LIGHTROCC: LIGHTWEIGHT 4D OCCUPANCY FORECASTING VIA INSTANCE-CENTRIC 3D GAUSSIANS

Anonymous authors
Paper under double-blind review

ABSTRACT

Forecasting future 3D occupancy from surround-view cameras is essential for autonomous driving, yet existing approaches rely on dense voxel or bird's-eye-view representations whose cost grows rapidly with spatial resolution and prediction horizon. Because these representations do not explicitly maintain object identities, they also struggle to preserve instance consistency over time. We present **LighTROcc**, a lightweight instance-centric framework that represents movable objects with a compact set of learned queries and predicts present and future occupancy in a single forward pass. LighTROcc localizes each query through attention-guided forward lifting, combining image-space cross-attention, query-specific depth, and camera geometry to estimate its 3D center. Each instance is modeled as a mixture of anisotropic 3D Gaussians and propagated across future steps using predicted displacements, producing continuous, temporally consistent occupancy forecasts. Experiments on nuScenes and supplemented nuScenes-Occupancy show that LighTROcc outperforms the evaluated dense and instance-wise baselines in instance-level forecasting accuracy while maintaining strong voxel-level occupancy quality. Across different model configurations, LighTROcc achieves a favorable balance between forecasting accuracy and computational efficiency, demonstrating the potential of compact instance-centric modeling for camera-based 4D occupancy forecasting.

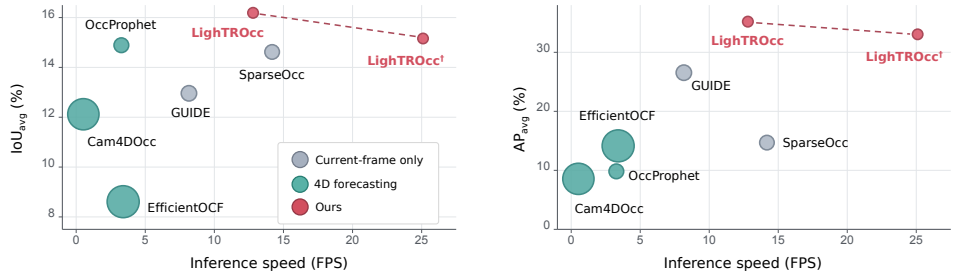

Figure 1: Accuracy-efficiency comparison with current-only and 4D forecasting methods. LighTROcc achieves a favorable trade-off between inference speed and both IoU and AP while predicting current and future occupancy in a single forward pass.

1 INTRODUCTION

Safe and real-time autonomous driving requires an accurate and efficient understanding of which regions of the surrounding space are free or occupied. Extensive research has investigated fine-grained camera-based occupancy perception to provide a comprehensive representation of the current driving environment (Huang et al., 2023; Wei et al., 2023; Zhang et al., 2023; Wang et al., 2023b; Tong et al., 2023; Ma et al., 2024b). From the perspective of planning and control, however, estimating only the present state is insufficient. An autonomous vehicle must also anticipate whether currently free space will remain traversable or become occupied when the vehicle reaches it. This need has motivated growing interest in 4D occupancy forecasting, which predicts a sequence of future 3D occupancy states from past observations. Since temporal occupancy changes are primarily driven by movable objects, these forecasting methods commonly focus on general movable objects (GMOs).

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

Recent 4D occupancy forecasting methods typically model temporal evolution by extending spatially indexed representations, such as voxels or bird's-eye-view (BEV) cells, along the temporal dimension (Ma et al., 2024a; Xu et al., 2025; Chen et al., 2025). Many of these methods rely on volumetric processing, such as 3D convolutions, to reconstruct and forecast occupancy. Although such representations retain an explicit volumetric description of the scene, they require substantial computation, and their cost increases rapidly with spatial resolution and forecasting horizon (Chen et al., 2025). Consequently, forecasting multiple future frames while satisfying real-time inference requirements remains challenging. A recent approach factorizes 3D occupancy into a 2D BEV representation and column-wise height information to avoid explicit full-volume processing (Xu et al., 2025). While this lower-dimensional formulation can alleviate the burden of volumetric processing, its column-based reconstruction may inadequately capture complex object geometry and fine-grained volumetric extents. More broadly, existing 4D occupancy forecasting methods are generally not designed in an instance-centric manner. When object identities are required, instances must instead be recovered through additional association, grouping, or flow-based processing. Such indirect recovery can lead to temporal inconsistencies, including the merging of distinct objects or the fragmentation of a single object across frames.

Maintaining consistent object identities over time therefore calls for an instance-centric formulation that directly recognizes and forecasts individual objects. Existing approaches have explored instance query candidates, where each query is intended to represent an object (Wang et al., 2022; Liu et al., 2022; 2024; Hu et al., 2026). In many such methods, queries are initialized from spatial anchor points, and a large candidate set is required to provide sufficient coverage of the 3D scene. Because this set is predefined to be substantially larger than the actual number of object instances, it can produce redundant predictions and incur unnecessary computational overhead. Furthermore, these queries are generally designed to represent instances in the current scene rather than to persist as object-specific states throughout future frames. These observations motivate a lightweight instance-centric framework that compactly represents the current scene and directly propagates object instances across multiple future time steps.

Building on this principle, we introduce **LighTROcc**, a lightweight instance-centric transformer for camera-based 4D occupancy forecasting. Rather than distributing a large set of spatially initialized candidate queries throughout the 3D scene, LighTROcc directly localizes a compact set of queries in 3D through attention-guided forward lifting. Specifically, each query derives its image-space location from the attention map and combines it with a query-specific depth prediction and camera geometry. The localized query further predicts a mixture of 3D Gaussians that compactly represents the full volumetric extent of the corresponding instance. The resulting instance representation is then propagated to future frames using predicted displacements, maintaining object-level consistency over time. This compact formulation enables LighTROcc to forecast multiple future occupancy states in a single forward pass, yielding a favorable balance between forecasting accuracy and inference speed, as shown in Figure 1. We evaluate our method on nuScenes (Caesar et al., 2020) and supplemented annotations from nuScenes-Occupancy (Wang et al., 2023b), where it outperforms the evaluated baselines, demonstrating the effectiveness of the proposed formulation.

Our contributions are summarized as follows:

- We propose LighTROcc, a lightweight instance-centric network that forecasts future 4D occupancy states in a single forward pass.
- We introduce an attention-guided forward lifting formulation that directly localizes 3D instance centers from attention map and depth, enabling compact instance prediction.
- Extensive experiments on nuScenes and supplemented nuScenes-Occupancy demonstrate the highest accuracy with lower computational cost and faster inference.

2 Related Work

2.1 3D Occupancy Representations and Query-Based Perception

Occupancy representations. 3D occupancy prediction has been explored using dense voxel grids, reduced-dimensional representations, and continuous 3D primitives. Early semantic scene completion and camera-based methods construct dense voxel volumes to preserve full 3D structure, but their computational cost scales with spatial resolution (Song et al., 2017; Roldão et al., 2020; Cao &

de Charette, 2022; Wei et al., 2023; Li et al., 2023b). To reduce this burden, FlashOcc compresses volumetric occupancy by mapping BEV features to height-aware outputs, while PointOcc factorizes 3D space into cylindrical tri-perspective representations (Yu et al., 2023; Zuo et al., 2023). More recently, Gaussian-based approaches provide continuous alternatives to voxel grids using sparse 3D Gaussians or Gaussian splatting representations for occupancy prediction (Huang et al., 2024; 2025; Gan et al., 2025; Jiang et al., 2025).

Geometry-indexed query methods. A major query-based line assigns queries to predefined geometric elements such as BEV cells, planes, or voxels. BEVFormer learns dense BEV queries, TPVFormer introduces three orthogonal query planes, and VoxFormer and PanoOcc adopt voxel-indexed representations for occupancy prediction (Li et al., 2022; Huang et al., 2023; Li et al., 2023a; Wang et al., 2024b). Subsequent methods improve efficiency through perspective supervision, polar layouts, dual-path decoding, octree queries, and compact query representations (Yang et al., 2023; Jiang et al., 2023; Zhang et al., 2023; Lu et al., 2024; Ma et al., 2024b; Oh et al., 2025). However, spatially indexed queries remain coupled to the predefined discretization, causing computational and memory costs to grow with spatial resolution.

Object-centric query methods. DETR3D associates queries with 3D reference points projected into camera views, while PETR and PETRv2 incorporate 3D positional information into image features (Wang et al., 2022; Liu et al., 2022; 2023b). Subsequent methods extend sparse queries to temporal modeling, tracking, sensor fusion, and end-to-end driving (Lin et al., 2022; 2023a;b; Liu et al., 2023a; Wang et al., 2023a; Xiong et al., 2023; Zhang et al., 2022; Chen et al., 2023; Sun et al., 2025). Nevertheless, several approaches still rely on reference points or anchor-based initialization, requiring large candidate sets for scene coverage (Wang et al., 2022; Lin et al., 2022). Sparse query formulations have also been extended to occupancy prediction through sparse points, sets, contextual queries, and Gaussian representations (Tang et al., 2024; Liu et al., 2024; Wang et al., 2024a; Shi et al., 2024; Jiang et al., 2024; Hu et al., 2026; Park et al., 2026). S2GO uses spatially anchored 3D queries with Gaussian decoding for streaming occupancy, whereas LighTROcc adopts compact object-centric queries with attention-guided localization for multi-step occupancy forecasting.

2.2 Camera-Only 4D Occupancy Forecasting

Camera-only 4D occupancy forecasting predicts present and future 3D occupancy states from historical multi-view images. Cam4DOcc introduced a benchmark and OCFNet, an end-to-end baseline that processes multi-frame 3D voxel features (Ma et al., 2024a). OccProphet improves efficiency with a lightweight Observer-Forecaster-Refiner framework that aggregates multi-frame voxel features (Chen et al., 2025), while EfficientOCF decouples spatial and temporal processing through BEV occupancy, height, and flow prediction (Xu et al., 2025). Despite these advances, their forecasts remain spatially indexed rather than maintaining objects as persistent instance states. LighTROcc instead represents each object with an instance-specific query and forecasts its occupancy across future frames, preserving object-level consistency throughout the forecasting horizon.

3 Method

3.1 Problem formulation

Let $\boldsymbol{I}_t^v$ denote the RGB image captured by camera $v \in \{1, \dots, V\}$ at time $t \in \{-T_p, \dots, 0\}$, where T_p is the number of past frames preceding the present and $t = 0$ denotes the present frame. Following prior camera-only occupancy forecasting methods (Ma et al., 2024a; Xu et al., 2025; Chen et al., 2025), we focus on the occupancy evolution of general movable objects (GMOs). Given the multi-view image history and the corresponding camera intrinsics, extrinsics, and ego poses, the task is to predict the present and future voxel occupancy states:

$$\{\hat{\boldsymbol{O}}_k\}_{k=0}^{T_f}, \qquad \hat{\boldsymbol{O}}_k \in [0,1]^{X \times Y \times Z}. \tag{1}$$

Here, $\hat{\boldsymbol{O}}_k(x, y, z)$ denotes the probability that voxel (x, y, z) is occupied at time k, where $k = 0$ corresponds to the present and $k > 0$ to the future. We set $T_p = 2$ and forecast $T_f = 4$ frames.

Unlike conventional dense approaches that construct voxel-wise features over a discretized 3D grid, LighTROcc represents movable objects with a compact set of instance queries. Each query predicts confidence, depth, Gaussian mixture parameters, and a future trajectory. The resulting continuous Gaussian volumes are rasterized into the occupancy sequence $\{\hat{\boldsymbol{O}}_k\}_{k=0}^{T_f}$.

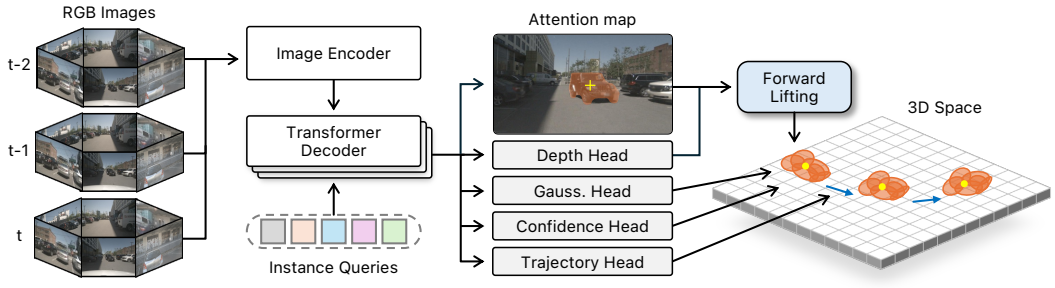

Figure 2: Overview of LighTROcc. Compact instance queries are sequentially refined through a transformer decoder using multi-view image features. Each query is localized in 3D via attention-guided forward lifting, while heads predict 3D Gaussians, confidence, and future trajectory.

3.2 Transformer-based Query Decoder

As illustrated in Fig. 2, a shared image encoder extracts multi-scale context features $\boldsymbol{F}_{t,\ell}^{v} \in \mathbb{R}^{H_\ell \times W_\ell \times D}$ from each surrounding image $\boldsymbol{I}_t^v$. Since the shared encoder processes each image independently, the resulting visual features do not explicitly encode their spatial locations, camera viewpoints, or acquisition times. We therefore augment each image token with a 2D sinusoidal positional encoding, a camera embedding derived from the corresponding camera parameters and ego pose, a learned camera-ID embedding, and a temporal embedding representing the observation time. For decoder layer ℓ, the augmented tokens from all cameras are flattened to form the multi-view context features $\boldsymbol{F}_{t,\ell}$.

We initialize the decoder with N_q learned instance queries $\boldsymbol{Q} \in \mathbb{R}^{N_q \times D}$. For each observed frame, the queries sequentially interact with the multi-view features through cross-attention and are refined by self-attention and a feed-forward network. At decoder layer ℓ, the update is formulated as

$$\begin{aligned} \bar{\boldsymbol{Q}}_t^{\ell} &= \mathrm{LN}\big(\boldsymbol{Q}_t^{\ell-1} + \mathrm{CA}_\ell(\boldsymbol{Q}_t^{\ell-1}, \boldsymbol{F}_{t,\ell})\big)\,, \\ \widetilde{\boldsymbol{Q}}_t^{\ell} &= \mathrm{LN}\big(\bar{\boldsymbol{Q}}_t^{\ell} + \mathrm{SA}_\ell(\bar{\boldsymbol{Q}}_t^{\ell})\big)\,, \\ \boldsymbol{Q}_t^{\ell} &= \mathrm{LN}\Big(\widetilde{\boldsymbol{Q}}_t^{\ell} + \mathrm{FFN}_\ell(\widetilde{\boldsymbol{Q}}_t^{\ell})\Big)\,. \end{aligned} \tag{2}$$

Here, LN denotes layer normalization, while CA_ℓ, SA_ℓ, and FFN_ℓ denote the cross-attention, self-attention, and feed-forward modules at layer ℓ, respectively. The observation frames are processed chronologically, with the final query state of each frame initializing the next. This sequential propagation serves as a compact temporal memory without constructing dense spatiotemporal features.

For efficient coarse-to-fine decoding, successive layers attend to image features of increasing spatial resolution. Early layers capture broad context at low computational cost, while the final layer retains the spatial detail required for 3D lifting. We use three decoder layers with four attention heads.

3.3 Attention-guided forward lifting

After query decoding, each query contains object-specific 2D visual evidence, but directly regressing its metric 3D location from the latent query vector remains non-trivial. We therefore decompose 3D localization into two simpler predictions: image-plane localization and query-specific depth estimation. The known camera parameters and ego pose then serve as a geometric bridge that lifts these predictions into 3D. Specifically, as illustrated in Figure 3, we reuse the query-to-image attention as an image-plane localization cue and combine it with the predicted depth to recover a 3D center.

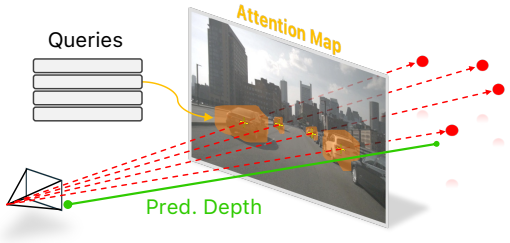

Figure 3: Illustration of attention-guided forward lifting, combining query attention and predicted depth to recover each instance center in 3D.

Let $\boldsymbol{A}^{v}_{t,q} \in [0,1]^{H\times W}$ denote the final-layer cross-attention map of query q over camera v at time t. We spatially normalize the attention map within each view and obtain a differentiable image-plane center through soft-argmax:

$$\boldsymbol{p}^{v}_{t,q} = \sum_{\boldsymbol{u}\in\Omega} \bar{\boldsymbol{A}}^{v}_{t,q}(\boldsymbol{u})\,\boldsymbol{u}, \tag{3}$$

where Ω denotes the spatial domain of the feature map, $\boldsymbol{u} = [u, v]^{\top}$ is a feature-map coordinate, and $\bar{\boldsymbol{A}}^{v}_{t,q}$ is the normalized attention map. To preserve gradients and stabilize image-plane localization, we use soft-argmax rather than discrete argmax. In parallel, a lightweight depth head predicts a categorical distribution $\boldsymbol{\pi}_{t,q} \in [0,1]^{N_d}$ over N_d predefined depth bins. Given a depth range $[d_{\min}, d_{\max}]$, the center of the b-th bin is defined as

$$\rho_b = d_{\min} + \left(b - \frac{1}{2}\right)\frac{d_{\max} - d_{\min}}{N_d}, \qquad b = 1, \ldots, N_d, \tag{4}$$

and the continuous query depth is obtained as the expectation over the fixed bin centers:

$$d_{t,q} = \sum_{b=1}^{N_d} \pi_{t,q,b}\rho_b. \tag{5}$$

The attention map determines the image region associated with the query, while the depth distribution resolves its position along the corresponding camera ray. For each query, we select the valid camera with the largest total mass in the original cross-attention map and map its location back to the original image coordinates. The resulting image-plane location is then lifted into 3D using the predicted depth and the corresponding camera geometry:

$$\boldsymbol{c}_{t,q} = \mathrm{Lift}\left(\boldsymbol{p}_{t,q}, d_{t,q}; \boldsymbol{K}_t, \boldsymbol{E}_t\right), \tag{6}$$

where $\boldsymbol{K}_t$ and $\boldsymbol{E}_t$ denote the intrinsic and extrinsic parameters of the selected camera, respectively. The $\mathrm{Lift}(\cdot)$ operation maps the image-plane location to 3D along its camera ray using the predicted depth and camera geometry. This attention-guided forward lifting provides each query with a consistent 3D center without introducing spatially initialized 3D candidates.

3.4 Occupancy Forecasting

To capture finer object geometry than the coarse occupancy volumes derived from inflated 3D boxes, we adopt an instance-level Gaussian representation similar to GUIDE (Hu et al., 2026). Given the lifted 3D center $\boldsymbol{c}_q$ and decoded feature of each query, we represent its volumetric extent using a compact mixture of N_g anisotropic 3D Gaussians. For each component, lightweight prediction heads estimate a center offset $\Delta\boldsymbol{\mu}_{q,g}$, axis-wise standard deviations $\boldsymbol{\sigma}_{q,g}$, a rotation matrix $\boldsymbol{R}_{q,g}$, and an occupancy weight $\alpha_{q,g}$. Its mean and covariance are defined as

$$\boldsymbol{\mu}_{q,g} = \boldsymbol{c}_q + \Delta\boldsymbol{\mu}_{q,g}, \qquad \boldsymbol{\Sigma}_{q,g} = \boldsymbol{R}_{q,g}\,\mathrm{diag}\left(\boldsymbol{\sigma}^2_{q,g}\right)\boldsymbol{R}^{\top}_{q,g}. \tag{7}$$

The resulting Gaussian components jointly describe the object geometry as a continuous occupancy field that can be discretized at different voxel resolutions. Unlike voxel-grid representations tied to a predefined discretization, our continuous Gaussian representation can be discretized at different resolutions during inference without retraining or modifying the network architecture.

To forecast future occupancy, a lightweight trajectory head predicts a sequence of displacements for each query. These displacements are accumulated from the present 3D center to determine the query location at each future step. The associated Gaussian mixture is then translated along the predicted trajectory while retaining its estimated offsets, covariance, and occupancy weights. This enables the model to generate multiple future occupancy states in a single forward pass.

3.5 Optimization

In this section, we describe the ground-truth assignment and training objectives used to optimize the instance-wise queries. Since the learned queries are permutation-invariant, we establish a bipartite correspondence between the predicted queries and ground-truth instances using Hungarian matching (Kuhn, 1955). The matching cost combines the 3D center distance over the observed frames, the

Table 1: Quantitative comparison of dense voxel-wise and instance-wise methods for GMO occupancy forecasting. LighTROcc† denotes the resolution-matched variant using 704×256 inputs. Best and second-best results are shown in bold and underlined, respectively.

Method	Type	Input Res.	IoU_{cur}	IoU_{fut}	IoU_{avg}	AP_{cur}	AP_{fut}	AP_{avg}	FPS
Cam4DOcc (Ma et al., 2024a)	Dense Voxel wise	1600×896	13.42	11.80	12.12	10.71	8.02	8.56	0.52
EfficientOCF (Xu et al., 2025)		1600×896	8.67	8.59	8.61	14.25	14.13	14.15	3.40
OccProphet (Chen et al., 2025)		800×448	17.57	14.23	14.89	20.14	7.25	9.83	3.27
SparseOcc (Liu et al., 2024)	Instance wise	704×256	18.42	13.67	14.62	20.05	13.36	14.70	<u>14.18</u>
GUIDE (Hu et al., 2026)		704×256	15.21	12.40	12.96	34.73	24.51	26.55	8.16
LighTROcc† (Ours)		704×256	<u>18.46</u>	<u>14.34</u>	<u>15.16</u>	<u>40.03</u>	<u>31.30</u>	<u>33.05</u>	**25.10**
LighTROcc (Ours)		1600×896	**19.76**	**15.30**	**16.19**	**42.62**	**33.34**	**35.20**	12.80

spatial overlap between the query attention map and the projected ground-truth instance region, and the predicted instance confidence. Geometric, occupancy, and trajectory supervision is applied only to matched pairs, while unmatched queries are treated as background for confidence supervision.

The confidence head predicts a foreground probability $s_q \in [0, 1]$ for each query. Based on the matching result, matched and unmatched queries are assigned foreground and background targets, respectively. The confidence loss $\mathcal{L}_{\text{conf}}$ is then computed using cross-entropy between the predicted probabilities and the corresponding binary targets. At inference, only queries with $s_q > \tau_{\text{conf}}$ are retained for occupancy prediction, where $\tau_{\text{conf}} = 0.9$.

For each matched pair, we compute the lifted 3D center loss $\mathcal{L}_{\text{cen}}$ using the ℓ_1 norm and supervise the predicted depth distribution with the cross-entropy loss $\mathcal{L}_{\text{depth}}$. To encourage accurate attention-based 2D localization, the attention localization loss $\mathcal{L}_{\text{attn}}$ is defined as the negative log-likelihood of the normalized attention mass assigned to the projected ground-truth region. To supervise object geometry, the predicted Gaussian mixture is voxelized and optimized against the ground-truth occupancy using focal and Tversky losses (Lin et al., 2017; Salehi et al., 2017), denoted by $\mathcal{L}_{\text{occ}}$. Finally, the trajectory head is supervised with an ℓ_1 displacement loss $\mathcal{L}_{\text{traj}}$ over the valid future frames. The overall training objective is

$$\mathcal{L} = \lambda_{\text{conf}}\mathcal{L}_{\text{conf}} + \lambda_{\text{cen}}\mathcal{L}_{\text{cen}} + \lambda_{\text{depth}}\mathcal{L}_{\text{depth}} + \lambda_{\text{attn}}\mathcal{L}_{\text{attn}} + \lambda_{\text{occ}}\mathcal{L}_{\text{occ}} + \lambda_{\text{traj}}\mathcal{L}_{\text{traj}}, \tag{8}$$

where each λ controls the contribution of the corresponding loss term.

4 Experiments

4.1 Experimental Setup

Datasets. We train and evaluate LighTROcc and the baseline models using nuScenes (Caesar et al., 2020) and nuScenes-Occupancy (Wang et al., 2023b). We obtain multi-view RGB images, camera parameters, and ego poses from nuScenes, and use the fine-grained voxel occupancy annotations from nuScenes-Occupancy for supervision. Following the prior work (Ma et al., 2024a; Xu et al., 2025; Chen et al., 2025), we use 700 scenes for training and 150 for evaluation, yielding 23,930 and 5,119 sequences, respectively. Each sequence contains two past observations, one present observation, and four future occupancy targets. Since the nuScenes keyframes are sampled at 2 Hz, these targets correspond to prediction horizons of 0.5, 1.0, 1.5, and 2.0 seconds. The evaluation range is $[-51.2, 51.2]$ m along both horizontal axes and $[-5, 3]$ m vertically, discretized into a $[512 \times 512 \times 40]$ grid with a voxel size of 0.2 m. Since the LiDAR-derived nuScenes-Occupancy annotations are highly sparse, we supplement them with auxiliary assets. Further details are provided in the Appendix.

Evaluation protocols and metrics. Following prior works (Ma et al., 2024a; Xu et al., 2025; Chen et al., 2025), we evaluate foreground general movable object (GMO) instances in both the present and future frames. For current-only methods, we follow Cam4DOcc and EfficientOCF by using current-frame predictions for all future steps. To assess occupancy forecasting performance, we report voxel-wise IoU against the fine-grained 3D occupancy ground truth. For instance-level

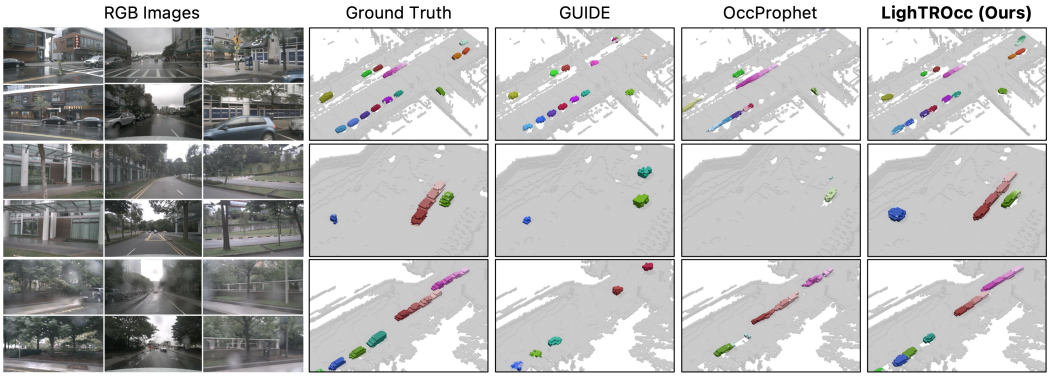

Figure 4: Qualitative comparison of GMO occupancy predictions from multi-view RGB images. LighTROcc more faithfully captures object locations and future motion, with darker colors indicating later forecasting steps.

evaluation, we additionally report average precision (AP) based on the voxel IoU between predicted and ground-truth instance occupancy masks. Following GUIDE (Hu et al., 2026), we compute AP at IoU thresholds of 0.1, 0.2, and 0.3 and report the mean. For dense voxel-wise methods, we recover instance masks using the official flow-based grouping and compute AP using an evaluation scheme inspired by Panoptic-DeepLab (Cheng et al., 2020). For center localization, we report $\mathrm{AP}_{\mathrm{ctr}}$ and $\mathrm{Recall}_{\mathrm{loc}}$ at a 1 m threshold, measuring confidence-ranked center localization and active-query GT coverage, respectively. All baseline models are reproduced using their official code, and all methods are evaluated on a single NVIDIA RTX Pro 6000 GPU. Further details are provided in the Appendix.

4.2 Experimental Results

Forecasting General Movable Object Occupancy. We conduct quantitative comparisons to evaluate 4D occupancy forecasting for general movable objects. For a resolution-matched comparison with prior instance-wise methods, we additionally report LighTROcc†, a lighter variant using 704×256 inputs. As shown in Table 1, dense voxel-wise approaches achieve competitive voxel-wise IoU, indicating effective reconstruction of overall occupied regions. However, their instance-agnostic forecasts require flow-based post-processing to recover object masks, resulting in relatively low AP and weaker object-level consistency over time. In contrast, instance-wise approaches generally achieve higher AP while maintaining competitive IoU. LighTROcc obtains the best results across all reported IoU and AP metrics, reaching $\mathrm{IoU}_{\mathrm{avg}} = 16.19$ and $\mathrm{AP}_{\mathrm{avg}} = 35.20$ while running at 12.80 FPS. Notably, LighTROcc† retains comparable occupancy accuracy while running nearly twice as fast as the full-resolution model. These results demonstrate that LighTROcc maintains reliable occupancy forecasting and object-level consistency across different input resolutions while remaining suitable for real-time inference.

Qualitative results. To qualitatively assess LighTROcc, we compare its predictions with representative baselines. As shown in Figure 4, the results visualize instance-wise GMO occupancy and its evolution over time, while GUIDE (Hu et al., 2026) is shown only for the current frame because it does not model future states. GUIDE localizes objects well in the current frame, but cannot forecast future occupancy due to the absence of future-state modeling. OccProphet (Chen et al., 2025) supports future occupancy forecasting, but often fails to preserve consistent object geometry across time. In contrast, LighTROcc more accurately captures future object locations while maintaining consistent object representations throughout the forecasting horizon. Its accurate trajectory forecasting further demonstrates its potential for downstream planning and control in autonomous driving. Additional qualitative results and analyses are provided in the Appendix.

4.3 Computational Efficiency and Model Compactness

We compare LighTROcc with prior baselines in terms of computational efficiency and model compactness. Table 2 reports parameters, GFLOPs, latency, FPS, and forecasting accuracy, with all models benchmarked using official implementations on a single NVIDIA RTX Pro 6000 GPU.

Table 2: Comparison of model compactness, computational efficiency, and forecasting accuracy. Runtime is measured using official implementations on a single NVIDIA RTX Pro 6000 GPU. LighTROcc† denotes the resolution-matched variant using 704×256 inputs. Best and second-best results are shown in bold and underlined, respectively.

Category	Method	Input Res.	Params. (M)	GFLOPs	Latency (ms)	FPS	IoU_{avg}	AP_{avg}
Current-only	SparseOcc (Liu et al., 2024)	704×256	80	<u>568</u>	<u>70.53</u>	<u>14.18</u>	14.62	14.70
	GUIDE (Hu et al., 2026)	704×256	90	1,530	122.51	8.16	12.96	26.55
4D forecasting	Cam4DOcc (Ma et al., 2024a)	1600×896	370	13,168	1,902.26	0.52	12.12	8.56
	EfficientOCF (Xu et al., 2025)	1600×896	388	5,143	294.24	3.40	8.61	14.15
	OccProphet (Chen et al., 2025)	800×448	82	2,020	305.93	3.27	14.89	9.83
	LighTROcc† (Ours)	704×256	**42**	**154**	**39.84**	**25.10**	<u>15.16</u>	<u>33.05</u>
	LighTROcc (Ours)	1600×896	**42**	1,225	78.12	12.80	**16.19**	**35.20**

Table 3: Ablation of 3D center localization strategies. Direct 3D regression is compared with hard- and soft-argmax variants of attention-guided forward lifting under the same model configuration.

Localization	$\text{IoU}_{\text{avg}} \uparrow$	$\text{AP}_{\text{avg}} \uparrow$	$\text{AP}_{\text{ctr}} \uparrow$	$\text{Recall}_{\text{loc}} \uparrow$
Direct regression (MLP)	11.13	26.10	17.23	21.14
Forward Lifting (hard)	11.20	28.50	25.43	29.84
Forward Lifting (soft)	**16.19**	**35.20**	**27.31**	**31.06**

Table 4: Effect of instance query count on computational efficiency and forecasting accuracy. All variants are evaluated using full-resolution inputs.

# Queries	FPS ↑	$\text{IoU}_{\text{avg}} \uparrow$	$\text{AP}_{\text{avg}} \uparrow$
100	**12.95**	15.16	34.01
200	12.80	**16.19**	**35.20**
900	11.42	14.13	31.61

Among current-only methods, SparseOcc (Liu et al., 2024) is the most efficient, partly aided by an input resolution nearly $8\times$ lower than our full-resolution setting. Among 4D forecasting methods, LighTROcc achieves both the highest inference speed and the best accuracy. For a resolution-matched comparison, LighTROcc† runs about $1.8\times$ faster than SparseOcc while also achieving higher accuracy. Notably, the full-resolution LighTROcc maintains real-time inference despite processing high-resolution inputs and forecasting multiple frames in a single pass, while achieving the best performance. This highlights the effectiveness of our compact instance-centric formulation.

4.4 Ablation Studies

Ablation on 3D center localization. A key factor enabling accurate instance recognition and localization with a compact query set is the proposed attention-guided forward lifting. Because decoded queries primarily encode 2D visual evidence, directly regressing metric 3D centers without explicit spatial priors is highly underconstrained. Forward lifting eases this localization problem, and we further employ soft argmax rather than hard peak selection to obtain stable image-plane centers. To isolate the contribution of each design choice, we compare direct MLP regression, hard-argmax lifting, and our soft-argmax lifting. As shown in Table 3, direct regression yields the lowest IoU_{avg} and AP_{avg}, together with substantially lower AP_{ctr} and $\text{Recall}_{\text{loc}}$, indicating that poor 3D grounding also degrades overall occupancy prediction. Hard-argmax lifting improves both forecasting accuracy and center localization, but remains vulnerable to spurious outlier peaks in the attention map. In contrast, soft argmax aggregates the full attention distribution, providing more reliable center grounding and achieving the best performance across all four metrics. Figure 5 further shows that the learned attention maps capture the corresponding object regions, while soft argmax estimates their centers more reliably than hard argmax.

Query scaling and efficiency-accuracy trade-off. We analyze how query count affects instance-centric perception and forecasting. Many prior approaches based on fixed spatial anchors initialize 900 or more query candidates to ensure sufficient scene coverage (Lin et al., 2023a; Liu et al., 2023a; Hu et al., 2026). In contrast, our attention-guided forward lifting explicitly localizes queries, allowing LighTROcc to operate with a substantially smaller query set. Table 4 compares the resulting accuracy and computational overhead, with all variants trained using full-resolution inputs.

Empirically, 200 queries provide the best balance between accuracy and efficiency. Reducing the query count to 100 yields only a marginal speed improvement while slightly degrading AP and

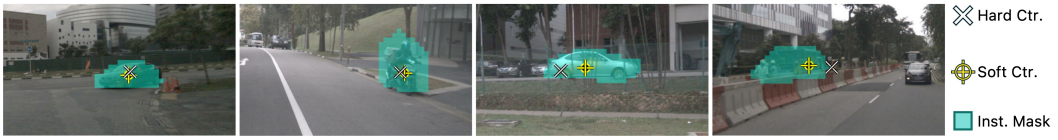

Figure 5: Qualitative comparison of hard- and soft-argmax localization from query attention maps. Hard argmax is sensitive to spurious outlier peaks, while soft argmax provides more reliable image-plane centers within the corresponding instance regions.

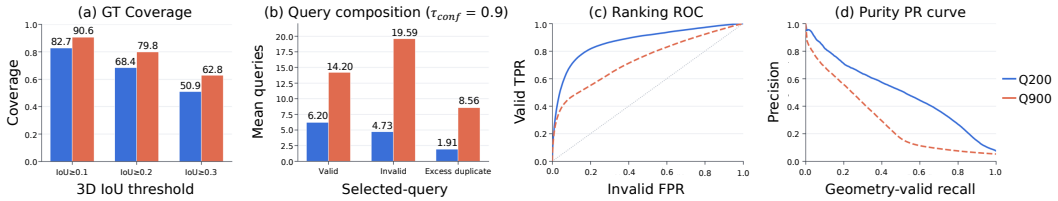

Figure 6: Statistical analysis of query scaling over the evaluation set. Queries with voxel IoU ≥ 0.2 are treated as geometry-valid. Although 900 queries improve GT coverage, they introduce more redundant and invalid predictions, whereas 200 queries provide stronger confidence discrimination and cleaner query selection.

IoU, indicating insufficient representation capacity. Conversely, increasing the query count by $4.5\times$ to 900, comparable to the scale commonly used in anchor-based methods, introduces substantial attention overhead and noticeably degrades prediction accuracy. This suggests that excessive query redundancy is not only computationally inefficient but can also hinder effective instance prediction. We provide further statistical analysis of query scaling below. We therefore adopt 200 queries as our default setting, which achieves the most favorable balance between accuracy and efficiency.

Statistical analysis of query scaling. We further analyze the effect of query scaling over the entire evaluation set. For this analysis, we regard a query as geometry-valid if its voxel IoU with a GT object is at least 0.2, and invalid otherwise. As shown in Figure 6 (a), increasing the number of queries from 200 to 900 consistently improves GT object coverage across different voxel-IoU thresholds. However, this increased coverage comes with substantial redundancy. Figure 6 (b) shows that the number of invalid queries grows considerably faster than the number of valid queries, while excess duplicate predictions also increase markedly. This excessive redundancy degrades the model's discriminative ability, making it harder to separate valid queries from redundant or invalid predictions based on confidence.

This behavior is reflected in the confidence statistics. In Figure 6 (c), the 200-query model exhibits a consistently stronger ROC curve, indicating better discrimination between valid and invalid predictions. Figure 6 (d) provides a complementary view: across confidence thresholds, the 200-query model captures a larger fraction of geometry-valid queries while admitting fewer invalid ones. Together, these results indicate that although additional queries improve scene coverage, excessive query redundancy weakens confidence-based selection and degrades instance-level prediction quality. This analysis supports our choice of 200 queries as a compact yet expressive query set.

5 Conclusion

We presented LighTROcc, a lightweight instance-centric framework for camera-based 4D occupancy forecasting. LighTROcc represents movable objects with a compact set of learned queries, avoiding dense spatiotemporal feature processing and large sets of spatially initialized candidates. Each query is localized through attention-guided forward lifting, modeled with a mixture of 3D Gaussians, and propagated to future frames using predicted displacements. This design predicts present and future occupancy in a single forward pass while maintaining instance-specific representations. Experiments on nuScenes and supplemented nuScenes-Occupancy show that LighTROcc outperforms all evaluated baselines in IoU and AP while achieving the highest inference speed among 4D forecasting methods. Ablations validate the localization and compact query design, demonstrating an accuracy-efficiency trade-off for real-time 4D occupancy forecasting.

AI USE STATEMENT

We used AI in a limited manner for grammar and style polishing only. It did not contribute to research ideation, experimental design, or analysis. The authors take full responsibility for all content.

REPRODUCIBILITY STATEMENT

We provide the model architecture and training objectives in Section 3, and the datasets and evaluation protocols in Section 4.1. Appendix A details the dataset construction and evaluation metrics, while Appendix B provides the network architecture and training implementation details.

REFERENCES

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 *2020 IEEE/CVF conference on computer vision and pattern recognition (CVPR)*, pp. 11618–11628. IEEE, 2020.

Anh-Quan Cao and Raoul de Charette. MonoScene: Monocular 3d semantic scene completion. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3991–4001, 2022.

Angel X. Chang, Thomas Funkhouser, Leonidas Guibas, Pat Hanrahan, Qixing Huang, Zimo Li, Silvio Savarese, Manolis Savva, Shuran Song, Hao Su, Jianxiong Xiao, Li Yi, and Fisher Yu. ShapeNet: An information-rich 3D model repository. Technical Report 1512.03012, arXiv preprint, December 2015.

Junliang Chen, Huaiyuan Xu, Yi Wang, and Lap-Pui Chau. OccProphet: Pushing the efficiency frontier of camera-only 4d occupancy forecasting with an observer-forecaster-refiner framework. In *International Conference on Learning Representations (ICLR)*, 2025.

Xuanyao Chen, Tianyuan Zhang, Yue Wang, Yilun Wang, and Hang Zhao. FUTR3D: A unified sensor fusion framework for 3d detection. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, pp. 172–181, 2023.

Bowen Cheng, Maxwell D Collins, Yukun Zhu, Ting Liu, Thomas S Huang, Hartwig Adam, and Liang-Chieh Chen. Panoptic-deeplab: A simple, strong, and fast baseline for bottom-up panoptic segmentation. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 12472–12482. IEEE, 2020.

Wanshui Gan, Fang Liu, Hongbin Xu, Ningkai Mo, and Naoto Yokoya. GaussianOcc: Fully self-supervised and efficient 3d occupancy estimation with gaussian splatting. In *IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 28980–28990, 2025.

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

Chunyong Hu, Qi Luo, Jianyun Xu, Song Wang, Qiang Li, and Sheng Yang. GUIDE: Gaussian unified instance detection for enhanced obstacle perception in autonomous driving. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 4816–4824, 2026.

Yuanhui Huang, Wenzhao Zheng, Yunpeng Zhang, Jie Zhou, and Jiwen Lu. Tri-perspective view for vision-based 3d semantic occupancy prediction. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 9223–9232, 2023.

Yuanhui Huang, Wenzhao Zheng, Yunpeng Zhang, Jie Zhou, and Jiwen Lu. GaussianFormer: Scene as gaussians for vision-based 3d semantic occupancy prediction. In *European Conference on Computer Vision (ECCV)*, pp. 376–393, 2024.

Yuanhui Huang, Amonnut Thammatadatrakoon, Wenzhao Zheng, Yunpeng Zhang, Dalong Du, and Jiwen Lu. GaussianFormer-2: Probabilistic gaussian superposition for efficient 3d occupancy prediction. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 27477–27486, 2025.

Haoyi Jiang, Tianheng Cheng, Naiyu Gao, Haoyang Zhang, Tianwei Lin, Wenyu Liu, and Xinggang Wang. Symphonize 3d semantic scene completion with contextual instance queries. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 20258–20267, 2024.

Haoyi Jiang, Liu Liu, Tianheng Cheng, Xinjie Wang, Tianwei Lin, Zhizhong Su, Wenyu Liu, and Xinggang Wang. GaussTR: Foundation model-aligned gaussian transformer for self-supervised 3d spatial understanding. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 11960–11970, 2025.

Yanqin Jiang, Li Zhang, Zhenwei Miao, Xiatian Zhu, Jin Gao, Weiming Hu, and Yu-Gang Jiang. PolarFormer: Multi-camera 3d object detection with polar transformer. In *AAAI Conference on Artificial Intelligence*, volume 37, pp. 1042–1050, 2023.

Harold W Kuhn. The hungarian method for the assignment problem. *Naval research logistics quarterly*, 2(1-2):83–97, 1955.

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 *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 9087–9098, 2023a.

Zhiqi Li, Wenhai Wang, Hongyang Li, Enze Xie, Chonghao Sima, Tong Lu, Yu Qiao, and Jifeng Dai. BEVFormer: Learning bird's-eye-view representation from multi-camera images via spatiotemporal transformers. In *European Conference on Computer Vision (ECCV)*, pp. 1–18, 2022.

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*, 2023b.

Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection. In *Proceedings of the IEEE international conference on computer vision*, pp. 2980–2988, 2017.

Xuewu Lin, Tianwei Lin, Zixiang Pei, Lichao Huang, and Zhizhong Su. Sparse4D: Multi-view 3d object detection with sparse spatial-temporal fusion. *arXiv preprint arXiv:2211.10581*, 2022.

Xuewu Lin, Tianwei Lin, Zixiang Pei, Lichao Huang, and Zhizhong Su. Sparse4D v2: Recurrent temporal fusion with sparse model. *arXiv preprint arXiv:2305.14018*, 2023a.

Xuewu Lin, Zixiang Pei, Tianwei Lin, Lichao Huang, and Zhizhong Su. Sparse4D v3: Advancing end-to-end 3d detection and tracking. *arXiv preprint arXiv:2311.11722*, 2023b.

Haisong Liu, Yao Teng, Tao Lu, Haiguang Wang, and Limin Wang. SparseBEV: High-performance sparse 3d object detection from multi-camera videos. In *IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 18580–18590, 2023a.

Haisong Liu, Yang Chen, Haiguang Wang, Zetong Yang, Tianyu Li, Jia Zeng, Li Chen, Hongyang Li, and Limin Wang. Fully sparse 3d occupancy prediction. In *European Conference on Computer Vision (ECCV)*, pp. 54–71, 2024.

Yingfei Liu, Tiancai Wang, Xiangyu Zhang, and Jian Sun. PETR: Position embedding transformation for multi-view 3d object detection. In *European Conference on Computer Vision (ECCV)*, pp. 531–548, 2022.

Yingfei Liu, Junjie Yan, Fan Jia, Shuailin Li, Aqi Gao, Tiancai Wang, and Xiangyu Zhang. PETRv2: A unified framework for 3d perception from multi-camera images. In *IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 3262–3272, 2023b.

Matthew Loper, Naureen Mahmood, Javier Romero, Gerard Pons-Moll, and Michael J. Black. SMPL: A skinned multi-person linear model. *ACM Trans. Graphics (Proc. SIGGRAPH Asia)*, 34(6):248:1–248:16, October 2015.

Ilya Loshchilov and Frank Hutter. Decoupled Weight Decay Regularization. In *International Conference on Learning Representations*, 2019. URL https://mlanthology.org/iclr/2019/loshchilov2019iclr-decoupled/.

Yuhang Lu, Xinge Zhu, Tai Wang, and Yuexin Ma. OctreeOcc: Efficient and multi-granularity occupancy prediction using octree queries. In *Advances in Neural Information Processing Systems (NeurIPS)*, pp. 79618–79641, 2024.

Junyi Ma, Xieyuanli Chen, Jiawei Huang, Jingyi Xu, Zhen Luo, Jintao Xu, Weihao Gu, Rui Ai, and Hesheng Wang. Cam4DOcc: Benchmark for camera-only 4d occupancy forecasting in autonomous driving applications. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 21486–21495, 2024a.

Qihang Ma, Xin Tan, Yanyun Qu, Lizhuang Ma, Zhizhong Zhang, and Yuan Xie. COTR: Compact occupancy transformer for vision-based 3d occupancy prediction. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 19936–19945, 2024b.

Gyeongrok Oh, Sungjune Kim, Heeju Ko, Hyung-gun Chi, Jinkyu Kim, Dongwook Lee, Daehyun Ji, Sungjoon Choi, Sujin Jang, and Sangpil Kim. 3d occupancy prediction with low-resolution queries via prototype-aware view transformation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 17134–17144, 2025.

Jinhyung Park, Chensheng Peng, Yihan Hu, Wenzhao Zheng, Kris Kitani, and Wei Zhan. S2go: Streaming sparse gaussian occupancy. In *International Conference on Learning Representations (ICLR)*, 2026.

Luis Roldão, Raoul de Charette, and Anne Verroust-Blondet. LMSCNet: Lightweight multiscale 3d semantic completion. In *International Conference on 3D Vision (3DV)*, pp. 111–119, 2020.

Seyed Sadegh Mohseni Salehi, Deniz Erdogmus, and Ali Gholipour. Tversky loss function for image segmentation using 3d fully convolutional deep networks. In *International workshop on machine learning in medical imaging*, pp. 379–387. Springer, 2017.

Yiang Shi, Tianheng Cheng, Qian Zhang, Wenyu Liu, and Xinggang Wang. Occupancy as set of points. In *European Conference on Computer Vision (ECCV)*, pp. 72–87, 2024.

Shuran Song, Fisher Yu, Andy Zeng, Angel X. Chang, Manolis Savva, and Thomas Funkhouser. Semantic scene completion from a single depth image. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1746–1754, 2017.

Wenchao Sun, Xuewu Lin, Yining Shi, Chuang Zhang, Haoran Wu, and Sifa Zheng. SparseDrive: End-to-end autonomous driving via sparse scene representation. In *IEEE International Conference on Robotics and Automation (ICRA)*, pp. 8795–8801, 2025.

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 *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 15035–15044, 2024.

Wenwen Tong, Chonghao Sima, Tai Wang, Li Chen, Silei Wu, Hanming Deng, Yi Gu, Lewei Lu, Ping Luo, Dahua Lin, and Hongyang Li. Scene as occupancy. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 8406–8415, October 2023.

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. In *Advances in Neural Information Processing Systems (NeurIPS)*, pp. 119861–119885, 2024a.

Shihao Wang, Yingfei Liu, Tiancai Wang, Ying Li, and Xiangyu Zhang. Exploring object-centric temporal modeling for efficient multi-view 3d object detection. In *IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 3621–3631, 2023a.

Xiaofeng Wang, Zheng Zhu, Wenbo Xu, Yunpeng Zhang, Yi Wei, Xu Chi, Yun Ye, Dalong Du, Jiwen Lu, and Xingang Wang. OpenOccupancy: A large scale benchmark for surrounding semantic occupancy perception. In *IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 17804–17813, 2023b.

Yue Wang, Vitor Campagnolo Guizilini, Tianyuan Zhang, Yilun Wang, Hang Zhao, and Justin Solomon. DETR3D: 3d object detection from multi-view images via 3d-to-2d queries. In *Conference on Robot Learning (CoRL)*, pp. 180–191, 2022.

Yuping Wang, Xiangyu Huang, Xiaokang Sun, Mingxuan Yan, Shuo Xing, Zhengzhong Tu, and Jiachen Li. Uniocc: A unified benchmark for occupancy forecasting and prediction in autonomous driving. In *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 25560–25570. IEEE, 2025.

Yuqi Wang, Yuntao Chen, Xingyu Liao, Lue Fan, and Zhaoxiang Zhang. PanoOcc: Unified occupancy representation for camera-based 3d panoptic segmentation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 17158–17168, 2024b.

Yi Wei, Linqing Zhao, Wenzhao Zheng, Zheng Zhu, Jie Zhou, and Jiwen Lu. SurroundOcc: Multi-camera 3d occupancy prediction for autonomous driving. In *IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 21729–21740, 2023.

Kaixin Xiong, Shi Gong, Xiaoqing Ye, Xiao Tan, Ji Wan, Errui Ding, Jingdong Wang, and Xiang Bai. CAPE: Camera view position embedding for multi-view 3d object detection. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 21570–21579, 2023.

Jingyi Xu, Xieyuanli Chen, Junyi Ma, Jiawei Huang, Jintao Xu, Yue Wang, and Ling Pei. Spatiotemporal decoupling for efficient vision-based occupancy forecasting. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 22338–22347, 2025.

Chenyu Yang, Yuntao Chen, Hao Tian, Chenxin Tao, Xizhou Zhu, Zhaoxiang Zhang, Gao Huang, Hongyang Li, Yu Qiao, Lewei Lu, Jie Zhou, and Jifeng Dai. BEVFormer v2: Adapting modern image backbones to bird's-eye-view recognition via perspective supervision. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 17830–17839, 2023.

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.

Tianyuan Zhang, Xuanyao Chen, Yue Wang, Yilun Wang, and Hang Zhao. MUTR3D: A multi-camera tracking framework via 3d-to-2d queries. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, pp. 4537–4546, 2022.

Yunpeng Zhang, Zheng Zhu, and Dalong Du. OccFormer: Dual-path transformer for vision-based 3d semantic occupancy prediction. In *IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 9433–9443, 2023.

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.

A Evaluation Protocol and Dataset Analysis

A.1 Evaluation Metrics

Occupancy and instance evaluation. In this section, we provide additional details on the evaluation protocol for the instance-level average precision (AP) and voxel-wise intersection over union (IoU) reported in the main paper. For instance-level AP, each predicted object is represented by a voxel occupancy mask and an instance confidence score. Predicted instances are matched to ground-truth instances based on their voxel IoU. A prediction is considered a true positive if its IoU with the matched ground-truth instance exceeds a threshold τ, while unmatched predictions are counted as false positives. We compute AP independently at $\tau \in \{0.1, 0.2, 0.3\}$ and report the mean across these thresholds for each temporal step.

Let AP_k denote the threshold-averaged AP at time step k, where $k = 0$ corresponds to the present frame and $k \in \{1, \ldots, T_f\}$ denotes a future frame. The reported temporal metrics are computed as

$$\mathrm{AP}_{\mathrm{cur}} = \mathrm{AP}_0, \qquad \mathrm{AP}_{\mathrm{fut}} = \frac{1}{T_f} \sum_{k=1}^{T_f} \mathrm{AP}_k, \tag{9}$$

and

$$\mathrm{AP}_{\mathrm{avg}} = \frac{1}{T_f + 1} \sum_{k=0}^{T_f} \mathrm{AP}_k. \tag{10}$$

Thus, $\mathrm{AP}_{\mathrm{fut}}$ measures forecasting accuracy exclusively over future frames, whereas $\mathrm{AP}_{\mathrm{avg}}$ summarizes performance over both the present and future states.

For voxel-wise evaluation, all GMO instances at each temporal step are merged into a binary foreground occupancy map. Given the predicted and ground-truth foreground voxel sets $\mathcal{P}_k$ and $\mathcal{G}_k$, respectively, we compute

$$\mathrm{IoU}_k = \frac{|\mathcal{P}_k \cap \mathcal{G}_k|}{|\mathcal{P}_k \cup \mathcal{G}_k|}. \tag{11}$$

$\mathrm{IoU}_{\mathrm{cur}}$, $\mathrm{IoU}_{\mathrm{fut}}$, and $\mathrm{IoU}_{\mathrm{avg}}$ are aggregated over the temporal dimension in the same manner as their AP counterparts.

For dense voxel-wise baselines, instance masks required for AP evaluation are recovered from their occupancy and flow predictions using the corresponding official grouping procedure. This conversion is used only for instance-level evaluation, while voxel-wise IoU is computed directly from the predicted foreground occupancy.

Center localization evaluation. In addition to occupancy prediction, we evaluate query-level center localization on the current frame. We report $\mathrm{AP}_{\mathrm{ctr}}$ and $\mathrm{Recall}_{\mathrm{loc}}$ at a 1 m threshold, measuring confidence-ranked center localization and active-query GT coverage, respectively. For $\mathrm{AP}_{\mathrm{ctr}}$, foreground queries are matched to ground-truth instances based on the center distance. Queries are processed in descending confidence order, and a prediction is considered a true positive if its distance to an unmatched ground-truth instance is within the threshold. The resulting precision-recall curve is used to compute $\mathrm{AP}_{\mathrm{ctr}}$. For $\mathrm{Recall}_{\mathrm{loc}}$, we evaluate active queries after inference-time filtering and match them with ground-truth instances using one-to-one radius matching. It measures the ratio of successfully localized ground-truth instances:

$$\mathrm{Recall}_{\mathrm{loc},\tau} = \frac{N_{\mathrm{matched}}^{d \le \tau}}{N_{\mathrm{GT}}}, \tag{12}$$

where $N_{\mathrm{matched}}^{d \le \tau}$ denotes the number of ground-truth instances matched within the distance threshold τ. Both metrics are computed using center distances.

A.2 Dataset Analysis and Ground Truth Supplementation

Following prior works (Ma et al., 2024a; Xu et al., 2025; Chen et al., 2025), we evaluate occupancy predictions using the ground-truth annotations from nuScenes-Occupancy (Wang et al., 2023b). Since its annotations are derived from LiDAR observations, however, they are inherently incomplete.

LiDAR primarily captures visible object surfaces rather than their complete volumetric extent, often resulting in hollow or incomplete occupancy structures even when the overall object shape appears visually plausible. This issue becomes more pronounced for distant or occluded objects, where only a small fraction of the object surface may be observed and the resulting occupancy can be severely fragmented. The upper row of Figure 7 (a) illustrates representative examples of such sparse annotations. Beyond their limited visual completeness, these annotations can also introduce undesirable biases in both training and evaluation.

Nearby objects generally contain denser occupancy observations and therefore provide relatively strong signal. In contrast, distant or low-visibility objects may occupy only a few voxels, providing substantially weaker signals. This imbalance can bias a model toward predicting well-observed nearby objects while suppressing predictions for difficult instances. As illustrated in Figure 7 (b), predicting the full spatial extent of a sparsely annotated object can introduce many apparent false-positive voxels, even when the predicted geometry is physically reasonable. Conversely, missing sparsely observed regions may incur only a limited penalty under voxel-based evaluation. This creates an unfavorable evaluation bias that can encourage prediction suppression for difficult objects. Such behavior is undesirable in autonomous driving, where reliably detecting distant and partially occluded objects is particularly important.

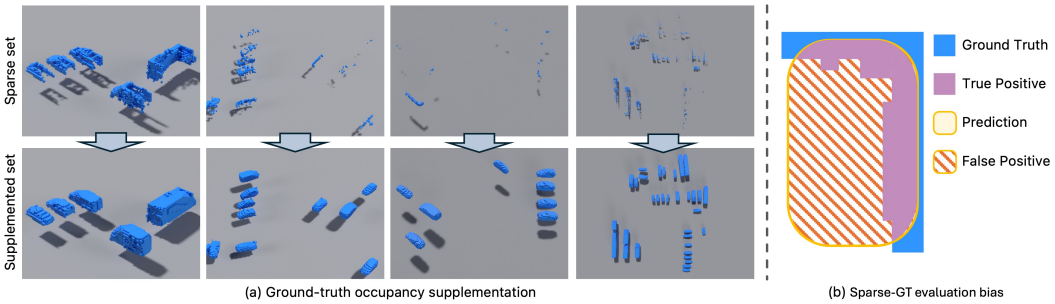

Figure 7: Illustration of ground-truth occupancy supplementation and evaluation bias from sparse annotations. (a) Original sparse occupancy annotations (top) and their supplemented counterparts using 3D assets (bottom). (b) Sparse ground truth can introduce apparent false-positive predictions even when the predicted geometry is physically reasonable.

To mitigate this issue, we supplement the original sparse occupancy annotations with 3D geometric prototypes from ShapeNet (Chang et al., 2015) and SMPL (Loper et al., 2015). For each object, we obtain its 3D bounding-box center, dimensions, orientation, and semantic category from the nuScenes annotations, and place a corresponding 3D mesh according to its estimated pose and extent. We use watertight meshes so that the enclosed volume can be treated as occupied rather than retaining only the visible surface. Importantly, the original nuScenes-Occupancy annotation is preserved rather than replaced. The asset-based occupancy is combined with the original annotation through a voxel-wise union, thereby retaining the observed object geometry while completing severely missing regions. The lower row of Figure 7 (a) visualizes the resulting supplemented annotations. This procedure provides more balanced evaluation across objects with different distances and visibility levels, while imposing a clearer penalty for missed instances and reducing the artificial false-positive penalty caused by sparse LiDAR observations.

To quantify the effectiveness of supplementation and its impact on evaluation fairness, we measure voxel density under different distance and visibility conditions. We analyze 518,702 3D object annotations from 29,049 training and evaluation sequences. The voxel density is computed as

$$D = \frac{\sum_i |S_i \cap B_i|}{\sum_i |B_i|}, \tag{13}$$

where S_i denotes either the original sparse voxel set or the supplemented voxel set of object i, and B_i denotes the set of voxels inside its ground-truth 3D bounding box from nuScenes. As shown in Table 5, the overall density increases from 17.62% to 71.51% ($4.06\times$).

When grouped by distance, the original annotations become increasingly sparse as objects move farther away, with density falling below 14% beyond 30 m and reaching only 8.11% beyond 50 m.

Consequently, even predictions that correctly cover the object location and extent can incur a large number of apparent false-positive voxels under the original annotations. In contrast, the supplemented annotations maintain substantially more stable densities across distance, ranging only from 74.06% to 68.90%, thereby providing more consistent evaluation for distant objects. A similar trend is observed across visibility levels, where supplementation substantially reduces the density gap between poorly and highly visible objects. These results show that the supplemented annotations effectively mitigate the distance- and visibility-dependent sparsity of the original annotations, reducing the resulting evaluation bias and enabling a more balanced assessment across different observation conditions.

Table 5: Occupancy density before and after 3D asset supplementation. Gain denotes the ratio of supplemented to original occupancy density.

Subset	Range	Original (%)	Supplemented (%)	Gain (×)
Overall	All	17.62	**71.51**	**4.06**
Distance (m)	$[0, 10)$	28.07	74.06	2.64
	$[10, 20)$	23.39	72.73	3.11
	$[20, 30)$	17.63	71.59	4.06
	$[30, 40)$	13.95	70.79	5.08
	$[40, 50)$	11.55	69.97	6.06
	≥ 50	8.11	68.90	8.50
Visibility (%)	V1 (0 to 40)	11.63	69.19	5.95
	V2 (40 to 60)	14.54	70.24	4.83
	V3 (60 to 80)	16.94	71.15	4.20
	V4 (80 to 100)	21.94	73.24	3.34

A.3 Geometric Consistency of Supplemented Annotations

In this section, we perform complementary geometric checks to validate the supplemented annotations against the original nuScenes-Occupancy (Wang et al., 2023b) annotations and LiDAR observations. We first measure the surface distance to quantify how closely the added assets fit the original LiDAR-derived occupied voxels. For each annotated object, let $\mathcal{S}_i$ denote the centers of its original occupied voxels and $\mathcal{M}_i$ the surface of the fitted asset. For each $\mathbf{x} \in \mathcal{S}_i$, we compute the nearest-surface distance as

$$d(\mathbf{x}, \mathcal{M}_i) = \min_{\mathbf{y} \in \mathcal{M}_i} \|\mathbf{x} - \mathbf{y}\|_2. \tag{14}$$

We report the mean and 90th-percentile distances, where smaller values indicate closer agreement with the geometry captured by the original observations.

We additionally measure the free-space violation rate to assess whether the supplemented geometry introduces excessive occupancy beyond the observed object geometry. We trace the original LiDAR rays and define the observed free space along each ray up to its measured return range as

$$\mathcal{F}(r) = \{\mathbf{s} + t\mathbf{d} \mid 0 \leq t < r\}, \tag{15}$$

where $\mathbf{s}$ is the LiDAR sensor origin, $\mathbf{d}$ is the unit ray direction, r is the measured return range, and t is the distance from the sensor origin along the ray. We then measure the fraction of supplemented occupancy that overlaps this observed free space, where a lower value indicates less conflict with the original observations.

As shown in Table 6, the fitted assets achieve a mean nearest-surface distance of 0.20 m and a $P90$ distance of 0.38 m. Given the 0.2 m voxel resolution used for evaluation, these correspond to approximately 1.0 and 1.9 voxels, respectively. The free-space violation rate is only 2.4%, indicating that the supplemented geometry rarely conflicts with directly observed free space. Together with the density analysis in Appendix A.2, these results show that the supplementation increases geometric completeness while remaining largely consistent with the original sensor evidence.

Table 6: Geometric consistency of supplemented annotations. Surface distance measures agreement with occupied voxels, while free-space violation measures overlap with LiDAR-observed free space.

Metric	Result
Nearest-surface distance, mean ↓	0.20 m
Nearest-surface distance, $P90$ ↓	0.38 m
LiDAR free-space violation ↓	2.4%

A.4 Robustness to Ground-Truth Annotation Protocol

To examine whether the comparative advantage of LighTROcc depends on the supplemented annotations, we additionally evaluate all methods using the original sparse nuScenes-Occupancy annotations. As analyzed in Appendix A.2, these annotations cover only 17.62% of the object bounding-box volume on average and become increasingly sparse with distance. Consequently, even geometrically plausible predictions may be counted as false positives when they occupy regions that are simply unobserved in the sparse ground truth.

To mitigate this annotation-induced penalty, we adopt a conservative bbox-aware evaluation protocol. Predicted voxels that lie inside annotated 3D bounding boxes but are absent from the sparse ground truth are excluded from false-positive counting, without being reassigned as true positives. The resulting IoU is computed as

$$\mathrm{IoU} = \frac{TP}{TP + FN + FP - FP^{\mathrm{bbox}}}, \tag{16}$$

where FP^{bbox} denotes false-positive voxels inside the annotated 3D bounding boxes. The same ignore mask is applied when computing voxel overlaps for instance-level AP. Thus, the protocol does not assume that the entire bounding box is occupied, but only avoids treating unlabeled in-box regions as reliable negative evidence.

As shown in Table 7, LighTROcc retains the highest aggregate performance, achieving $\mathrm{IoU}_{\mathrm{avg}} = 9.16$ and $\mathrm{AP}_{\mathrm{avg}} = 13.93$. Although SparseOcc achieves a slightly higher current-frame IoU, LighTROcc obtains the best future and average IoU and the best AP across all temporal aggregations. Moreover, Table 1 shows that all evaluated methods achieve higher average IoU and AP under the supplemented annotations than under the original sparse-label protocol. This consistent improvement across methods suggests that supplementation alleviates a common sparsity-induced evaluation penalty rather than providing an advantage specific to LighTROcc.

Together with the density analysis in Appendix A.2, these results provide complementary evidence for our evaluation design. The original sparse annotations can introduce unintended evaluation bias due to their incomplete object coverage. Although the bbox-aware evaluation mitigates this issue by excluding ambiguous in-box false positives, it still provides no positive target for unobserved object regions. Consequently, missing predictions in these regions are not counted as false negatives, particularly for distant or occluded objects where the annotations are severely incomplete. To address both limitations, we adopt the supplemented annotations as a more complete evaluation target. The quantitative analyses and comparative experiments collectively support the rationale and effectiveness of this supplementation strategy.

B Implementation Details

B.1 Network Architecture

We use a ResNet-18 (He et al., 2016) initialized with pretrained weights as the image backbone for feature extraction. The extracted image features are projected to the same embedding dimension as the instance queries, with $D = 128$, before being processed by the transformer decoder. We initialize a compact set of 200 learnable instance queries. The transformer decoder consists of $L = 3$ layers with $H = 4$ attention heads. To encode the spatial, camera-specific, and temporal information of the

Table 7: Quantitative comparison of dense voxel-wise and instance-wise methods under the original sparse nuScenes-Occupancy annotations using the bbox-aware evaluation protocol. Best and second-best results are shown in bold and underlined, respectively.

Method	Type	IoU_{cur}	IoU_{fut}	IoU_{avg}	AP_{cur}	AP_{fut}	AP_{avg}
Cam4DOcc (Ma et al., 2024a)	Dense Voxel wise	11.23	3.84	5.32	8.70	2.80	3.98
EfficientOCF (Xu et al., 2025)		1.52	1.43	1.45	<u>11.56</u>	<u>10.94</u>	<u>11.06</u>
OccProphet (Chen et al., 2025)		10.50	<u>8.09</u>	8.57	11.00	8.13	8.70
SparseOcc (Liu et al., 2024)	Instance wise	**11.91**	8.06	<u>8.83</u>	9.48	8.14	8.41
GUIDE (Hu et al., 2026)		6.48	5.68	5.84	10.71	9.77	9.96
LighTROcc (Ours)		<u>11.42</u>	**8.59**	**9.16**	**17.56**	**13.02**	**13.93**

image tokens, we incorporate camera-parameter embeddings together with 2D sinusoidal positional encodings, learned camera-ID embeddings, and learned temporal embeddings.

The depth head is implemented as a lightweight two-layer multilayer perceptron (MLP) that predicts a categorical depth distribution over $N_d = 64$ bins within the range of $[2, 58]$ m. The Gaussian head is likewise implemented using a two-layer MLP and predicts the parameters of 48 Gaussian components for each instance. These components are converted to an occupancy field as

$$P_{q,k}(\mathbf{x}) = 1 - \prod_{g=1}^{N_g} \left[1 - \alpha_{q,g} \exp\left(-\frac{1}{2} (\mathbf{x} - \mu_{q,g,k})^\top \Sigma_{q,g}^{-1} (\mathbf{x} - \mu_{q,g,k}) \right) \right]. \tag{17}$$

Here, $\alpha_{q,g} \in [0, 1]$ denotes the occupancy weight of each Gaussian. The trajectory and confidence heads are also lightweight two-layer MLPs. The trajectory head predicts four displacement vectors corresponding to the future forecasting steps, while the confidence head estimates the existence probability of each query.

B.2 Training Details

We train LighTROcc for 20 epochs using the AdamW optimizer (Loshchilov & Hutter, 2019) with an initial learning rate of 3×10^{-4} and a weight decay of 0.01. The learning rate is scheduled using cosine annealing. Training is conducted on eight NVIDIA RTX Pro 6000 Blackwell GPUs with a batch size of one per GPU.

For Hungarian matching, we use the 3D center distance, attention overlap cost, and confidence cost with weights of 10.0, 0.3, and 0.2, respectively. For the training objective in Eq. (8), we set $(\lambda_{\text{conf}}, \lambda_{\text{cen}}, \lambda_{\text{depth}}, \lambda_{\text{attn}}, \lambda_{\text{occ}}, \lambda_{\text{traj}}) = (1.0, 0.3, 1.0, 1.0, 1.0, 1.0)$. For trajectory prediction, we augment each query feature with motion priors derived from the observed frames. Specifically, given three observed frames $(t-2, t-1, t)$, we compute two displacement vectors corresponding to $t-2 \rightarrow t-1$ and $t-1 \rightarrow t$. These observed displacements are concatenated with the query feature and fed to the trajectory head, which predicts four future displacement vectors. To improve training stability, we employ teacher forcing during early training and gradually transition to prediction-based training. Under teacher forcing, the motion priors are computed from ground-truth instance centers, whereas prediction-based training uses predicted centers. The teacher-forcing ratio is progressively reduced during warm-up and reaches zero from epoch 8 onward, after which all motion priors are derived from the predicted centers.

C Additional Experiments and Analysis

C.1 Flow-Based Instance Recovery in Dense Voxel-wise Models

In this section, we further analyze the instance segmentation behavior of dense voxel-wise occupancy forecasting methods. As discussed in the main paper, methods such as Cam4DOcc (Ma et al., 2024a), EfficientOCF (Xu et al., 2025), and OccProphet (Chen et al., 2025) are not inherently designed in an instance-centric manner and therefore rely on an additional flow prediction module to recover object identities from dense occupancy maps. These methods typically predict backward

centripetal flow, where each occupied voxel is encouraged to point toward the center of its corresponding object in the previous frame. The instance identity of a voxel is then determined according to the object center reached by its predicted flow. Voxels whose flows do not reach a valid object center cannot be reliably associated with any instance and may consequently remain unassigned. As a result, the instance segmentation quality of these dense methods is closely coupled with the accuracy of their flow predictions.

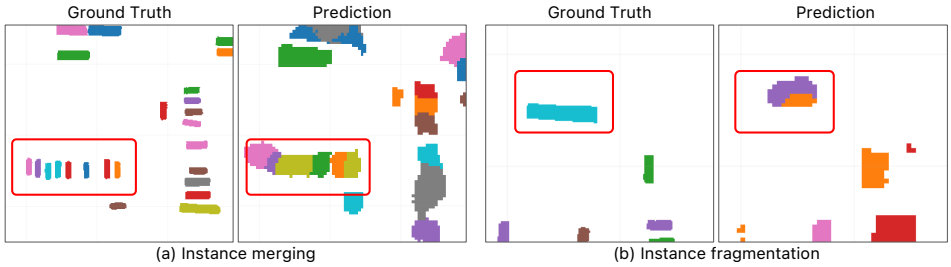

Figure 8: Qualitative examples of failure modes in flow-based instance recovery for dense voxel-wise models. (a) Instance merging, where voxels from multiple nearby objects are incorrectly assigned to a single identity. (b) Instance fragmentation, where voxels belonging to one object are split into multiple identities. Red boxes highlight representative errors.

As shown in Figure 8, even small flow errors can cause voxels to inherit incorrect instance identities, particularly in crowded scenes with nearby objects. This can merge multiple objects into a single instance or fragment one object into multiple identities, increasing both false-positive and false-negative predictions. Such association errors consequently degrade AP, which reflects both occupancy overlap and instance separation. These observations highlight the limitation of indirectly recovering object identities from dense occupancy and flow, and motivate the instance-centric design of LighTROcc, which maintains object-specific representations directly throughout forecasting.

C.2 Temporal Shape Consistency

A key advantage of LighTROcc is that its instance-centric representation preserves the predicted geometry of each object throughout the forecasting horizon. As discussed in Section C.1, dense voxel-wise methods recover instance identities indirectly from predicted flow. Flow errors can assign voxels to incorrect objects or leave them unassigned, resulting in fragmented instance masks and unstable object extents across frames. In contrast, LighTROcc propagates each predicted Gaussian mixture using the displacement vectors while retaining its component offsets, covariances, and occupancy weights, thereby preserving the shape and extent of each instance across future frames.

Following UniOcc's object-aligned shape consistency evaluation (Wang et al., 2025), we compute Temporal Shape Consistency (TSC) to measure the preservation of predicted object geometry across consecutive frames, independent of global translation and orientation. Let $\hat{O}_i^k$ denote the occupied voxel set of instance i at time step k, and $\mathcal{K}_i$ its evaluated consecutive time steps. After transforming each shape into an object-local coordinate system through centroid subtraction and PCA-based alignment, denoted by $\mathcal{A}(\cdot)$, the TSC is computed as

Table 8: Temporal shape consistency of 4D occupancy forecasting methods. Higher TSC indicates more stable geometry over time.

Method	TSC (%) ↑
Cam4DOcc	51.04
EfficientOCF	<u>65.67</u>
OccProphet	36.38
LighTROcc (Ours)	**91.32**

$$\mathrm{TSC} = \frac{1}{|\mathcal{I}|} \sum_{i \in \mathcal{I}} \frac{1}{|\mathcal{K}_i|} \sum_{k \in \mathcal{K}_i} \mathrm{IoU}\left(\mathcal{A}(\hat{O}_i^k), \mathcal{A}(\hat{O}_i^{k+1})\right), \tag{18}$$

where $\mathcal{I}$ denotes the set of evaluated predicted instances.

As shown in Table 8, dense voxel-wise baselines exhibit lower temporal shape consistency, reflecting greater variation in instance geometry across frames. LighTROcc achieves a TSC of 91.32, compared with 65.67 for EfficientOCF. This result reflects the temporal stability encouraged by the instance-centric representation.

Table 9: Forecasting performance across future horizons. LighTROcc (constant vel.) uses constant-velocity extrapolation instead of learned trajectory prediction. Best and second-best results are shown in bold and underlined, respectively.

Method	Type	IoU				AP			
		0.5 s	1.0 s	1.5 s	2.0 s	0.5 s	1.0 s	1.5 s	2.0 s
Cam4DOcc (Ma et al., 2024a)	Dense Voxel-wise	13.02	12.11	11.39	10.68	9.39	8.17	7.49	7.03
EfficientOCF (Xu et al., 2025)		9.16	8.77	8.41	8.02	15.36	14.62	13.75	12.78
OccProphet (Chen et al., 2025)		16.21	<u>14.46</u>	<u>13.47</u>	<u>12.78</u>	10.67	7.42	5.95	4.96
LighTROcc (Ours)	Instance-wise	**17.15**	**15.76**	**14.62**	**13.67**	**39.35**	**35.15**	**31.16**	**27.70**
LighTROcc (constant vel.)		<u>16.30</u>	12.72	12.25	11.18	<u>35.13</u>	<u>27.49</u>	<u>23.66</u>	<u>23.19</u>

C.3 Performance at Different Forecasting Horizons

To complement the aggregate results in Table 1 of the main paper, we report IoU and AP at each future forecasting horizon. The current-only instance-wise baselines are excluded from this comparison because they do not explicitly forecast future occupancy states. As shown in Table 9, all evaluated methods exhibit a gradual decrease in both IoU and AP as the forecasting horizon increases, reflecting the greater difficulty of longer-term occupancy prediction. Nevertheless, LighTROcc consistently achieves the best performance at every horizon, with particularly large margins in instance-level AP. Notably, at the longest horizon of $2.0\,s$, LighTROcc reaches an IoU of 13.67 and an AP of 27.70, outperforming the strongest prior baselines by 0.89 IoU and 14.92 AP points, respectively. These results demonstrate that the proposed instance-centric formulation retains a consistent forecasting advantage across all evaluated horizons.

To further isolate the contribution of the trajectory head, we replace the learned future displacements with constant-velocity extrapolation while retaining the same current instance predictions and Gaussian geometry. As shown in Table 9, the constant-velocity variant consistently underperforms LighTROcc across all forecasting horizons. From 0.5 to 2.0 s, LighTROcc decreases from 17.15 to 13.67 IoU (-20.3%) and from 39.35 to 27.70 AP (-29.6%), whereas constant velocity decreases from 16.30 to 11.18 IoU (-31.4%) and from 35.13 to 23.19 AP (-34.0%). These results demonstrate the benefit of learned step-specific motion over linear extrapolation.

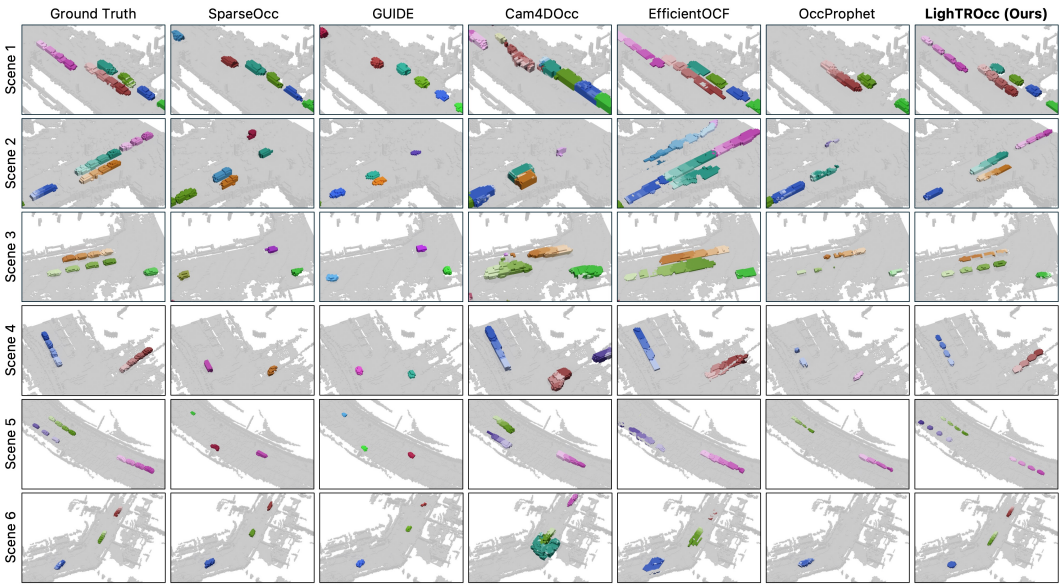

Figure 9: Additional qualitative comparison of GMO occupancy predictions with a broader set of baselines. LighTROcc consistently captures object geometry and temporal evolution across diverse scenes. Colors become progressively darker for later forecasting steps.

D Additional Qualitative Results

Figure 9 provides additional qualitative comparisons with a broader set of baselines to complement the results in the main paper. For a comprehensive comparison, we visualize predictions from all evaluated baselines. Since SparseOcc (Liu et al., 2024) and GUIDE (Hu et al., 2026) are current-only methods, their results are shown only for the present frame. Although these methods achieve competitive object localization at the current frame, they cannot model future occupancy evolution and therefore provide no predictions over the forecasting horizon. In contrast, the 4D forecasting methods explicitly reconstruct future occupancy states. Cam4DOcc (Ma et al., 2024a) produces temporally extended predictions, visualized from lighter to darker colors toward later steps, but tends to overestimate object extents. EfficientOCF (Xu et al., 2025) exhibits a similar overestimation tendency and, as shown in Scene 2, can merge nearby objects into a single instance. OccProphet (Chen et al., 2025) generally captures object extent more accurately, but its flow-based instance recovery is less stable, often causing object occupancy to progressively disappear at later forecasting steps. In comparison, LighTROcc consistently preserves instance geometry while accurately modeling future object evolution across diverse scenes. These qualitative results further demonstrate the effectiveness of the proposed instance-centric forecasting formulation.