A novel dataset and benchmarks for urban scene understanding in 2d and 3d. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(3):3292–3310, 2022. 7, 8

[28] Venice Erin Liong, Thi Ngoc Tho Nguyen, Sergi Widjaja, Dhananjai Sharma, and Zhuang Jie Chong. Amvnet: Assertion-based multi-view fusion network for lidar semantic segmentation. *arXiv preprint arXiv:2012.04934*, 2020. 2

[29] Yuhang Lu, Xinge Zhu, Tai Wang, and Yuexin Ma. Octreeocc: Efficient and multi-granularity occupancy prediction using octree queries. *arXiv preprint arXiv:2312.03774*, 2023. 2, 3

[30] Qihang Ma, Xin Tan, Yanyun Qu, Lizhuang Ma, Zhizhong Zhang, and Yuan Xie. Cotr: Compact occupancy transformer for vision-based 3d occupancy prediction. In *CVPR*, pages 19936–19945, 2024. 2

[31] Chen Min, Dawei Zhao, Liang Xiao, Jian Zhao, Xinli Xu, Zheng Zhu, Lei Jin, Jianshu Li, Yulan Guo, Junliang Xing, et al. Driveworld: 4d pre-trained scene understanding via world models for autonomous driving. In *CVPR*, pages 15522–15533, 2024. 2

[32] Zak Murez, Tarrence van As, James Bartolozzi, Ayan Sinha, Vijay Badrinarayanan, and Andrew Rabinovich. Atlas: End-to-end 3d scene reconstruction from posed images. In *ECCV*, pages 414–431, 2020. 7

[33] Luis Roldao, Raoul de Charette, and Anne Verroust-Blondet. Lmscnet: Lightweight multiscale 3d semantic completion. In *2020 International Conference on 3D Vision (3DV)*, pages 111–119, 2020. 2, 7

[34] Yiang Shi, Tianheng Cheng, Qian Zhang, Wenyu Liu, and Xinggang Wang. Occupancy as set of points. *arXiv preprint arXiv:2407.04049*, 2024. 3

[35] Shuran Song, Fisher Yu, Andy Zeng, Angel X Chang, Manolis Savva, and Thomas Funkhouser. Semantic scene completion from a single depth image. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1746–1754, 2017. 7

[36] Haotian Tang, Zhijian Liu, Shengyu Zhao, Yujun Lin, Ji Lin, Hanrui Wang, and Song Han. Searching efficient 3d architectures with sparse point-voxel convolution. In *ECCV*, pages 685–702, 2020. 2

[37] Pin Tang, Zhongdao Wang, Guoqing Wang, Jilai Zheng, Xiangxuan Ren, Bailan Feng, and Chao Ma. Sparseocc: Rethinking sparse latent representation for vision-based semantic occupancy prediction. In *CVPR*, pages 15035–15044, 2024. 2, 3

[38] Xiaoyu Tian, Tao Jiang, Longfei Yun, Yue Wang, Yilun Wang, and Hang Zhao. Occ3d: A large-scale 3d occupancy prediction benchmark for autonomous driving. *arXiv preprint arXiv:2304.14365*, 2023. 2

[39] Wenwen Tong, Chonghao Sima, Tai Wang, Li Chen, Silei Wu, Hanming Deng, Yi Gu, Lewei Lu, Ping Luo, Dahua Lin, et al. Scene as occupancy. In *ICCV*, pages 8406–8415, 2023. 2

[40] Jiabao Wang, Zhaojiang Liu, Qiang Meng, Liujiang Yan, Ke Wang, Jie Yang, Wei Liu, Qibin Hou, and Ming-Ming Cheng. Opus: occupancy prediction using a sparse set. *arXiv preprint arXiv:2409.09350*, 2024. 3

[41] Lening Wang, Wenzhao Zheng, Yilong Ren, Han Jiang, Zhiyong Cui, Haiyang Yu, and Jiwen Lu. Occsora: 4d occupancy generation models as world simulators for autonomous driving. *arXiv preprint arXiv:2405.20337*, 2024. 2

[42] 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. 7

[43] 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. *arXiv preprint arXiv:2303.03991*, 2023. 2

[44] Yi Wei, Linqing Zhao, Wenzhao Zheng, Zheng Zhu, Jie Zhou, and Jiwen Lu. Surroundocc: Multi-camera 3d occupancy prediction for autonomous driving. In *ICCV*, pages 21729–21740, 2023. 2, 4, 7

[45] Xu Yan, Jiantao Gao, Jie Li, Ruimao Zhang, Zhen Li, Rui Huang, and Shuguang Cui. Sparse single sweep lidar point cloud segmentation via learning contextual shape priors from scene completion. In *AAAI*, volume 35, pages 3101–3109, 2021. 2

[46] Ziyang Yan, Wenzhen Dong, Yihua Shao, Yuhang Lu, Liu Haiyang, Jingwen Liu, Haozhe Wang, Zhe Wang, Yan Wang, Fabio Remondino, et al. Renderworld: World model with self-supervised 3d label. *arXiv preprint arXiv:2409.11356*, 2024. 2

[47] Dongqiangzi Ye, Zixiang Zhou, Weijia Chen, Yufei Xie, Yu Wang, Panqu Wang, and Hassan Foroosh. Lidarmultinet: Towards a unified multi-task network for lidar perception. *arXiv preprint arXiv:2209.09385*, 2022. 2

[48] Maosheng Ye, Rui Wan, Shuangjie Xu, Tongyi Cao, and Qifeng Chen. Drinet++: Efficient voxel-as-point point cloud segmentation. *arXiv preprint arXiv: 2111.08318*, 2021. 2

[49] Zichen Yu, Changyong Shu, Jiajun Deng, Kangjie Lu, Zongdai Liu, Jiangyong Yu, Dawei Yang, Hui Li, and Yan Chen. Flashocc: Fast and memory-efficient occupancy prediction via channel-to-height plugin. *arXiv preprint arXiv:2311.12058*, 2023. 2

[50] 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. 2, 4, 7

[51] Wenzhao Zheng, Weiliang Chen, Yuanhui Huang, Borui Zhang, Yueqi Duan, and Jiwen Lu. Occworld: Learning a 3d occupancy world model for autonomous driving. In *ECCV*, 2024. 2

[52] Wenzhao Zheng, Junjie Wu, Yao Zheng, Sicheng Zuo, Zixun Xie, Longchao Yang, Yong Pan, Zhihui Hao, Peng Jia, Xianpeng Lang, et al. Gaussianad: Gaussian-centric end-to-end autonomous driving. *arXiv preprint arXiv:2412.10371*, 2024. 2

[53] Yin Zhou and Oncel Tuzel. Voxelnet: End-to-end learning for point cloud based 3d object detection. In *CVPR*, pages 4490–4499, 2018. 2

[54] Sicheng Zuo, Wenzhao Zheng, Yuanhui Huang, Jie Zhou, and Jiwen Lu. Pointocc: Cylindrical tri-perspective view for point-based 3d semantic occupancy prediction. *arXiv preprint arXiv:2308.16896*, 2023. 2

[55] Sicheng Zuo, Wenzhao Zheng, Yuanhui Huang, Jie Zhou, and Jiwen Lu. Gaussianworld: Gaussian world model for streaming 3d occupancy prediction. *arXiv preprint arXiv:2412.10373*, 2024. 2, 3

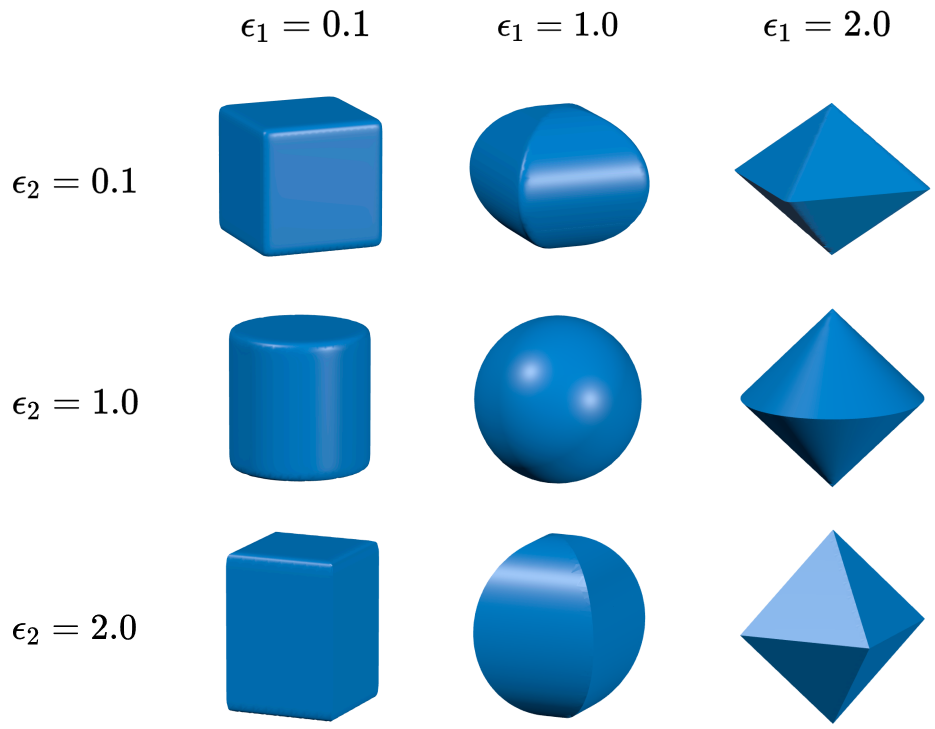


Figure 7: **Superquadrics of different shape parameters.** The figure illustrates how varying ϵ_1 and ϵ_2 produces a wide range of shapes, from star-like and rounded shapes to square-like structures. Such diversity enables superquadrics to flexibly model complex object geometries in 3D scenes.

A Additional Superquadric Details

Superquadrics are a powerful family of parameterized surfaces that can represent various geometric shapes. With just a few parameters, superquadrics can generate shapes ranging from basic ellipsoids, cuboids, and cylinders to more complex shapes with rounded corners, star-like profiles, and smooth transitions between them. This geometric flexibility makes superquadrics ideal for efficiently modeling diverse objects in autonomous driving scenes. The shape of a superquadric is mainly controlled by two groups of parameters. The first group consists of scaling factors (s_x, s_y, s_z), which define the superquadric's dimensions or "radii" along its three principal axes, determining the object's overall size and aspect ratio. The second group includes two key shape parameters (ϵ_1, ϵ_2) that determine the degree of "squareness" or "roundness" of the object. ϵ_1 primarily controls the object's profile in planes containing the z-axis (such as the xz- or yz-plane): smaller values (close to 0.1) create sharper profiles, $\epsilon_1 = 1.0$ produces elliptical outlines, and larger values (up to 2.0) result in flatter contours. Similarly, ϵ_2 controls the shape of the cross-section in the xy-plane. A small ϵ_2 yields a star-shaped cross-section, ϵ_2=1.0 gives a circular outline, and large ϵ_2 values produce square-like shapes. As shown in Fig 7, varying ϵ_1 and ϵ_2 of superquadrics results in a wide range of shapes. By combining these scaling and shape parameters, superquadrics can efficiently represent diverse object geometries in autonomous driving scenes. This capability allows them to capture complex structures with significantly fewer primitives than traditional representations (like ellipsoidal Gaussians), highlighting their superior modeling efficiency and expressive power for 3D scene understanding tasks.

B Additional Experiments

We visualize the position distributions of scene primitives using 1600 superquadrics versus 6400 Gaussians in Figure 8. Gaussian-based methods require a dense arrangement of Gaussians throughout the entire 3D space to model the scene, leading to numerous redundant Gaussians and low modeling efficiency. In contrast, our superquadric-based method learns well-structured spatial arrangements, enabling it to effectively model the scene structure with significantly fewer primitives.

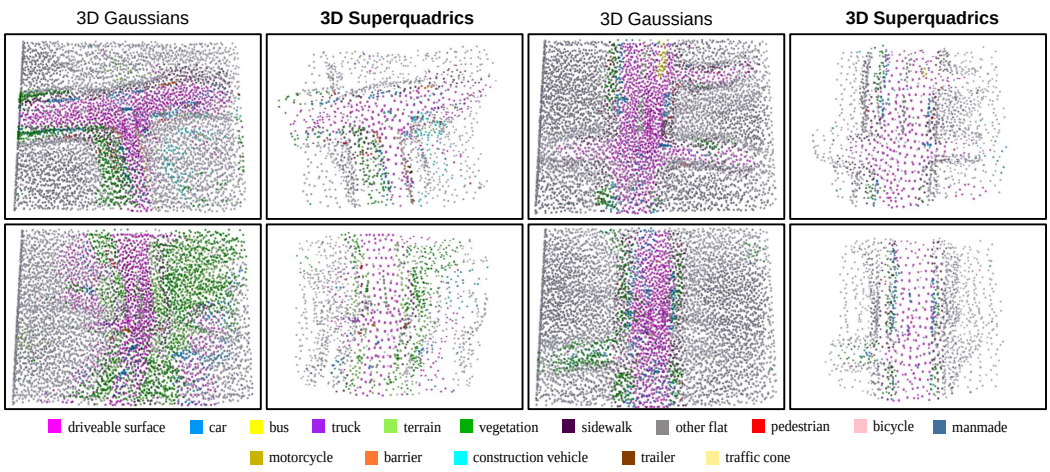


Figure 8: **Visualizations of primitive position distributions learned by different methods.** Our approach produces well-structured spatial arrangements while using significantly fewer primitives.

C Additional Implementation Details

We provide implementation details of the prunning-and-splitting module. To clarify, we take the QuadricFormer with N superquadrics as an example and describe the process as follows:

Initial Training: We first train a QuadricFormer with B=4 quadric-encoder blocks and without the prunning-and-splitting module. The model starts from N randomly initialized superquadrics Q_{init} and predicts adjusted superquadrics Q.

Prunning-and-Splitting Module: During experiments, we observed that some superquadrics in Q contribute little to scene modeling, which are usually located in empty regions with very small scales. To address this, we introduce the prunning-and-splitting module:

- We divide all superquadrics in Q into two groups based on the product of their scales: the N_{crop} superquadrics C with the smallest scales and the remaining N_{valid} superquadrics V , where $N = N_{valid} + N_{crop}$.
- We discard the smaller superquadrics C as they are most likely to contribute little to scene modeling.
- We randomly sample N_{crop} superquadrics from V to form S . The features of S remain unchanged, and only their positions are slightly adjusted.

Further Refinement: Finally, S and V are passed through two additional quadric-encoder blocks to further refine their attributes, resulting in the final superquadrics Q_{final} . At this stage, we load the pretrained model parameters and continue training for 10 more epochs.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: We claim a superquadric-based representation for efficient 3D semantic occupancy. These claims have been justified by the experimental results in Section 4.

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: Limitations have been discussed in the Section 5.

   Guidelines:

   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [NA]

Justification: No theoretical results.

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: We include details about our experiments in Section 4.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [Yes]

Justification: Code and instructions to reproduce our results on public datasets are provided in the supplementary materials.

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (https://nips.cc/public/guides/CodeSubmissionPolicy) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (https://nips.cc/public/guides/CodeSubmissionPolicy) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

   Answer: [Yes]

   Justification: We have provided all details in Section 4 and in the code.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [No]

   Justification: It is not a conventional procedure to report error bars in this field. However, the results do not fluctuate much through differen runs from our experiments.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
   - The assumptions made should be given (e.g., Normally distributed errors).

- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes] .

   Justification: All details have been provided in Section 4.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics https://neurips.cc/public/EthicsGuidelines?

   Answer: [Yes]

   Justification: This paper conforms with the NeurIPS Code of Ethics because authors have read and followed it.

   Guidelines:

   - The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: Broader impact is discussed in Section 5.

    Guidelines:

    - The answer NA means that there is no societal impact of the work performed.
    - If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.

- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

    Answer: [NA]

    Justification: Only existing datasets and models have been used in this work.

    Guidelines:

    - The answer NA means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: Authors of used code libraries, models and data have been cited and version details have been provided in our code package.

    Guidelines:

    - The answer NA means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.

- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [Yes]

    Justification: Our code contains instructions how to run experiments and text comments.

    Guidelines:

    - The answer NA means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [NA]

    Justification: Crowdsourcing has not been used.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [NA]

    Justification: Crowdsourcing has not been used.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does not impact the core methodology, scientific rigorousness, or originality of the research, declaration is not required.

Answer: [NA]

Justification: Authors did not use LLMs for core method development or any components.

Guidelines:

- The answer NA means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy (https://neurips.cc/Conferences/2025/LLM) for what should or should not be described.