INSTANCE-CENTRIC UNSUPERVISED MONOCULAR 3D OCCUPANCY PREDICTION

Anonymous authors
Paper under double-blind review

ABSTRACT

Unsupervised monocular 3D occupancy prediction aims to recover dense scene geometry from a single image without 3D dense occupancy annotations, but often produces blurred object boundaries and trailing artifacts under sparse and indirect image-based supervision. We attribute these limitations to the prevailing scene-level representation, which learns unified geometry without explicitly distinguishing individual object geometries, thereby overlooking the 3D shape modeling of individual objects. To address this issue, we propose ICOcc, an instance-centric unsupervised occupancy prediction framework, which reformulates occupancy learning from scene-level reconstruction to instance-centric prediction. Specifically, an instance-centric categorical occupancy formulation directly separates the entangled object geometries, without the need for a unified scene geometry. Based on this formulation, an instance decoupling network is introduced, where learnable queries are employed to capture shape priors and decouple object geometries from scene-level visual features. To supervise the decoupled outputs, an instance-anchored learning strategy further provides stronger constraints through multi-view supervision across dynamically selected views. Extensive experiments on KITTI-360 and CarlaOcc show that ICOcc improves occupancy prediction, achieving a 10.2% relative gain in invisible-empty recall on KITTI-360, while producing clearer object boundaries and fewer trailing artifacts.

1 INTRODUCTION

Vision-centric spatial perception (Zhou et al., 2026; Zhang et al., 2026; Cao & Vu, 2026) is a crucial yet challenging problem for autonomous systems (Jia et al., 2026; Strong et al., 2026). Monocular 3D occupancy prediction (Xue et al., 2026; Li et al., 2025) that infers a dense 3D volumetric scene representation from a single image is a key task for spatial perception. Such a representation captures occupied scene geometry, visible free space in front of foreground objects, and occluded regions behind them, providing a flexible and comprehensive description of scene geometry for downstream tasks such as forecasting (Du et al., 2026) and planning (Liu et al., 2026).

Existing vision-centric 3D occupancy prediction methods can be broadly divided into supervised approaches (Zhang et al., 2023; Jiang et al., 2024) and unsupervised approaches (Han et al., 2024; Li et al., 2024a). Supervised approaches require dense occupancy annotations, which are typically generated from sparse LiDAR point clouds (Wang et al., 2025; Tian et al., 2023; Wei et al., 2023; Wang et al., 2023). Such annotations are expensive to acquire and susceptible to labeling inaccuracies (Liu et al., 2025). In contrast, unsupervised approaches (Wimbauer et al., 2023; Feng et al., 2025) generate training signals from multi-view or temporally adjacent RGB images, commonly through neural rendering techniques such as Neural Radiance Fields (NeRFs) (Mildenhall et al., 2020). Since collecting such images is considerably easier than annotating dense 3D voxels, unsupervised learning provides a promising path toward scalable 3D occupancy pretraining.

However, generating effective supervision from images remains challenging, as these training signals are inherently sparse and indirect. Recent methods therefore incorporate additional priors, including semantic priors (Li et al., 2024a) and depth pseudo labels (Feng et al., 2025). For semantic occupancy, unsupervised approaches (Huang et al., 2024a; Jiang et al., 2025) learn 3D semantic predictions from 2D pseudo semantic labels through volume rendering. Despite the advances in generating supervision signals and incorporating semantic priors, existing methods still struggle to

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.

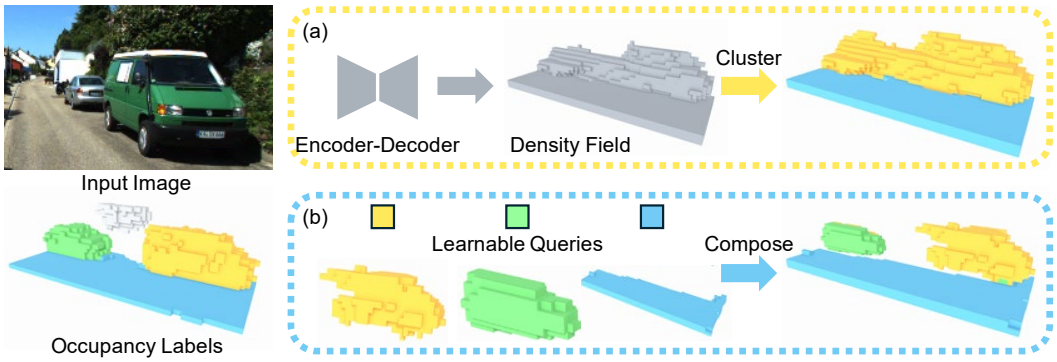


Figure 1: Comparison between SceneDINO (a) and ICOcc (b). Unlike SceneDINO, which first learns a complex unified scene geometry for semantic prediction, ICOcc predicts occupancy by composing geometries from different learnable queries. This instance-centric pipeline exploits shared 3D shape priors to recover clearer object boundaries and mitigate trailing artifacts.

recover clear object boundaries and frequently produce trailing artifacts behind foreground objects, as illustrated in Fig. 1. We attribute these limitations to the prevailing scene-centric formulation. During volume rendering, multiple objects and background regions at different distances along a camera ray are aggregated into a reconstructed pixel before supervision is applied. This entangles different object geometries along the ray, making their boundaries difficult to recover under sparse image supervision. A natural way to alleviate this entanglement is to model object geometries separately at the instance level, so that the shared 3D shape priors of same-category objects can be exploited during learning. Yet such instance-centric occupancy learning poses challenges in representing, predicting, and supervising instance-level geometry without 3D occupancy annotations.

Therefore, we introduce **I**nstance-**C**entric Unsupervised **Occ**upancy Prediction (**ICOcc**), which reformulates occupancy learning from scene-level reconstruction to instance-centric prediction. To resolve the entanglement of object geometries, we propose an **Instance-Centric Categorical Occupancy Formulation** that decomposes scene occupancy into non-empty occupancy slots and empty space, allowing individual object instances to be represented by dedicated slots. However, visual features extracted from an input image remain scene-level and do not naturally correspond to separated object geometries. We therefore develop an **Instance Decoupling Network**, where learnable queries capture object-level geometric priors and decouple object geometries from scene-level features. Finally, to effectively supervise the separated predictions from sparse image-space signals, we introduce an **Instance-Anchored Learning Strategy** that constructs multi-view supervision around co-visible instances, providing geometric constraints for improved instance separation and cross-view consistency. Extensive experimental results are conducted on KITTI-360 (Liao et al., 2022) and CarlaOcc (Feng et al., 2026). On KITTI-360, ICOcc improves invisible-empty recall by 10.2% relative to the strongest baseline. Together with the qualitative results, this indicates fewer trailing artifacts and clearer object boundaries, validating the state-of-the-art (SoTA) performance of our framework. The key contributions are summarized as follows:

- We introduce **ICOcc**, an instance-centric framework that reformulates monocular 3D occupancy prediction from scene reconstruction to individual object geometry learning.
- We propose an **Instance-Centric Categorical Occupancy Formulation**, which explicitly separates object geometries, mitigating their entanglement during rendering.
- We develop an **Instance Decoupling Network** that employs learnable queries in a query-based Transformer decoder to capture object-level geometric priors, enabling instance-centric occupancy prediction from scene-level visual features.
- We introduce an **Instance-Anchored Learning Strategy** that establishes multi-view supervision around co-visible instances and employs dedicated objectives to preserve instance decomposition, improve object boundaries, and encourage cross-view consistency.

2 RELATED WORKS

Instance-Centric Supervised Occupancy Prediction. Early supervised methods predict voxel-wise semantics at scene level (Tian et al., 2023; Tang et al., 2024; 2026; Zhao et al., 2024; Huang et al., 2024b; Wang et al., 2024a). Recent works have resorted to incorporating object-level cues: Symphonies (Jiang et al., 2024) introduces contextual instance queries, while Ocean (Wang et al., 2026) learns object-centric features for semantic scene completion. Beyond semantic occupancy, PanoOcc (Wang et al., 2024c) and SparseOcc (Liu et al., 2024) explicitly predict panoptic occupancy with both semantic and instance information, while PaSCo (Cao et al., 2024) and IPFormer (Gross et al., 2026) extend semantic scene completion toward panoptic scene completion. Zheng et al. (2024) further explores object-centric occupancy to model fine-grained object geometry. These methods demonstrate the benefit of incorporating instance-level structure, but require additional dense instance information that is not natively provided by mainstream occupancy annotations (Tian et al., 2023; Wei et al., 2023; Wang et al., 2025). To obtain instance-level supervision, SparseOcc (Liu et al., 2024) and Zheng et al. (2024) construct object occupancy using ground-truth 3D boxes, whereas PaSCo (Cao et al., 2024) and IPFormer (Gross et al., 2026) rely on pseudo panoptic labels derived by clustering. The lack of native instance-level annotations has also motivated dedicated benchmarks such as CarlaOcc (Feng et al., 2026), which generates fine-grained instance occupancy from meshes in the CARLA simulator (Dosovitskiy et al., 2017). However, such synthetic data remain limited by simulation realism, diversity, and the sim-to-real domain gap. These annotation dependencies hinder scalable supervised learning of instance-centric geometry.

Unsupervised Occupancy Prediction. Unsupervised approaches avoid dense 3D occupancy annotations by constructing training signals from multi-view or temporally adjacent RGB images, commonly through neural rendering (Wimbauer et al., 2023; Han et al., 2024; Huang et al., 2024a). However, such image-based supervision is inherently sparse and indirect. Recent methods therefore incorporate additional priors, including semantic priors (Li et al., 2024a) and depth pseudo labels (Feng et al., 2025), to provide stronger geometric constraints. For semantic occupancy, recent approaches further learn 3D semantic predictions from 2D pseudo semantic labels or foundation-model features (Huang et al., 2024a; Jiang et al., 2025; Jevtić et al., 2025). In contrast to supervised methods that explicitly model object-level structure, existing unsupervised approaches generally do not exploit the shared 3D shape priors of same-category objects. Despite increasingly rich supervision signals and semantic priors, they still struggle to recover clear object boundaries and frequently produce trailing artifacts behind foreground objects.

3 METHODOLOGY

3.1 PROBLEM SETUP AND FRAMEWORK OVERVIEW

Given a set of multi-view RGB images $\{\boldsymbol{I}_v\}_{v=0}^{V}$ during training, our goal is to train a model that predicts 3D occupancy from a single image, without any 3D occupancy annotations. During training, one image, denoted by $\boldsymbol{I}_0$, is selected as the input view, while the remaining views $\{\boldsymbol{I}_v\}_{v=1}^{V}$ provide image-based supervision, with their known camera intrinsics $\{\boldsymbol{K}_v\}_{v=1}^{V}$ and extrinsics $\{\boldsymbol{T}_v\}_{v=1}^{V}$ enabling cross-view projection. During inference, the model takes only $\{\boldsymbol{I}_0, \boldsymbol{K}_0\}$ as input, and predicts 3D occupancy of both visible and occluded regions within a predefined spatial range $\mathcal{R}$.

Given a query point $\boldsymbol{x} \in \mathcal{R}$, existing prior methods predict a scalar density $\{\boldsymbol{I}_0, \boldsymbol{K}_0, \boldsymbol{x}\} \mapsto \sigma(\boldsymbol{x}) \in \mathbb{R}^+$. In contrast, ICOcc predicts an occupancy probability vector: $\{\boldsymbol{I}_0, \boldsymbol{K}_0, \boldsymbol{x}\} \mapsto \boldsymbol{p}(\boldsymbol{x}) \in [0,1]^{Q+1}$, where $\boldsymbol{p}(\boldsymbol{x})$ denotes the probabilities over Q occupancy slots and one empty-space state. Occupancy slots have no predefined semantics: instances and non-instance regions are dynamically assigned to different slots through the matching process. In $\boldsymbol{p}(\boldsymbol{x})$, for $q \in \{1, \ldots, Q\}$, each component $p_q(\boldsymbol{x})$ denotes the occupancy probability of the q-th occupancy slot, while $p_0(\boldsymbol{x})$ denotes the empty probability. The probabilities on occupancy slots can be aggregated to obtain scene-level occupancy $p_{\text{occ}}(\boldsymbol{x})$ and directly determine the predicted slot label $\hat{q}(\boldsymbol{x}) = \arg\max_{q \in \{0,\ldots,Q\}} p_q(\boldsymbol{x})$.

Based on the problem setup above, ICOcc comprises three key components, as illustrated in Fig. 2. The Instance Decoupling Network first takes an input image and 3D query points and predicts Q occupancy compatibility scores at each queried 3D location. Under the proposed Instance-Centric Categorical Occupancy Formulation, these scores converted into a categorical occupancy distribution

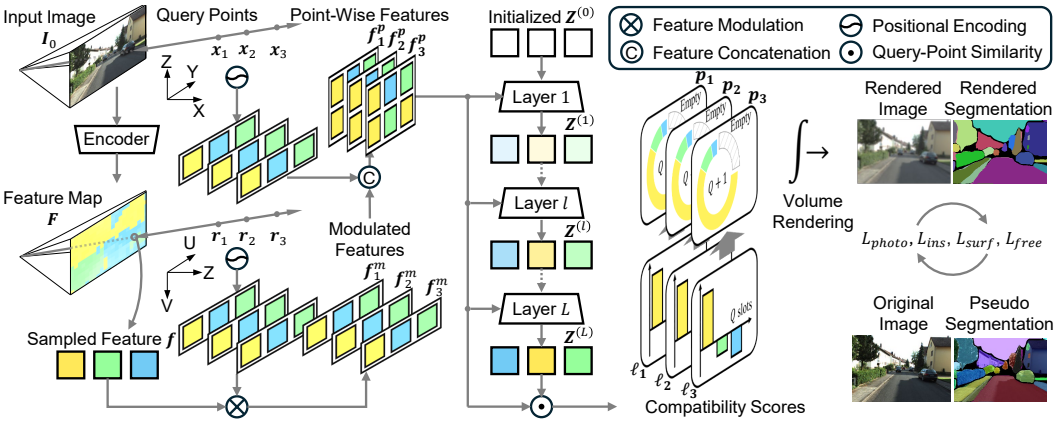


Figure 2: The architecture of ICOcc. Given an input image and sampled 3D query points, ICOcc progressively constructs point-wise features and decomposes scene occupancy through a set of learnable queries. The resulting categorical occupancy is rendered into RGB images and segmentation maps and optimized with image-based supervision across dynamically selected views.

over individual instances, non-instance regions, and empty space. Scene-level occupancy and opacity are then derived from this distribution to render reconstructed images and segmentation maps. To learn the categorical occupancy, the Instance-Anchored Learning Strategy constructs multi-view supervision around co-visible instances and introduces instance matching and geometric constraints across views to prevent predictions from degenerating into local scene occupancy.

3.2 Instance-Centric Categorical Occupancy Formulation

In existing approaches (Wimbauer et al., 2023), N query points $\{\boldsymbol{x}_i = \boldsymbol{o} + t_i\boldsymbol{d}\}_{i=1}^{N}$ are sampled along the camera ray corresponding to each pixel, where $\boldsymbol{o}$ and $\boldsymbol{d}$ denote the camera origin and unit direction vector of the ray. With scalar density $\sigma(\boldsymbol{x}_i)$ predicted by an occupancy network, opacity $\alpha_i = 1 - e^{-\sigma(\boldsymbol{x}_i)|t_{i+1}-t_i|}$ is computed for the sequential volume-rendering process as follows:

$$\hat{\boldsymbol{c}} = \sum_{i=1}^{N} \alpha_i T_i \boldsymbol{c}_i, \quad T_i = \prod_{j=1}^{i-1} (1 - \alpha_j), \tag{1}$$

where $\hat{\boldsymbol{c}}$ denotes the rendered RGB color of the pixel, $\boldsymbol{c}_i$ denotes the color at point $\boldsymbol{x}_i$ sampled from other views, and T_i represents the accumulated transmittance up to the i-th point. A photometric loss (Zhao et al., 2016) between $\hat{\boldsymbol{c}}$ and the observed RGB color $\boldsymbol{c}_{\text{gt}}$ supervises the predicted density field. Based on this scene-level rendering formulation, existing semantic approaches (Huang et al., 2024a; Jevtić et al., 2025) further predict semantic logits $\boldsymbol{s}(\boldsymbol{x}_i) \in \mathbb{R}^Q$ and derive semantic occupancy probabilities from $p_{\text{occ}}(\boldsymbol{x}_i) = \alpha_i$:

$$p_0(\boldsymbol{x}_i) = 1 - p_{\text{occ}}(\boldsymbol{x}_i), \quad p_q(\boldsymbol{x}_i) = \pi_q(\boldsymbol{x}_i) p_{\text{occ}}(\boldsymbol{x}_i), \quad \boldsymbol{\pi}(\boldsymbol{x}_i) = \text{softmax}(\boldsymbol{s}(\boldsymbol{x}_i)). \tag{2}$$

Here, the semantic prediction only partitions the already predicted occupied probability among different categories, while whether the point is occupied remains determined by the scene-level probability $p_{\text{occ}}(\boldsymbol{x}_i)$. Therefore, the occupancy is still learned as a unified scene representation.

In our proposed framework, we derive the probability vector $\boldsymbol{p}(\boldsymbol{x}_i)$ directly from the occupancy compatibility scores $\boldsymbol{\ell}(\boldsymbol{x}_i) \in \mathbb{R}^Q$ predicted by the network, decomposing occupancy learning into simpler instance-level predictions. Since the occupancy states are mutually exclusive and exhaustive, the resulting probabilities form the following categorical distribution over slots:

$$p_0(\boldsymbol{x}_i) + \sum_{q=1}^{Q} p_q(\boldsymbol{x}_i) = 1, \quad p_{\text{occ}}(\boldsymbol{x}_i) = 1 - p_0(\boldsymbol{x}_i) = \sum_{q=1}^{Q} p_q(\boldsymbol{x}_i). \tag{3}$$

To parameterize this distribution, we concatenate a fixed zero compatibility score for empty space with $\boldsymbol{\ell}(\boldsymbol{x}_i)$ and apply softmax to obtain $\boldsymbol{p}(\boldsymbol{x}_i)$. Under this parameterization, each compatibility

score indicates whether its corresponding slot is more or less probable than empty space at a given 3D point. Unlike previous formulations that directly learn a unified scene-level occupancy, our formulation decomposes this challenging task into learning the occupancy of individual slots, which are then composed into the scene-level occupancy $p_{\text{occ}}(\boldsymbol{x}_i)$. In particular, instance-related slots exhibit stronger 3D shape consistency across objects of the same category, providing useful geometric priors that simplify their occupancy learning in a decoupled manner. Fixing the empty-space score to zero only sets a reference and does not restrict the represented categorical distribution. A detailed analysis of this design is provided in Supplement D.1.

However, the conversion from scores to occupancy probability must also account for the sampling intervals $\delta_i = |t_{i+1} - t_i|$ along each ray (Ahn et al., 2024). Accordingly, the components of the probability vector are computed as follows (see Supplement D.1 for a detailed derivation):

$$p_q(\boldsymbol{x}_i) = \frac{\exp(\ell_q(\boldsymbol{x}_i) + \log \delta_i)}{1 + \sum_{k=1}^{Q} \exp(\ell_k(\boldsymbol{x}_i) + \log \delta_i)}, \quad p_0(\boldsymbol{x}_i) = \frac{1}{1 + \sum_{k=1}^{Q} \exp(\ell_k(\boldsymbol{x}_i) + \log \delta_i)}. \tag{4}$$

In the volume rendering process, the opacity α_i is equal to $p_{\text{occ}}(\boldsymbol{x}_i) = 1 - p_0(\boldsymbol{x}_i)$. We first accumulate the slot probabilities along each ray to obtain $\tilde{\boldsymbol{s}} = \sum_{i=1}^{N} T_i \boldsymbol{p}(\boldsymbol{x}_i)$. We then normalize $\tilde{\boldsymbol{s}}$ across the slot dimension at each pixel to form the rendered segmentation probability map $\widetilde{\boldsymbol{S}}$, which is used for the supervision described in Section 3.4.

3.3 Instance Decoupling Network

Generating the probabilities $\boldsymbol{p}(\boldsymbol{x}_i)$ introduced in the above subsection requires the occupancy compatibility scores $\boldsymbol{\ell}(\boldsymbol{x}_i)$ at each queried 3D location. However, the image encoder produces scene-level visual features in which different instances remain entangled. To produce such scores, we develop an Instance Decoupling Network that constructs point features from the input-view image features through a projection-guided feature modulation (PFM) process, and decouples occupancy slot geometry through a Transformer decoder with learnable queries, as illustrated in Fig. 2.

Projection-Guided Feature Modulation. The visual features extracted by the encoder are ambiguous for queried 3D points, as points at different distances along the same ray may share identical image features. To alleviate this ambiguity, we introduce projection-guided feature modulation to refine sampled features using point-specific projection information.

For each queried 3D point $\boldsymbol{x}_i = (x_i, y_i, z_i)^\top$, we first sample the point-wise feature $\boldsymbol{f}_i$ from the feature map $\boldsymbol{F} \in \mathbb{R}^{C \times H \times W}$ at the corresponding projection pixel $(u_i, v_i, 1)^\top = (1/z_i)\boldsymbol{K}_0 \boldsymbol{x}_i$. Since the sampled image feature only provides pixel-level information, we further introduce the UVZ coordinates $\boldsymbol{r}_i = (u_i, v_i, z_i)^\top$, which describe where and at what depth the queried point is observed in the input picture. The UVZ coordinates are encoded with a fixed positional encoding function $E(\cdot)$ to generate a point-specific factor for feature modulation:

$$\boldsymbol{g}_i = \tanh\left(W_g E(\boldsymbol{r}_i)\right), \qquad \boldsymbol{f}_i^m = \boldsymbol{f}_i \odot (1 + \gamma \boldsymbol{g}_i), \tag{5}$$

where W_g is a learnable linear transformation and γ is a predefined modulation scale. This process enhances the sampled image feature according to the specific image location and depth of each 3D query point, providing more discriminative visual evidence for subsequent instance decoupling.

While $\boldsymbol{f}_i^m$ provides point-specific visual evidence, learning instance-level 3D shape priors also requires explicit spatial information about the queried point. We therefore incorporate the positional encoding of each 3D query point in the camera coordinate system: $\boldsymbol{f}_i^p = [\boldsymbol{f}_i^m; E(\boldsymbol{x}_i)]$.

Query-Based Transformer Decoder. The proposed Instance-Centric Categorical Occupancy Formulation requires the network to predict occupancy compatibility scores $\boldsymbol{\ell}(\boldsymbol{x}_i) \in \mathbb{R}^Q$. Therefore, we represent each potential instance with a learnable query and employ a query-based Transformer decoder to transform the modulated point features $\boldsymbol{f}_i^p$ into occupancy compatibility scores $\boldsymbol{\ell}(\boldsymbol{x}_i)$.

Specifically, following the query-based decoding design in SparseOcc (Liu et al., 2024), we first initialize Q learnable queries $\boldsymbol{Z}^{(0)} = [\boldsymbol{z}_1^{(0)}, \ldots, \boldsymbol{z}_Q^{(0)}]$. At each layer, we update the queries by interacting them with point features $\boldsymbol{F}^p = [\boldsymbol{f}_1^p, \ldots, \boldsymbol{f}_N^p]$ through the Transformer decoder as follows:

$$\boldsymbol{Z}^{(l)} = \mathcal{D}^{(l)}\left(\boldsymbol{Z}^{(l-1)}, \boldsymbol{F}^p\right), \quad l = 1, \ldots, L, \tag{6}$$

where $\mathcal{D}^{(l)}$ denotes the l-th decoder layer. After aggregating information from the point features, these different queries progressively capture 3D shape priors corresponding to different slots. After the final decoder layer, the updated queries are compared with point features to measure their correspondence. We interpret the dot product between the q-th instance query $\boldsymbol{z}_q^{(L)}$ and the point feature $\boldsymbol{f}_i^p$ as the corresponding compatibility score:

$$\ell_q(\boldsymbol{x}_i) = \frac{1}{\sqrt{C}} \left\langle W_z \boldsymbol{z}_q^{(L)}, W_f \boldsymbol{f}_i^p \right\rangle, \tag{7}$$

where W_z and W_f are learnable linear transformations and C denotes the feature dimension.

3.4 Instance-Anchored Learning Strategy

Although the preceding formulation and network enable decoupled occupancy prediction, learning without 3D occupancy annotations remains challenging. Our proposed Instance-Anchored Learning Strategy provides sufficient supervision during training. We first construct instance-anchored view groups based on co-visible instances, then introduce segmentation and geometric constraints on side views in the constructed view groups, and finally integrate these constraints with the input-view reconstruction losses into an overall training objective for decoupled occupancy learning.

Instance-Anchored View Group. Unsupervised occupancy learning relies on multi-view consistency to constrain 3D predictions. To extend this consistency to instance occupancy, the selected side views should observe at least one object instance also visible in the input view, such that the same object occupancy can be constrained from different viewpoints. However, in Wimbauer et al. (2023) and following works, side views are selected using fixed or randomly sampled camera offsets, failing to guarantee the presence of such co-visible instances. We therefore design a dynamic side-view retrieval procedure (see Supplement D.3) that requires each selected side view $\boldsymbol{I}_k^S$ to satisfy $\mathcal{O}(\boldsymbol{I}_0) \cap \mathcal{O}(\boldsymbol{I}_k^S) \neq \varnothing$, where $\mathcal{O}(\boldsymbol{I})$ denotes the set of instances visible in view $\boldsymbol{I}$, precomputed from the dataset-provided instance boxes and camera calibration. Once selected, the side views, along with the front views, form an instance-anchored view group. They observe instances relevant to the predicted occupancy, enabling effective segmentation and geometric supervision.

Side-View Supervision. For each selected side view $\boldsymbol{I}_k^S$, we obtain a predicted depth map $\widehat{\boldsymbol{D}}_k^S$ from an auxiliary unsupervised depth estimator (Godard et al., 2019) and a pseudo segmentation map $\widehat{\boldsymbol{S}}_k^S$ from UnSAM+ (Wang et al., 2024b). These predictions provide supervision for query points before and near the observed object surfaces: points before the surfaces are constrained to have zero scene occupancy, while points near the surfaces are encouraged to match the slots according to the pseudo segmentation labels sampled from $\widehat{\boldsymbol{S}}_k^S$.

Specifically, given a query point $\boldsymbol{x}_i$, we project it onto $\boldsymbol{I}_k^S$ at pixel (u_i^k, v_i^k), and denote its depth in the side-view camera coordinate system by d_i^k. The predicted surface depth $\hat{d}_i^k$ and pseudo instance label $\hat{y}_i^k$ are then sampled from $\widehat{\boldsymbol{D}}_k^S$ and $\widehat{\boldsymbol{S}}_k^S$ at the projected pixel (u_i^k, v_i^k). We define the free-space point set as $\mathcal{P}_{\text{free}} = \left\{ \boldsymbol{x}_i \mid \exists k, d_i^k < \hat{d}_i^k \right\}$, and define the surface point set for view k as $\mathcal{P}_{\text{surf}}^k = \left\{ \boldsymbol{x}_i \mid \left| d_i^k - \hat{d}_i^k \right| < \tau_{\text{surf}} \right\}$, where τ_{surf} is a predefined surface threshold. Since the pseudo instance labels and predicted slots are both unordered, we establish their assignment through Hungarian matching. We formulate the constraints on the surface and free-space point sets as follows:

$$\mathcal{L}_{\text{surf}} = -\frac{1}{\sum_k \left|\mathcal{P}_{\text{surf}}^k\right|} \sum_k \sum_{\boldsymbol{x}_i \in \mathcal{P}_{\text{surf}}^k} \log \pi_{\rho^k(\hat{y}_i^k)}(\boldsymbol{x}_i), \qquad \mathcal{L}_{\text{free}} = \frac{1}{|\mathcal{P}_{\text{free}}|} \sum_{\boldsymbol{x}_i \in \mathcal{P}_{\text{free}}} p_{\text{occ}}(\boldsymbol{x}_i), \tag{8}$$

where ρ^k denotes the resulting assignment from pseudo labels $\hat{y}_i^k$ to instance slots, and $\pi_q(\boldsymbol{x}_i) = p_q(\boldsymbol{x}_i)/p_{\text{occ}}(\boldsymbol{x}_i)$ is a conditional probability given that $\boldsymbol{x}_i$ is occupied. Together, the surface instance constraint supervises the slot assignment of occupied surface points, while the free-space geometric constraint suppresses false occupancy in front of all observed surfaces.

Overall Training Objective. Following the rendering process introduced in Sec. 3.2, we render a reconstructed image and an instance probability map $\widetilde{\boldsymbol{S}}$ from the predicted occupancy probabilities

$\boldsymbol{p}(\boldsymbol{x}_i)$. The reconstructed image is constrained by the photometric loss $\mathcal{L}_{\text{photo}}$ (Zhao et al., 2016). In addition, $\widetilde{\boldsymbol{S}}$ is supervised by the input-view pseudo label $\widehat{\boldsymbol{S}}$ with the Hungarian matching strategy:

$$\mathcal{L}_{\text{ins}} = -\frac{1}{|\Omega|} \sum_{\boldsymbol{u} \in \Omega} \log \widetilde{S}_{\rho^0(\widehat{\boldsymbol{s}}(\boldsymbol{u}))}(\boldsymbol{u}), \tag{9}$$

where Ω denotes the set of valid input-view pixels. Finally, the overall training objective combines the input-view reconstruction losses with the side-view surface and free-space constraints:

$$\mathcal{L} = \lambda_{\text{photo}}\mathcal{L}_{\text{photo}} + \lambda_{\text{ins}}\mathcal{L}_{\text{ins}} + \lambda_{\text{surf}}\mathcal{L}_{\text{surf}} + \lambda_{\text{free}}\mathcal{L}_{\text{free}}, \tag{10}$$

where each λ weights its corresponding loss term.

4 Experiments

4.1 Experimental Setup

Datasets. Experiments are conducted on KITTI-360 (Liao et al., 2022) and CarlaOcc (Feng et al., 2026). We follow the dataset split in Wimbauer et al. (2023) for KITTI-360. For CarlaOcc, we use the 13 sequences from Town10HD and manually split them into training and test sets. The training set contains 8,860 keyframes from sequences 00–02, 05–10, and 12, with both front cameras. The test set contains 500 keyframes from the left front camera in sequences 03, 04, and 11. This split evaluates generalization to unseen driving sequences while covering a range of traffic conditions, reducing the risk of temporal leakage between training and testing. 3D occupancy annotations from Li et al. (2024b) and Feng et al. (2026) are used only for evaluation (Guo et al., 2026), while dataset-provided instance 3D bounding boxes are used only for offline side-view selection.

Network Details. ICOcc uses a pretrained DINOv2 ViT-B/14 encoder with frozen weights. Multi-level encoder features are then fused using trainable convolutional adapters and a skip-connected decoder adapted from Godard et al. (2019), producing a 64-channel image feature map. Bilinear interpolation is used for image feature sampling. The query-based Transformer decoder consists of two Transformer layers with eight attention heads and uses 32 learnable queries.

Training Configurations. ICOcc is trained for 25 epochs using Adam with a batch size of 12 on a single NVIDIA RTX 4090 GPU. The initial learning rate is set to 1×10^{-4} and decayed by a factor of 0.1 every 120,000 optimization steps. Mixed-precision training is used throughout. For both datasets, all input images are resized to 192×640 pixels. During training, each sample consists of 16 views from an instance-anchored view group, comprising two consecutive frames from each front camera and side views at three timestamps from each fisheye camera. Pseudo labels are precomputed before occupancy training. Pseudo segmentation labels are generated using UnSAM+ (Wang et al., 2024b) to supervise query-to-segment correspondence without assigning fixed semantic identities to the learnable queries. For pseudo depth labels, we follow the network architecture and training scheme of Godard et al. (2019) and train the depth model on side-view RGB images using their camera poses. The trained model is then used to infer pseudo depth for side-view occupancy supervision.

4.2 Comparison with State-of-the-art Methods

Geometric Occupancy Prediction. We reproduce representative unsupervised occupancy prediction methods under unified configurations. We first evaluate occupancy geometry prediction, *i.e.*, classifying each voxel as occupied or empty, using O_{Acc}, O_{Prec}, and IE_{Rec} in Table 1. ICOcc achieves the best overall performance on both datasets, with particularly strong gains in O_{Prec} and IE_{Rec}. Compared with the strongest baseline for each metric, ICOcc improves these two metrics by 3.1% and 10.2% on KITTI-360, and by 14.4% and 27.3% on CarlaOcc, respectively. These gains indicate that ICOcc substantially reduces trailing artifacts, which contributes to the overall improvement in occupancy geometry prediction.

Occupancy Prediction for Object Categories. We evaluate class-wise occupancy prediction by matching learned slots to semantic categories and computing RayIoU. Following the instance-class

Table 1: Quantitative comparison of occupancy prediction on KITTI-360 and CarlaOcc.

Method	KITTI-360			CarlaOcc		
	O_{Acc} ↑	O_{Prec} ↑	IE_{Rec} ↑	O_{Acc} ↑	O_{Prec} ↑	IE_{Rec} ↑
BTS (Wimbauer et al., 2023)	0.799	0.615	0.510	0.900	0.382	0.416
KDBTS (Han et al., 2024)	0.793	0.595	0.452	0.885	0.340	0.348
KYN (Li et al., 2024a)	**0.813**	0.652	0.550	0.896	0.357	0.388
ViPOcc (Feng et al., 2025)	0.747	0.523	0.367	0.890	0.338	0.396
SceneDINO (Jevtić et al., 2025)	0.793	0.622	0.465	0.904	0.374	0.450
Rebench (Guo et al., 2026)	0.798	0.591	0.408	0.903	0.374	0.429
Ours	**0.813**	**0.672**	**0.606**	**0.917**	**0.437**	**0.573**

Table 2: Class-wise occupancy prediction results for selected object categories on KITTI-360.

Method	Class-Wise RayIoU (%) ↑					Avg. RayIoU (%) ↑
	Car	Motor.	Truck	Other-Veh.	Building	
SceneDINO	14.9	0.0	9.0	0.8	11.1	7.2
Ours	**24.3**	**3.2**	**13.1**	**6.9**	**12.8**	**12.1**

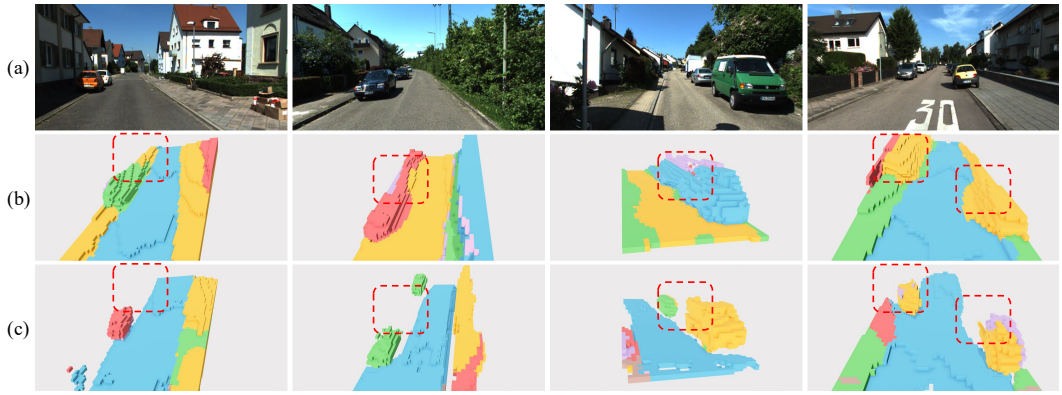


Figure 3: Qualitative comparisons of unsupervised semantic occupancy prediction in ROI: (a) input RGB images; (b) SceneDINO results; (c) our results. The colors are randomly assigned, as the learned slots in both methods do not have fixed semantic correspondence.

definitions in KITTI-360 and the corresponding mapping in SSCBench-KITTI-360, we report results for selected representative categories in Table 2. ICOcc improves the average RayIoU across these categories by 68.1% over SceneDINO. The improvements are particularly significant for vehicle categories. ICOcc improves RayIoU of cars, motorcycles, trucks, and other vehicles from 14.9%, 0.0%, 9.0%, and 0.8% to 24.3%, 3.2%, 13.1%, and 6.9%, respectively. These results suggest that ICOcc captures class-specific 3D shape priors and produces more reliable vehicle geometry.

Qualitative Analysis. We visualize the predicted occupancy results within the Region of Interest (ROI) for qualitative comparison. Since the learned occupancy slots do not have fixed semantic identities, the colors in different visualizations are randomly assigned and do not indicate matching semantic categories. As shown in Fig. 3, ICOcc produces more compact object geometries with substantially fewer trailing artifacts compared with SceneDINO. In particular, the object boundaries on the invisible side of instances are better preserved, indicating that ICOcc can better recover complete 3D object shapes from a single view. These improvements can be attributed to the different instance modeling paradigms adopted by the two methods. SceneDINO first reconstructs scene-level occupancy and then obtains instance separation through clustering. Therefore, its instance prediction quality is highly dependent on the clustering process, which lacks explicit 3D shape priors and struggles to separate geometrically similar objects. As a result, neighboring instances with connected occupancy patterns can be merged into a single instance. In contrast, ICOcc directly models

Table 3: Ablation study of the main designs in ICOcc on KITTI-360.

Configuration	O_{Acc} ↑	O_{Prec} ↑	IE_{Rec} ↑
Full ICOcc	**0.813**	**0.672**	**0.606**
Instance-Anchored Learning			
w/o Side-View Retrieval	0.808	0.657	0.567
w/o $\mathcal{L}_{\text{free}}$	0.803	0.627	0.486
w/o $\mathcal{L}_{\text{ins}}$ & $\mathcal{L}_{\text{surf}}$	0.796	0.612	0.452
Instance Decoupling Network			
w/o Projection-Guided Feature Modulation	0.803	0.620	0.414
Query Transformer → MLP from $\boldsymbol{f}_i^p$ to $\boldsymbol{\ell}(\boldsymbol{x}_i)$	0.689	0.448	0.176
Baseline	0.797	0.605	0.452

individual object geometries through instance-centric occupancy prediction, allowing each instance to be reconstructed with its own shape prior and reducing instance merging.

We further observe that ICOcc exhibits a more conservative prediction behavior in regions behind foreground instances, where no direct instance evidence is available from the input views. Compared with SceneDINO, it avoids extending occupied regions into these areas and tends to preserve empty space instead of hallucinating unsupported random geometry. This behavior is consistent with the visibility-aware annotation strategy adopted by occupancy benchmarks, where voxels without reliable observations, such as those hidden by occlusion or outside sensor coverage, are marked as invalid rather than assigned uncertain occupancy labels (Tian et al., 2023). Therefore, ICOcc naturally avoids making unreliable predictions in regions that are difficult to verify.

4.3 Ablation Studies

As demonstrated in Table 3, all ablations are conducted by modifying only the specified component while keeping other parts unchanged, except for the baseline, which restores the conventional scene-level pipeline while using the same image encoder for a fair comparison. The full ICOcc achieves the best performance, validating the effectiveness of the proposed framework. Removing side-view retrieval slightly reduces performance, but eliminates the need for 3D bounding-box annotations during view selection. Even without this annotation-dependent step, ICOcc remains competitive with prior methods and achieves the highest $\mathrm{O}_{\mathrm{Prec}}$ and $\mathrm{IE}_{\mathrm{Rec}}$ on KITTI-360. Removing the free-space constraint degrades occupancy prediction, indicating that explicit empty-space supervision helps regulate occupancy estimation and suppress false-positive predictions. Removing $\mathcal{L}_{\mathrm{ins}}$ and $\mathcal{L}_{\mathrm{surf}}$ causes a more significant performance drop, and the visualization reveals a slot collapse phenomenon where one slot dominates the whole scene. This suggests that these constraints provide necessary guidance for preserving slot diversity and enforcing instance-level decomposition. Replacing PFM or the query Transformer with simpler alternatives leads to performance degradation, demonstrating that the proposed design provides stronger capacity for capturing 3D shape priors than direct point-wise prediction.

5 Conclusion and Future Work

In this work, we revisited unsupervised monocular 3D occupancy prediction from an instance-centric perspective. Rather than directly learning a unified scene geometry, ICOcc decomposes occupancy learning into decoupled predictions, allowing instance-related geometry to benefit from the shared 3D shape priors of same-category objects. To realize this perspective, ICOcc integrates an instance-centric categorical occupancy formulation, an instance decoupling network, and an instance-anchored learning strategy. Experiments on the KITTI-360 and CarlaOcc datasets demonstrate consistent improvements in occupancy prediction, with clearer object boundaries and substantially reduced trailing artifacts. Future work will explore scaling this instance-centric framework to unsupervised 3D occupancy pretraining on larger image collections and evaluate the transfer of the resulting representations across datasets and downstream 3D perception tasks.

REFERENCES

Joshua Ahn, Haochen Wang, Raymond A Yeh, and Greg Shakhnarovich. Alpha invariance: On inverse scaling between distance and volume density in neural radiance fields. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 20396–20405. IEEE, 2024.

Anh-Quan Cao and Tuan-Hung Vu. Occany: Generalized unconstrained urban 3d occupancy. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 28599–28609, June 2026.

Anh-Quan Cao, Angela Dai, and Raoul De Charette. Pasco: Urban 3d panoptic scene completion with uncertainty awareness. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 14554–14564. IEEE, 2024.

Alexey Dosovitskiy, German Ros, Felipe Codevilla, Antonio Lopez, and Vladlen Koltun. Carla: An open urban driving simulator. In *Conference on Robot Learning*, pp. 1–16. PMLR, 2017.

Jiayuan Du, Yiming Zhao, Zhenglong Guo, Yong Pan, Wenbo Hou, Zhihui Hao, Kun Zhan, and Qijun Chen. Sparseworld-tc: Trajectory-conditioned sparse occupancy world model. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 7425–7434, 2026.

Yi Feng, Yu Han, Xijing Zhang, Tanghui Li, Yanting Zhang, and Rui Fan. ViPOcc: Leveraging visual priors from vision foundation models for single-view 3d occupancy prediction. In *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, pp. 3004–3012, 2025.

Yi Feng, Zizhan Guo, Yu Ma, Hanli Wang, Rui Fan, et al. An instance-centric panoptic occupancy prediction benchmark for autonomous driving. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 14219–14228, June 2026.

Clément Godard, Oisin Mac Aodha, Michael Firman, and Gabriel J Brostow. Digging into self-supervised monocular depth estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3828–3838, 2019.

Markus Gross, Aya Fahmy, Danit Niwattananan, Dominik Muhle, Rui Song, Daniel Cremers, and Henri Meeß. IPFormer: Visual 3d panoptic scene completion with context-adaptive instance proposals. *Advances in Neural Information Processing Systems (NeurIPS)*, 38:4989–5014, 2026.

Zizhan Guo, Yi Feng, Mengtan Zhang, Haoran Zhang, Wei Ye, and Rui Fan. Rebenchmarking unsupervised monocular 3d occupancy prediction. *arXiv preprint arXiv:2602.06488*, 2026.

Keonhee Han, Dominik Muhle, Felix Wimbauer, and Daniel Cremers. Boosting self-supervision for single-view scene completion via knowledge distillation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 9837–9847, June 2024.

Yuanhui Huang, Wenzhao Zheng, Borui Zhang, Jie Zhou, and Jiwen Lu. Selfocc: Self-supervised vision-based 3d occupancy prediction. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (CVPR)*, pp. 19946–19956, 2024a.

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

Aleksandar Jevtić, Christoph Reich, Felix Wimbauer, Oliver Hahn, Christian Rupprecht, Stefan Roth, and Daniel Cremers. Feed-forward SceneDINO for unsupervised semantic scene completion. In *IEEE/CVF International Conference on Computer Vision (ICCV)*, 2025. in press.

Xiaosong Jia, Chenhe Zhang, Yule Jiang, Songbur Wong, Zhiyuan Zhang, Chen Chen, Shaofeng Zhang, Xuanhe Zhou, Xue Yang, Junchi Yan, et al. Spatial retrieval augmented autonomous driving. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 17787–17797, 2026.

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 *Proceedings of the 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 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 11960–11970, 2025.

Rui Li, Tobias Fischer, Mattia Segu, Marc Pollefeys, Luc Van Gool, and Federico Tombari. Know your neighbors: Improving single-view reconstruction via spatial vision-language reasoning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 9848–9858, 2024a.

Shijie Li, Zhongyao Cheng, Rong Li, Shuai Li, Juergen Gall, Xun Xu, and Xulei Yang. Global-aware monocular semantic scene completion with state space models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 25550–25559. IEEE, 2025.

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)*, pp. 13333–13340, 2024b.

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.

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 *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 54–71. Springer, 2024.

Lin Liu, Caiyan Jia, Guanyi Yu, Ziying Song, JunQiao Li, Feiyang Jia, Peiliang Wu, Xiaoshuai Hao, and Yadan Luo. Guideflow: Constraint-guided flow matching for planning in end-to-end autonomous driving. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3719–3728, June 2026.

Yili Liu, Linzhan Mou, Xuan Yu, Chenrui Han, Sitong Mao, Rong Xiong, and Yue Wang. Let occ flow: Self-supervised 3d occupancy flow prediction. In *Conference on Robot Learning*, pp. 2895–2912. PMLR, 2025.

Ben Mildenhall, Pratul P Srinivasan, Matthew Tancik, Jonathan T Barron, Ravi Ramamoorthi, and Ren Ng. NeRF: Representing scenes as neural radiance fields for view synthesis. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 405–421, 2020.

Matthew Strong, Wei-Jer Chang, Quentin Herau, Jiezhi Yang, Yihan Hu, Chensheng Peng, and Wei Zhan. Learning to drive is a free gift: Large-scale label-free autonomy pretraining from unposed in-the-wild videos. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 32144–32153, June 2026.

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

Pin Tang, Zhongdao Wang, Guoqing Wang, Xiangxuan Ren, and Chao Ma. Sparseocc++: Geometry-aware sparse latent representation for semantic occupancy prediction. *arXiv preprint arXiv:2607.04732*, 2026.

Xiaoyu Tian, Tao Jiang, Longfei Yun, Yucheng Mao, Huitong Yang, Yue Wang, Yilun Wang, and Hang Zhao. Occ3D: A large-scale 3d occupancy prediction benchmark for autonomous driving. *Advances in Neural Information Processing Systems (NeurIPS)*, pp. 64318–64330, 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. *Advances in Neural Information Processing Systems (NeurIPS)*, 37:119861–119885, 2024a.

Weihua Wang, Yubo Cui, Xiangru Lin, Zhiheng Li, and Zheng Fang. Towards 3d object-centric feature learning for semantic scene completion. In *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, volume 40, pp. 10136–10144, 2026.

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

XuDong Wang, Jingfeng Yang, and Trevor Darrell. Segment anything without supervision. *Advances in Neural Information Processing Systems (NeurIPS)*, 37:138731–138755, 2024b.

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. *Computing Reasearch Repository (CoRR)*, pp. 1–14, 2025.

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

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

Felix Wimbauer, Nan Yang, Christian Rupprecht, and Daniel Cremers. Behind the scenes: Density fields for single view reconstruction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 9076–9086, 2023.

Yu Xue, Longjun Gao, Yuanqi Su, HaoAng Lu, and Xiaoning Zhang. Sparsity-aware voxel attention and foreground modulation for 3d semantic scene completion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 5751–5761, June 2026.

Fengyi Zhang, Xiangyu Sun, Huitong Yang, Zheng Zhang, Zi Huang, and Yadan Luo. Test-time 3d occupancy prediction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 35691–35701, June 2026.

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 (ICCV)*, pp. 9433–9443, October 2023.

Hang Zhao, Orazio Gallo, Iuri Frosio, and Jan Kautz. Loss functions for image restoration with neural networks. *IEEE Transactions on Computational Imaging*, 3(1):47–57, 2016.

Linqing Zhao, Xiuwei Xu, Ziwei Wang, Yunpeng Zhang, Borui Zhang, Wenzhao Zheng, Dalong Du, Jie Zhou, and Jiwen Lu. Lowrankocc: Tensor decomposition and low-rank recovery for vision-based 3d semantic occupancy prediction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 9806–9815. IEEE, 2024.

Chaoda Zheng, Feng Wang, Naiyan Wang, Shuguang Cui, and Zhen Li. Towards flexible 3d perception: Object-centric occupancy completion augments 3d object detection. *Advances in Neural Information Processing Systems (NeurIPS)*, 37:116060–116083, 2024.

Changqing Zhou, Yueru Luo, and Changhao Chen. Generalizing visual geometry priors to sparse gaussian occupancy prediction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 28578–28587, June 2026.

A AI USE STATEMENT

In this work, we used generative AI tools only to edit and polish the manuscript to improve language, clarity, and readability. We did not use generative AI tools to develop research ideas or hypotheses, design the methodology or experiments, implement the proposed method, generate or process data, analyze or interpret experimental results, formulate mathematical claims, or produce scientific conclusions. We carefully reviewed all AI-assisted edits to ensure that they preserved the intended technical meaning and did not introduce new claims or factual content. We take full responsibility for the final content of this work, including text produced with the aid of generative AI.

B ETHICS

In this study, we utilize the KITTI-360 (Liao et al., 2022), SSCBench-KITTI-360 (Li et al., 2024b), and CarlaOcc (Feng et al., 2026) datasets for the development and evaluation of 3D occupancy prediction networks. It is important to emphasize that we strictly adhere to the terms of usage for each dataset. We acknowledge that these datasets may contain images with visible faces and other personal data collected without consent. Nevertheless, we confirm that no biometric information has been processed. All images are used in accordance with the CC-BY license or in a manner compatible with the Data Analysis Permission.

C REPRODUCIBILITY STATEMENT

Our results are reproducible by consulting section 4 and corresponding sections in appendix for the experimental details of our work, and by downloading the supplementary materials which contain the complete source code. The network's weights to reproduce the experimental results claimed in section 4 will be made available upon publication.

D ADDITIONAL METHODOLOGICAL DETAILS

D.1 INSTANCE-CENTRIC CATEGORICAL OCCUPANCY FORMULATION DETAILS

Why Not Predict an Empty-Space Score? A straightforward alternative is to let the network predict $(Q+1)$-dimensional scores, including an additional score for empty space, and apply softmax to obtain the categorical occupancy probabilities. However, empty space is not an object instance but the complement of scene occupancy. Predicting it with an independent score would therefore retain an explicit scene-level occupancy prediction target, rather than deriving occupancy entirely from the predicted instance geometries. This weakens the instance-centric formulation, since the network may determine whether a point is occupied through the empty-space score without fully exploiting the 3D shape priors shared across object instances.

Why Fixing the Empty-space Score to Zero does not Restrict the Representation? Because softmax is invariant to a common shift of all scores, any set of $Q+1$ scores with an empty-space score e is equivalent to subtracting e from all scores, yielding a zero empty-space score without changing the resulting probabilities. Therefore, fixing the empty-space score to zero only sets a reference and does not restrict the categorical distribution that can be represented.

$$\mathrm{softmax}([e, \boldsymbol{\ell}]) = \mathrm{softmax}([0, \boldsymbol{\ell} - e])\,. \tag{11}$$

Instance Occupancy Probabilities Derivation. As discussed in Sec. 3.2, when the occupancy compatibility scores are directly applied with softmax, the categorical distribution over slots is computed as follows:

$$p_q(\boldsymbol{x}_i) = \frac{\exp(\ell_q(\boldsymbol{x}_i))}{1 + \sum_{k=1}^{Q} \exp(\ell_k(\boldsymbol{x}_i))}, \quad p_0(\boldsymbol{x}_i) = \frac{1}{1 + \sum_{k=1}^{Q} \exp(\ell_k(\boldsymbol{x}_i))}. \tag{12}$$

However, the magnitude of the compatibility scores is ambiguous due to the scale ambiguity in volume rendering (Ahn et al., 2024). In volume rendering, the contribution of a sampled point is

determined by its opacity $\alpha_i = 1 - \exp(-\sigma_i \delta_i)$, which depends on both the density σ_i and the spatial interval $\delta_i = |t_{i+1} - t_i|$. Therefore, the same density value can produce different occupancy contributions under different sampling intervals (Guo et al., 2026).

In previous methods (Wimbauer et al., 2023; Feng et al., 2025; Jevtić et al., 2025) and our pipeline, non-uniform point sampling is performed along each ray to obtain 3D query points during training. Consequently, different sampling densities induce different spatial neighborhoods δ_i around sampled points, leading to inconsistent occupancy contributions. To compensate for this inconsistency, the compatibility score should account for the spatial interval δ_i associated with each query point.

Specifically, the effective contribution of an occupied point is proportional to the product between its density and sampling interval:

$$\sigma_i \delta_i \propto \exp(\ell_i(\boldsymbol{x}_i))\delta_i = \exp(\ell_i(\boldsymbol{x}_i) + \log \delta_i). \tag{13}$$

Therefore, the sampling interval compensation can be introduced in the log domain by adding $\log \delta_i$ to the compatibility score. The resulting categorical occupancy distribution is formulated as:

$$p_q(\boldsymbol{x}_i) = \frac{\exp(\ell_q(\boldsymbol{x}_i) + \log \delta_i)}{1 + \sum_{k=1}^{Q} \exp(\ell_k(\boldsymbol{x}_i) + \log \delta_i)}, \quad p_0(\boldsymbol{x}_i) = \frac{1}{1 + \sum_{k=1}^{Q} \exp(\ell_k(\boldsymbol{x}_i) + \log \delta_i)}. \tag{14}$$

D.2 Instance Decoupling Network Details

Difference between UVZ and XYZ Positional Encodings. The UVZ and XYZ coordinates provide complementary positional information for each queried point. The UVZ coordinates $\boldsymbol{r}_i = (u_i, v_i, z_i)^\top$ describe where and at what depth the point is observed in the input view. They are therefore camera-dependent and are used to modulate the sampled image feature according to the point-specific observation. In contrast, the XYZ coordinates $\boldsymbol{x}_i = (x_i, y_i, z_i)^\top$ specify the point location directly in the 3D scene coordinate system. They are independent of the image projection and provide explicit spatial information for learning object geometries and shape priors. Accordingly, UVZ positional encoding is used for PFM, whereas XYZ positional encoding is concatenated with the modulated visual feature to construct the final point-wise feature.

Query-based Transformer Decoder Details. As discussed in Sec. 3.3, the query-based Transformer decoder adopts a multi-layer structure. The learnable instance queries are progressively updated by interacting with the point-wise features extracted by PFM:

$$\boldsymbol{Z}^{(l)} = \mathcal{D}^{(l)}\left(\boldsymbol{Z}^{(l-1)}, \boldsymbol{F}^p\right), \quad l = 1, \ldots, L, \tag{15}$$

where $\boldsymbol{Z}^{(l)} \in \mathbb{R}^{Q \times d}$ denotes the query features at the l-th decoder layer, $\boldsymbol{F}^p = [\boldsymbol{f}_1^p, \ldots, \boldsymbol{f}_N^p] \in \mathbb{R}^{N \times d}$ denotes the point-wise features of N sampled 3D query points, Q is the number of learnable queries, and d is the feature dimension. The initial query features $\boldsymbol{Z}^{(0)}$ are learnable parameters.

Each decoder layer consists of query self-attention, query-to-point cross-attention, and query-guided feature readout. We denote the self-attention and cross-attention operations as SA and CA, respectively. Both modules adopt pre-layer normalization and residual connections:

$$\mathrm{SA}(\boldsymbol{X}) = \boldsymbol{X} + \mathrm{Drop}(\boldsymbol{W}_{\mathrm{o}}\,\mathrm{MHA}(\mathrm{LN}(\boldsymbol{X}), \mathrm{LN}(\boldsymbol{X}), \mathrm{LN}(\boldsymbol{X}))), \tag{16}$$

$$\mathrm{CA}(\boldsymbol{X}, \boldsymbol{F}^p) = \boldsymbol{X} + \mathrm{Drop}(\boldsymbol{W}_{\mathrm{o}}\,\mathrm{MHA}(\boldsymbol{W}_{\mathrm{q}}\,\mathrm{LN}_{\mathrm{q}}(\boldsymbol{X}), \boldsymbol{W}_{\mathrm{k}}\,\mathrm{LN}_{\mathrm{F}}(\boldsymbol{F}^p), \boldsymbol{W}_{\mathrm{v}}\,\mathrm{LN}_{\mathrm{F}}(\boldsymbol{F}^p))), \tag{17}$$

where MHA denotes multi-head scaled dot-product attention, LN denotes layer normalization, Drop denotes dropout, and all projection matrices are learnable parameters.

The feed-forward network (FFN) adopts a pre-layer-normalized residual structure:

$$\mathrm{FFN}(\boldsymbol{X}) = \boldsymbol{X} + \mathrm{Drop}(\boldsymbol{W}_2\,\mathrm{Drop}(\mathrm{GELU}(\boldsymbol{W}_1\,\mathrm{LN}(\boldsymbol{X})))). \tag{18}$$

The FFN parameters and normalization layers are independently optimized for different decoder layers and sublayers.

After cross-attention, the query features are further refined through query-guided feature readout:

$$\boldsymbol{U}^{(l)} = \mathrm{FFN}\Big(\mathrm{CA}\Big(\mathrm{FFN}\Big(\mathrm{SA}(\boldsymbol{Z}^{(l-1)})\Big), \boldsymbol{F}^p\Big)\Big). \tag{19}$$

To aggregate point-wise features for each query, the decoder computes intermediate query-point compatibility logits:

$$\boldsymbol{M}^{(l)} = \frac{\left(\boldsymbol{U}^{(l)}\boldsymbol{W}_q^{(l)}\right)\left(\boldsymbol{F}^p\boldsymbol{W}_f^{(l)}\right)^\top}{\sqrt{d}}, \qquad \boldsymbol{M}^{(l)} \in \mathbb{R}^{Q\times N}, \tag{20}$$

where $\boldsymbol{W}_q^{(l)}$ and $\boldsymbol{W}_f^{(l)}$ are learnable linear projections. The sigmoid-transformed compatibility logits are normalized over points for each query:

$$A_{qn}^{(l)} = \frac{\sigma(M_{qn}^{(l)})}{\sum_{j=1}^{N}\sigma(M_{qj}^{(l)}) + \epsilon}, \quad q = 1,\dots,Q, \quad n = 1,\dots,N, \tag{21}$$

where $\sigma(\cdot)$ denotes the sigmoid function and ϵ is a small constant for numerical stability. The normalized weights $\boldsymbol{A}^{(l)} \in \mathbb{R}^{Q\times N}$ aggregate point-wise features and update the queries:

$$\boldsymbol{Z}^{(l)} = \boldsymbol{U}^{(l)} + \mathrm{LN}_\mathrm{r}\left(\boldsymbol{A}^{(l)}\boldsymbol{F}^p\right)\boldsymbol{W}_\mathrm{r}^{(l)}, \tag{22}$$

where LN_r denotes the normalization layer for feature readout and $\boldsymbol{W}_\mathrm{r}^{(l)}$ is a learnable weight.

After the final decoder layer, the updated queries are compared with point features using the same query-point compatibility formulation. Specifically, the compatibility score between the q-th query and the i-th point is computed as:

$$\ell_q(\boldsymbol{x}_i) = \frac{1}{\sqrt{C}}\left\langle W_z\boldsymbol{z}_q^{(L)}, W_f\boldsymbol{f}_i^p\right\rangle, \tag{23}$$

where W_z and W_f are learnable linear transformations and C denotes the feature dimension. The final compatibility scores $\ell_q(\boldsymbol{x}_i)$ are used as the query-wise occupancy logits.

D.3 Instance-Anchored Learning Strategy Details

Detailed Side-view Retrieval Procedure. To construct the instance-anchored view group, we first establish the correspondence between front-view frames and side-camera frames offline. For each training sequence, we collect candidate images from the front cameras and the side cameras. Given the annotated 3D bounding boxes, we determine their visibility in candidate views by projecting the 3D bounding boxes into each camera view using intrinsic and extrinsic parameters, as demonstrated in Fig. 4. Only frames within a temporal margin of the annotated instance frames are considered during visibility checking.

Specifically, we use a frame margin of 20 to restrict the search range. The projected instances are further filtered according to predefined depth ranges, where the valid ranges are set to 3–50 m for front cameras and 0–10 m for side cameras. After visibility filtering, only instance-view mappings with observable instances are retained. We additionally apply a semantic category whitelist to remove irrelevant instances. Instances with insufficient visible views are discarded. The obtained instance-view mappings are then used to construct retrieval maps from front-view frame to side-view frame. For each front-view frame, we first collect the instances visible in that frame and retrieve all side-camera frames where the same instances are observed. Duplicate side-camera frames shared by multiple instances are merged. The resulting mappings are stored separately for each sequence.

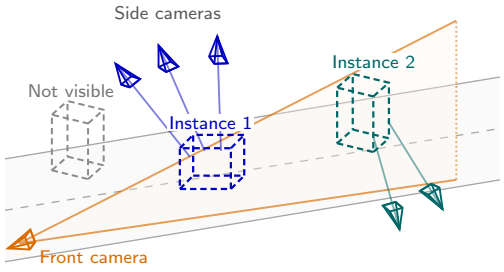


Figure 4: Illustration for instance-anchored side-view retrieval. For each front-camera frame, 3D bounding boxes are projected into candidate camera views to identify visible instances. Frames from side cameras observing the same instances are retrieved and merged into a front-to-side frame mapping. The gray instance lies outside the front camera's field of view and does not contribute to retrieval.

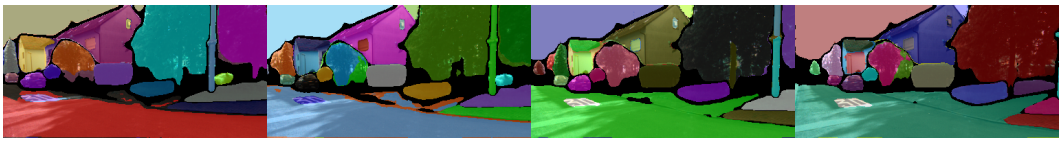


Figure 5: Visualization of pseudo segmentation labels from four consecutive frames. The same objects across frames are assigned different IDs, demonstrating that the pseudo segmentation labels do not provide cross-frame correspondence.

Given a training sample, the retrieved side-camera frames are further selected online. During training, only the camera ID and frame index of front-view frames are required for view retrieval. For each side camera, retrieved frames are sorted by temporal order and filtered by a minimum temporal gap of 7 pose indices from the reference front-view frame. The selected frames are sampled greedily from earlier to later timestamps, while enforcing the same minimum interval between selected frames. If fewer than the required number of side views are available, additional frames are sampled according to temporal spacing within the sequence range. At sequence boundaries, the indices are clamped to valid ranges.

With the training configuration discussed in Sec. 4.1, each retrieved anchor frame is expanded to two consecutive frames. Consequently, each training sample contains four front-view stereo images (two frames from the left camera and two frames from the right camera), and six images from each side camera (three retrieved anchors with two consecutive frames each). Therefore, each sample contains 16 RGB images in total, which together form an instance-anchored view group. Most views within the group are connected by shared visible instances, providing dense instance-level correspondences across viewpoints.

Pseudo Depth Label Details. We generate pseudo depth labels for side views by training a dedicated monocular depth estimation network based on Monodepth2 (Godard et al., 2019). Following the original self-supervised training scheme, the depth network is optimized with the photometric reconstruction loss between temporally adjacent views. Unlike the original Monodepth2 framework, which jointly trains a PoseNet to estimate relative camera motion, we remove the PoseNet branch and directly use the ground-truth camera poses provided by the dataset during view reconstruction. Therefore, the relative camera transformation used for photometric supervision has accurate metric scale rather than the scale ambiguity introduced by estimated poses. The depth network is trained specifically on side-camera adjacent images, and the predicted depth maps are used as pseudo depth supervision for side-view occupancy learning. Since the camera poses provide accurate scene scale during training, the resulting depth predictions can be directly interpreted as metric depth without additional scale alignment.

Pseudo Segmentation Label Details. We use UnSAM+ (Wang et al., 2024b) to generate pseudo segmentation labels for supervising query-to-segment correspondence. Unlike supervised semantic segmentation models that rely on manually defined semantic categories, UnSAM+ discovers segmentation regions without introducing category-specific annotations. Therefore, the generated labels provide category-agnostic spatial grouping cues rather than predefined semantic identities, as illustrated in Fig. 5. This property is important for our instance-centric formulation, as the pseudo segmentation labels are only used to provide correspondence and separation constraints between queries and image regions, instead of assigning semantic meanings to individual queries. The learned queries are encouraged to discover instance-level structures from visual consistency and focus on modeling instance geometry.

Specifically, the pseudo segmentation labels are generated offline using the pretrained UnSAM+ whole-image segmentation model. We directly use the publicly released UnSAM+ checkpoint `unsam_plus_sa1b_1perc_ckpt_50k.pth` for inference, without modifying the original model architecture. The model predicts segmentation regions for each input image, and proposals with confidence scores higher than 0.5 are retained to construct a pixel-wise region ID map. Pixels not assigned to any region are treated as background. The region IDs are independently assigned for each image and are not associated across different views or frames. During training, the generated pseudo segmentation labels are directly used for front-view and side-view supervision.

Detailed Hungarian Matching Procedure. Since the learnable queries in ICOcc do not have fixed identities, there is no predefined correspondence between queries and pseudo segmentation regions. In particular, the region IDs generated by UnSAM+ are independently assigned for each image and do not maintain consistency across different views or training samples. Therefore, directly assigning each region to a specific query is infeasible. We adopt Hungarian matching to establish permutation-invariant correspondence between pseudo segmentation regions and rendered segmentation during training.

For each training sample, the matching is performed online. Given the sampled pixel set $\mathcal{P}$ with pseudo region labels y_i and the rendered query probabilities p_{iq}, we first compute the affinity between each foreground region and each object query:

$$A_{kq} = \frac{1}{|\mathcal{P}_k|} \sum_{i \in \mathcal{P}_k} p_{iq}, \qquad \mathcal{P}_k = \{i \in \mathcal{P} : y_i = k\}, \tag{24}$$

where k denotes a foreground pseudo segmentation region and q denotes an object query. The Hungarian algorithm is then applied to the affinity matrix $\boldsymbol{A}$ to find the maximum-affinity assignment between regions and queries.

The background region is handled separately. Pixels with pseudo label 0 are directly assigned to the first query, which is designated as the background query. The Hungarian matching is only performed between foreground regions and the remaining object queries. After matching, each foreground pixel is supervised by the query assigned to its corresponding region. Since the assignment is a discrete operation, the matching step itself does not participate in gradient computation, while the subsequent probability-based supervision remains fully differentiable.

The matching is performed independently for each training sample and does not create a fixed mapping between pseudo segmentation regions and queries. This permutation-invariant design allows the queries to discover different scene regions without requiring predefined semantic identities.

Query Point Sampling Details. During training, query points are sampled along rays generated from sampled image pixels. Previous occupancy methods typically adopt patch-based pixel sampling, where several continuous image patches are randomly selected to construct training rays. However, this strategy may produce limited pseudo segmentation diversity, as a sampled patch can contain only a small number of segmentation regions. Such limited region diversity makes the Hungarian matching between pseudo segmentation labels and occupancy queries less reliable, and also reduces the variety of spatial structures observed during training.

To address this issue, we replace patch sampling with a grid-offset sampling strategy, as illustrated in Fig. 6. Instead of sampling continuous patches, we divide each image into regular grid cells with the same spatial scale as the original patches and randomly select one pixel with the same offset inside each cell. Therefore, each sampling operation covers the entire image while maintaining a controlled number of sampled rays. Compared with local patch sampling, this strategy provides a more diverse set of pseudo segmentation regions within each training iteration, leading to more stable query-region matching and improved supervision for instance decoupling.

After obtaining sampled pixels, we follow previous occupancy prediction methods (Wimbauer et al., 2023) to sample 3D query points along the corresponding camera rays as follows:

$$t_i = 1/\left(\left(1 - \frac{i+r}{N}\right)\frac{1}{t_n} + \frac{i+r}{N}\frac{1}{t_f}\right), \tag{25}$$

where N denotes the number of points sampled per ray, $r \sim \mathcal{U}(-0.5, 0.5)$ denotes a random variable from a uniform distribution, and t_n and t_f represent the near and far bounds in the rendering field, respectively. The sampled points are defined as $\boldsymbol{x}_i = \boldsymbol{o} + t_i\boldsymbol{d}$, where $\boldsymbol{o}$ denotes the camera origin and $\boldsymbol{d}$ represents the unit direction vector of the sampled ray.

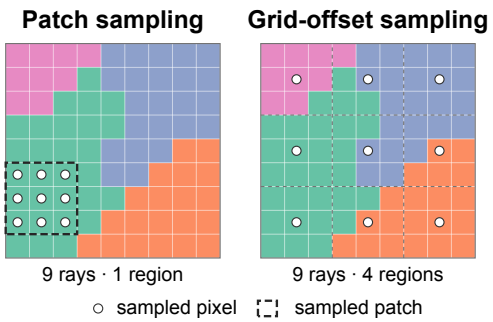


Figure 6: Comparison of patch and grid-offset sampling on a pseudo segmentation map.

E Additional Experiments and Visualization

E.1 Detailed Experiment Configurations

KITTI-360 and SSCBench-KITTI-360 Dataset Preprocessing. The SSCBench-KITTI-360 dataset (Li et al., 2024b) provides 3D occupancy ground truth along with the corresponding 2D images in KITTI-360. We follow Guo et al. (2026) to align the frame id and camera extrinsics between these two datasets.

CarlaOcc Dataset Preprocessing. In CarlaOcc, keyframes are taken from pose lists filtered by a 0.05 m inter-frame translation threshold, and the final 30 retained frames of each sequence are excluded in the training set. These selections provide suitable temporal spacing and scene coverage for constructing instance-anchored view groups with reliable cross-view instance correspondences. As discussed in Sec. 4.1, we construct a sequence-disjoint Town10HD split from the filtered pose lists. Training uses 8860 retained keyframes from sequences 00–02, 05–10, and 12, excluding the last 30 retained frames of each sequence; both front cameras yield 17,720 training entries. Evaluation uses 500 left-front-camera frames from held-out sequences 03, 04, and 11, with respective quotas of 125, 125, and 250. Within each held-out sequence, we sample one frame from each of uniformly partitioned temporal-index bins using seed 42. This sampling is designed to balance the distribution of dynamic objects across sequences with different traffic densities.

Additional Network Details. We use the registered DINOv2 ViT-B/14 model and extract intermediate representations from layers 3, 5, 7, 9, and 11. The multi-level features are passed through convolutional adapters and a subsequent skip-connected decoder, which outputs a 64-channel feature map at the input resolution. The pretrained ViT weights are frozen throughout training, while the convolutional adapters and skip-connected decoder are trainable. Both positional codes use six frequency bands with a frequency factor of 1.5 and include the original input. Each 39-dimensional code is embedded into 32 dimensions.

Additional Training Configuration. Each training sample is constructed from an instance-anchored view group. Among the input views, only the first perspective image is used for feature extraction by the image encoder. The remaining views provide multi-view supervision. At each iteration, one stereo-temporal perspective pair is randomly selected for ray sampling, while the complementary perspective views and side-camera views are used as rendering targets. This design provides both temporal and stereo observations for occupancy learning, while side-camera views introduce additional geometric and segmentation constraints.

For rendering supervision, the sampled 3D points are projected into the selected rendering views to obtain RGB observations. The perspective views provide individual photometric targets, while the side-camera views are aggregated to provide additional multi-view constraints. The selected side-camera views also provide pseudo-depth and pseudo-segmentation supervision for the instance-anchored learning strategy.

Detailed Evaluation Protocols. At evaluation, the model is run on single-frame samples with ground-truth voxel labels. The voxel grid has resolution $256 \times 256 \times 32$ and voxel size 0.2 m. Occupancy is thresholded at 0.5 and evaluated within the configured spatial range $x \in -4.5, 4.5$ m and $z \in 3, 50$ m. We report metrics as follows:

$$\mathrm{O}_{\mathrm{Acc}} = \frac{\sum_i \left[\hat{o}^{(i)} = o^{(i)}\right] m_f^{(i)}}{\sum_i m_f^{(i)}}, \tag{26}$$

$$\mathrm{O}_{\mathrm{Prec}} = \frac{\sum_i o^{(i)} \hat{o}^{(i)} m_f^{(i)}}{\sum_i \hat{o}^{(i)} m_f^{(i)}}, \tag{27}$$

$$\mathrm{IE}_{\mathrm{Rec}} = \frac{\sum_i (1-\hat{o}^{(i)})(1-o^{(i)})(1-m_v^{(i)}) m_f^{(i)}}{\sum_i (1-o^{(i)})(1-m_v^{(i)}) m_f^{(i)}}, \tag{28}$$

where $\hat{o}^{(i)}$ and $o^{(i)}$ denote the predicted and ground-truth occupancies for the i-th voxel, respectively. $m_v^{(i)}$ and $m_f^{(i)}$ are the visibility and frustum masks, respectively, which are generated following Guo et al. (2026). For semantic evaluation, the 32 queries are matched to the 15 semantic classes in SSCBench-KITTI-360. The Hungarian matching is operated per sample. Camera-ray IoU is computed at depth-error thresholds of 1, 2, and 4 m, using a ray stride of four pixels.

E.2 Additional Experimental Results

Cross-scene Semantic Consistency of Query Slots. The class-wise occupancy prediction results in Sec. 4 require a Hungarian matching between predicted query slots and semantic classes in 3D occupancy ground truth during evaluation. Although such matching is necessary due to the permutation-invariant nature of unsupervised models, it remains unclear whether ICOcc learns meaningful object-level shape priors or simply performs clustering as in previous scene-level approaches such as SceneDINO (Jevtić et al., 2025). To investigate this property, we analyze whether the same query identity maintains consistent semantic composition across different scenes. Note that this experiment does not assume that a query slot has a fixed correspondence to a ground-truth semantic category. Instead, the semantic occupancy annotations are only used as a reference to measure the stability of the learned slot representations.

For each frame in the test set, we first collect the voxel-level semantic composition of each non-background query slot. Specifically, we construct a slot-category count tensor $R \in \mathbb{N}^{S\times Q\times C}$, where S denotes the number of test frames, Q denotes the number of non-background query slots, and C denotes the number of evaluation semantic categories. For SSCBench-KITTI-360, $C = 15$. Each element in R is computed as:

$$R_{s,q,c} = \sum_{\boldsymbol{v}\in \mathrm{ROI}_s} \mathbb{1}[p_{\mathrm{occ}}(\boldsymbol{v}) \geq 0.5]\mathbb{1}[\hat{q}(\boldsymbol{v}) = q]\mathbb{1}[g_s(\boldsymbol{v}) = c], \tag{29}$$

where $g_s(\boldsymbol{v})$ is the ground-truth semantic category on the voxel $\boldsymbol{v}$ in the test sample s. ROI_s denotes the region of interest in the same sample. $\mathbb{1}[\cdot]$ is the Iverson Bracket. Empty and invalid voxels are excluded from these statistics. We further obtain the semantic composition of each query slot in each frame. A frame-slot pair is considered valid if the corresponding query occupies at least one voxel:

$$\mathcal{S}_q = \left\{ s \mid \sum_{c=1}^{C} R_{s,q,c} > 0 \right\}, \tag{30}$$

where $\mathcal{S}_q$ denotes the set of valid frames for query slot q. For each valid frame-slot pair, the semantic composition is normalized as:

$$p_{s,q,c} = \frac{R_{s,q,c}}{\sum_{c'=1}^{C} R_{s,q,c'}}, \tag{31}$$

where $p_{s,q,c}$ represents the proportion of semantic category c within query slot q in frame s.

The first metric measures cross-frame semantic consistency. It evaluates whether the same query slot preserves a similar semantic composition across different scenes. For each query slot q, we construct a frame-by-category matrix $\mathbf{P}^{(q)} \in \mathbb{R}^{M_q\times C}$, where each row corresponds to the semantic composition vector $\boldsymbol{p}_{s,q} = [p_{s,q,1}, ..., p_{s,q,C}]$ of the same query slot observed in one valid frame, and $M_q = |\mathcal{S}_q|$ denotes the number of valid observations of query q. The consistency score is defined as the dominant singular-value energy ratio:

$$C_q = \frac{\sigma_1(\mathbf{P}^{(q)})^2}{\sum_j \sigma_j(\mathbf{P}^{(q)})^2}, \tag{32}$$

where σ_j denotes the j-th singular value of $\mathbf{P}^{(q)}$. The score is bounded in the range of $(0, 1]$. When a query slot maintains an identical semantic composition across all observed frames, all rows of $\mathbf{P}^{(q)}$ become identical and the matrix is approximately rank-one. Consequently, the first singular value contains all matrix energy and C_q approaches 1. In contrast, when the same query slot corresponds to different semantic compositions across frames, the energy is distributed among multiple singular values, resulting in a lower C_q. Therefore, a higher C_q indicates that query slot q preserves a more stable semantic composition across different scenes.

Table 4: Quantitative cross-scene semantic consistency analysis of ICOcc on KITTI-360.

Method	Consistency ↑ (%)	Purity ↑ (%)	Switching Gap ↓ (%)
ICOcc Slots	74.4	88.6	19.9
Random Slot Permutation	38.0	88.6	54.2

Second, we measure within-frame semantic purity:

$$U_q = \frac{1}{M_q} \sum_{s \in \mathcal{S}_q} \max_c p_{s,q,c}. \tag{33}$$

The purity score measures whether the voxel assignment of query slot q in an individual frame is dominated by a single semantic category. Therefore, U_q reflects the granularity consistency between the learned slot partition and the semantic partition defined by the dataset annotations. A higher U_q indicates that the learned slots tend to align with category-level object regions rather than mixing multiple semantic categories within the same slot.

Different from the cross-frame consistency score C_q, U_q does not evaluate whether the same query slot preserves its semantic composition across different frames. Therefore, randomly permuting slot identities across frames does not affect U_q, since the permutation preserves the semantic composition within each individual slot and only changes the correspondence between slots across frames.

Third, we quantify the semantic switching gap:

$$\bar{p}_{q,c} = \frac{1}{M_q} \sum_s p_{s,q,c}, \tag{34}$$

$$G_q = U_q - \max_c \bar{p}_{q,c}. \tag{35}$$

The first term allows each frame to independently select its dominant category, while the second term represents the dominant category shared across all frames. Therefore, G_q measures the inconsistency caused by enforcing a single category identity for the slot.

The final model-level scores are obtained by weighted averaging over valid slots:

$$C = \sum_q w_q C_q, \qquad U = \sum_q w_q U_q, \qquad G = \sum_q w_q G_q, \tag{36}$$

where w_q is proportional to the number of valid observations of slot q.

To verify that the measured consistency is caused by meaningful slot identities rather than accidental clustering, we further introduce a random slot permutation baseline. Specifically, we independently shuffle the non-background slot identities in each frame while preserving the voxel assignment and semantic composition of each slot. This operation does not change the single-frame segmentation quality, but destroys the correspondence of slot identities across different frames.

As shown in Table 4, the original slot identities achieve substantially higher cross-frame consistency and lower semantic switching gap than the random permutation baseline. Meanwhile, the within-frame purity remains almost unchanged after permutation, as expected, since the semantic composition of each individual prediction is preserved. These results demonstrate that the learned query identities contain non-random cross-scene organization. Although a query slot is not explicitly assigned to a predefined semantic category during training, the same slot ID tends to represent similar types of objects across different scenes. Therefore, ICOcc learns object-level representations with stable geometric structures, rather than performing independent segmentation clustering for each test sample.

Additional Qualitative Results. Figure 7 presents additional qualitative comparisons with the previous method SceneDINO. Our approach produces clearer object boundaries and significantly reduces trailing artifacts. In particular, SceneDINO often generates fragmented or over-extended occupancy regions around object boundaries, causing neighboring objects or background regions to be incorrectly connected. In contrast, ICOcc reconstructs more compact object geometries and better preserves the separation between individual instances, especially for vehicle regions. These results further demonstrate that the proposed instance-centric formulation enables more accurate object-level geometry modeling and reduces the ambiguity caused by scene-level occupancy reconstruction.

Table 5: Ablation results for class-wise occupancy prediction on selected object categories in KITTI-360.

Configuration	Class-wise RayIoU (%) ↑					Avg. RayIoU (%) ↑
	Car	Motor.	Truck	Other-Veh.	Building	
Instance-Anchored Learning						
w/o Side-View Retrieval	20.4	0.0	9.7	1.8	9.7	8.3
w/o $\mathcal{L}_{\text{free}}$	14.0	0.0	6.1	1.6	10.3	6.4
w/o $\mathcal{L}_{\text{ins}}$	9.1	0.0	3.3	0.2	4.2	3.4
w/o $\mathcal{L}_{\text{surf}}$	22.6	0.5	10.7	3.5	10.2	9.5
Instance Decoupling Network						
w/o UVZ Positional Encodings	20.9	0.0	12.7	1.3	9.0	8.8
w/o XYZ Positional Encodings	20.3	0.0	10.6	1.5	7.5	8.0
Instance-Centric Categorical Occupancy Formulation						
Predicting Empty-Space Scores	23.6	1.3	11.8	6.2	11.5	10.9
w/o $\log \delta_i$ correction term	21.3	0.0	8.8	3.8	11.5	9.1
Full ICOcc	24.3	3.2	13.1	6.9	12.8	12.1

E.3 Additional Ablation Studies

We further conduct detailed ablation studies on the proposed components, as summarized in Table 5. The results consistently demonstrate the effectiveness of each design choice for instance-centric occupancy prediction.

For instance-anchored learning, removing the 3D instance-guided side-view retrieval reduces the average RayIoU across the selected categories from 12.1% to 8.3%, indicating that retrieving side views with shared 3D instances provides additional geometric constraints and improves instance shape reconstruction. Removing $\mathcal{L}_{\text{free}}$ also degrades the performance, indicating that explicit empty-space supervision helps reduce erroneous occupancy predictions. The removal of $\mathcal{L}_{\text{ins}}$ causes the largest performance drop, reducing the average RayIoU to 3.4%. This demonstrates that instance-level supervision is essential for maintaining instance separation and learning object-specific geometry.

For the instance decoupling network, removing either UVZ or XYZ positional encodings consistently decreases the class-wise RayIoU. This indicates that both image-space and 3D positional information contribute to accurate instance geometry modeling.

For the instance-centric categorical occupancy formulation, directly predicting empty-space scores instead of modeling empty occupancy as an additional category reduces the average RayIoU from 12.1% to 10.9%. Furthermore, removing the interval-length compensation term $\log \delta_i$ causes additional degradation, especially for vehicle categories, demonstrating the importance of compensating for non-uniform ray sampling in categorical occupancy prediction.



Figure 7: Qualitative comparisons of unsupervised semantic occupancy prediction in ROI in additional scenes: (a) input RGB images; (b) SceneDINO results; (c) our results.