OccAny: Generalized Unconstrained Urban 3D Occupancy

Anh-Quan Cao Tuan-Hung Vu
Valeo.ai, Paris, France
https://valeoai.github.io/OccAny

Abstract

Relying on in-domain annotations and precise sensor-rig priors, existing 3D occupancy prediction methods are limited in both scalability and out-of-domain generalization. While recent visual geometry foundation models exhibit strong generalization capabilities, they were mainly designed for general purposes and lack one or more key ingredients required for urban occupancy prediction, namely metric prediction, geometry completion in cluttered scenes and adaptation to urban scenarios. We address this gap and present OccAny*, the first unconstrained urban 3D occupancy model capable of operating on out-of-domain uncalibrated scenes to predict and complete metric occupancy coupled with segmentation features.* OccAny *is versatile and can predict occupancy from sequential, monocular, or surround-view images. Our contributions are three-fold: (i) we propose the first generalized 3D occupancy framework with (ii) Segmentation Forcing that improves occupancy quality while enabling mask-level prediction, and (iii) a Novel View Rendering pipeline that infers novel-view geometry to enable test-time view augmentation for geometry completion. Extensive experiments demonstrate that* OccAny *outperforms all visual geometry baselines on 3D occupancy prediction task, while remaining competitive with in-domain self-supervised methods across three input settings on two established urban occupancy prediction datasets.* ***Our code is available at https://github.com/valeoai/OccAny*** .

1. Introduction

The innate ability to see and make sense of the world in three dimensions underpins how humans understand and navigate the space. Advancing 3D scene understanding is crucial for spatial intelligent systems such as autonomous driving, robotics, and augmented reality. A key task in this area is 3D occupancy prediction whose goal is to infer a voxelized map of the environment and, when required, provide the corresponding semantics. Despite advances in architecture design [11, 42, 53, 64], training algorithm [5, 24, 27, 40, 68] and dataset [4, 10, 12, 16], current state-of-the-art 3D models still lack the generalization of human perception, typically requiring constrained setup with precise sensor calibration. While humans can effortlessly infer complex 3D structures in any novel scenes, replicating this capability remains a demanding problem.

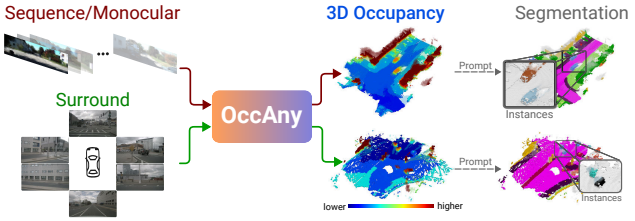


Figure 1. **OccAny** is a generalized 3D occupancy model that is trained once and can operate on out-of-domain sequential, monocular, or surround-view urban images. It produces SAM2-like features, enabling promptable segmentation.

State-of-the-art supervised approaches for 3D occupancy prediction [20, 29, 31, 60, 61, 68, 72] achieve remarkable results when the training and test data are drawn from the same distribution, *i.e*. both are collected using the same or a similar sensor rig under comparable conditions. A core component of these methods is the lifting of 2D features into 3D space, performed either via learnable mechanisms [20, 31] or via explicit camera modeling [5, 70]. However, this lifting operation inherently embeds sensor- and domain-specific biases into the models, which limits their ability to generalize to new sensor suites or environments. Recent self-supervised works [6, 14, 21, 25, 62] remove the need for 3D supervision by formulating occupancy prediction as a differentiable volume-rendering problem, thereby leveraging advances in neural rendering [27, 40]. Despite this, self-supervised models still struggle to generalize, as they remain specialized to a particular training domain with strong biases in camera poses and intrinsic parameters. As we look toward a near future with millions of autonomous fleets equipped with different sensor configurations, advancing 3D occupancy prediction requires generalizable and efficient solutions capable of leveraging heterogeneous training data to overcome

current generalization barriers.

The advent of visual geometry foundation models [3, 56, 58, 59], built around the concept of direct pointmap prediction, has demonstrated the strong generalization potential of large-scale transformer networks for 3D scene understanding. However, their general-purpose design remains insufficient for urban occupancy prediction, which simultaneously requires metric-scale accuracy, cluttered geometry completion, and adaptation to the complex nature of urban environments.

We introduce a novel pipeline for urban 3D occupancy prediction that emphasizes scalability and generalization. Our approach follows the recipe of geometry foundation models that train visual transformers with straightforward point-level objectives on diverse, large-scale datasets. Unlike those prior works, we specialize in the task of occupancy prediction and focus exclusively on outdoor urban datasets, which we argue is essential for optimal adaptation to the unique characteristics of urban scene perception. A major challenge in outdoor urban scenarios is the sparsity of supervised LiDAR point clouds, which leads to irregular predictions in non-supervised regions and exacerbates the difficulty of geometry completion, particularly in highly cluttered areas. To address this, we introduce *Segmentation Forcing*, a distillation strategy that enriches geometry-focused features with segmentation awareness and thus helps regularize predictions with consistent segmentation cues of object instances and homogeneous regions. For geometry completion, we develop a *Novel View Rendering* pipeline that infers arbitrary novel-view geometry from a global scene memory. Our rendering pipeline enables Test-time View Augmentation, allowing us to densify and complete scenes at both the point- and voxel-levels. Fig. 1 illustrates our model. In summary, our contributions are three-fold:

- We propose a generalized 3D occupancy framework, *OccAny*, the first designed to infer dense 3D occupancy and segmentation features for out-of-domain unconstrained urban scenes. A unified *OccAny* model can operate on either sequential, monocular or surround-view images.
- We introduce *Segmentation Forcing*, a novel regularization strategy to mitigate the sparsity of LiDAR supervision.
- We develop a *Novel View Rendering* pipeline targeting geometry completion.

OccAny is trained on five urban datasets and evaluated on two out-of-distribution occupancy datasets: SemanticKITTI and Occ3D-NuScenes. *OccAny* significantly outperforms baseline visual geometry networks and performs on par with domain-specific SOTA self-supervised occupancy networks trained directly on SemanticKITTI and Occ3D-NuScenes.

2. Related works

Visual geometry foundation model. Dust3r [59] introduced the visual geometry foundation model, which uses large-scale pointmap prediction to solve diverse 3D tasks. Research has rapidly expanded this paradigm beyond static, binocular inputs in several directions. One branch addresses dynamics by handling moving scenes [50, 74], dynamic video pose estimation [63], and camera rigs [30]. A major thrust has been multi-frame processing through feed-forward, sequential, and memory-based architectures [3, 55, 56, 58, 69]. Other works have explored downstream tasks such as indoor instance prediction [78] and image matching [28], or have leveraged known camera parameters [23]. While some methods explore novel view synthesis [26, 58], they often prioritize image synthesis over geometric fidelity [26] or exhibit limited applicability [58]. Unlike these approaches, we repurpose these models for occupancy prediction by introducing segmentation forcing to enhance geometric fidelity while enabling segmentation output. We further propose a novel pointmap rendering pipeline to enable complete geometry beyond visible scenery.

3D occupancy prediction . This task, which originates from 3D scene completion [49], aims to assign an occupancy state to each voxel in a 3D volume. Initially proposed for indoor depth scenes [49], it expanded to outdoor LiDAR [1, 7, 45, 65] and was later adapted for multi-view images [5]. Subsequent supervised research has focused on projection mechanisms [5, 31, 70], efficient representations [20, 22, 34, 47, 77], network architectures [31, 37, 75], and benchmark creation [32, 36, 54]. However, these methods' reliance on dense, voxel-wise annotations limits their scalability.

Self-supervised methods mitigate this label dependency by training on posed images, often via volume rendering [6, 62]. Subsequent NeRF-based approaches have improved performance through better losses [18, 21, 73], optimized ray sampling [6, 67, 73], and enhanced representations by distilling foundation models [24, 48, 57]. More recently, 3D Gaussian Splatting has emerged as a more efficient alternative to NeRF [9, 14]. However, these approaches generally require precise camera information and in-domain training data. [14] is a partial exception, avoiding 6D poses via camera overlap, but still requires camera intrinsics and domain-specific information (*i.e.*, adjacent camera overlap). Other works [25, 39, 71, 76] focus on pseudo-label generation, using open-vocabulary foundation models [25, 76] and sequence-level bundle adjustment [39]. While models trained on these pseudo-labels show promising cross-dataset generalization, they remain limited to specific settings.

3. Method

We build OccAny, a 3D occupancy framework that can generalize to arbitrary out-of-domain urban scenes. To this end, we adopt the transformer architecture from the Dust3r family and train the model on multiple urban datasets using standard point-level objectives commonly employed in prior works [3, 56, 59]. OccAny is supervised with metric-scale point-clouds enabling metric predictions at test time, a key

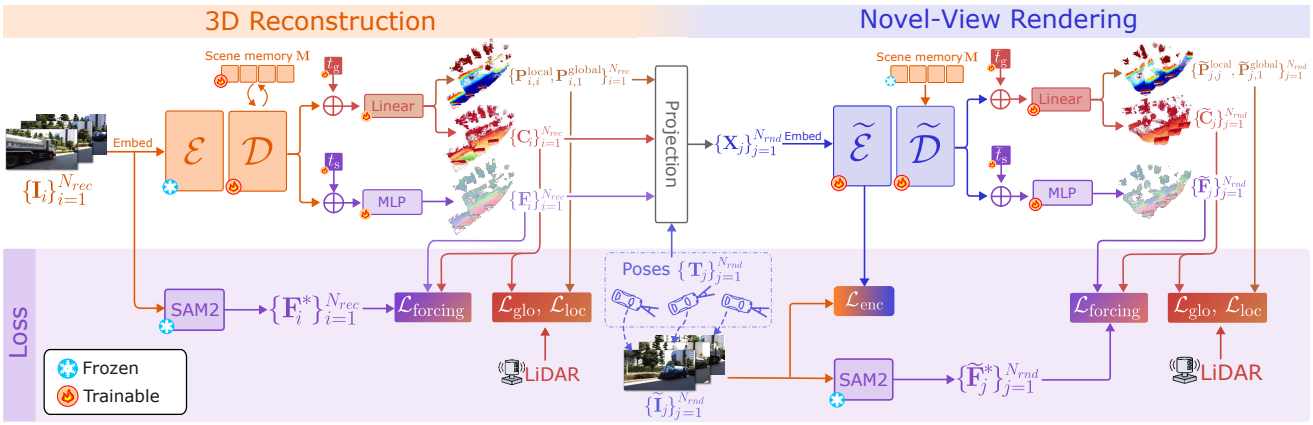


Figure 2. **OccAny Training** is done in two stages: (i) *3D Reconstruction* infers 3D scene using N_{rec} reconstruction frames and (ii) *Novel-View Rendering* renders geometry of N_{rnd} new views having camera poses $\{\mathbf{T}_j\}_{j=1}^{N_{rnd}}$. *Segmentation Forcing* with SAM2 features helps regularize and improve geometry prediction. The scene memory $\mathbf{M}$ is dynamically updated during reconstruction, while during rendering, the final scene memory output from the reconstruction stage is used without updating

element in occupancy prediction. We propose two novel strategies *Segmentation Forcing* and *Novel View Rendering* to accommodate the unique characteristics of 3D occupancy prediction in urban environments.

Fig. 2 illustrates OccAny training process, which consists of two stages: *3D Reconstruction* and *Novel View Rendering*. For each frame sequence, we randomly select N frames for training. In the reconstruction stage, we set the number of reconstruction frames to $N_{rec} = N$. In the rendering stage, we use non-overlapping sets of N_{rec} reconstruction frames and N_{rnd} rendering frames, with $N = N_{rec} + N_{rnd}$.

3.1. 3D Reconstruction with Segmentation Forcing

The *3D Reconstruction* stage aims to recover the scene geometry from a set of reconstruction frames, providing the geometry basis for the subsequent novel-view rendering stage. In this stage, OccAny extends MUSt3R [3], a multi-view geometry network, by adding a SAM2 feature prediction head. SAM2 [43] is a foundation model designed for promptable visual segmentation in images and videos; its features are thus rich in high-fidelity segmentation cues and are beneficial for resolving geometric ambiguity. The *Segmentation Forcing* loss compels OccAny to predict SAM2-like features. Our strategy regularizes geometry prediction by leveraging segmentation cues to enforce spatial and temporal feature consistency, thereby improving performance, especially in regions where LiDAR supervision is sparse.

OccAny processes N_{rec} reconstruction frames $\{\mathbf{I}_i\}_{i=1}^{N_{rec}} \in \mathbb{R}^{H\times W\times 3}$ as multi-view inputs to reconstruct the 3D scene. We feed N_{rec} frames in chronological order through a shared reconstruction encoder $\mathcal{E}$ followed by a shared decoder $\mathcal{D}$. The first frame is always designated as the reference frame; all non-reference frames are identified by a specialized token added at the beginning of the shared decoder. The two transformers produce, for each frame $\mathbf{I}_i$:

- SAM2-like feature maps $\mathbf{F}_i \in \mathbb{R}^{H'\times W'\times C}$,
- global pointmaps $\mathbf{P}_{i,1}^{\text{global}} \in \mathbb{R}^{H\times W\times 3}$ in the global camera coordinate of the reference frame 1,
- local pointmaps $\mathbf{P}_{i,i}^{\text{local}} \in \mathbb{R}^{H\times W\times 3}$ in the local camera coordinate of the current frame i,
- confidence maps $\mathbf{C}_i \in \mathbb{R}^{H\times W}$,
- and camera poses $\mathbf{v}_i \in \mathbb{R}^7$ inferred by registering the global and local pointmaps.

For each frame $i \in [3, N_{rec}]$, a scene memory $\mathbf{M}_{i-1}$ of all historical reconstruction frames $1..i-1$ is used in the decoding process to infer the geometry of the current frame i via cross-attention between tokens of frame i and memory tokens in $\mathbf{M}_{i-1}$. The scene memory $\mathbf{M}_i$ is then constructed by concatenating $\mathbf{M}_{i-1}$ with the decoder tokens of the current frame i. To initialize, $\mathbf{M}_2$ is formed by concatenating the decoder tokens of the first two frames. With a slight abuse of notation, we use $\mathbf{M}$ without a subscript to denote the final global scene memory, which aggregates information from the entire sequence; that is, $\mathbf{M} \equiv \mathbf{M}_{N_{rec}} \in \mathbb{R}^{H'\times W'\times (C\cdot N_{rec})}$.

The decoder is followed by linear heads for pointmap and confidence prediction, and an MLP head for SAM2-like feature prediction. Because the geometry and segmentation tasks differ in nature, we introduce two learnable task tokens: t_{g} for the pointmap heads and t_{s} for the SAM2 head. These tokens are added to all decoder tokens before the corresponding head is applied. For clarity, we omit task tokens in the equations and only visualize them in Fig. 2.

The SAM2 head consists of an MLP with two linear layers followed by two upsampling layers. Each upsampling layer uses bilinear interpolation to resize the features, followed by a convolution, layer norm, and GELU.

In summary, the output of this stage is:

$$\mathcal{D}(\mathcal{E}(\{\mathbf{I}_i\}_{i=1}^{N_{rec}})) = \left(\mathbf{M}, \{\mathbf{F}_i, \mathbf{P}_{i,1}^{\text{global}}, \mathbf{P}_{i,i}^{\text{local}}, \mathbf{C}_i, \mathbf{v}_i\}_{i=1}^{N_{rec}}\right). \quad (1)$$

3.2. Novel-View Rendering

We train a rendering encoder $\widetilde{\mathcal{E}}$ and decoder $\widetilde{\mathcal{D}}$ to predict pointmaps and SAM2-like features for arbitrary novel views along the reconstruction camera trajectories $\{\mathbf{v}_i\}_{i=1}^{N_{rec}}$ (*cf*. Eq. (1)). The reconstruction modules $\mathcal{E}$, $\mathcal{D}$ are frozen, and their outputs serve as inputs to the rendering stage.

During training, we sample N_{rec} reconstruction frames and N_{rnd} rendering frames from the same sequence; the first frame always belongs to the reconstruction set. Let $\{\mathbf{T}_j\}_{j=1}^{N_{rnd}}$ be the camera poses of rendering frames $\widetilde{\mathbf{I}}_j$. Our goal is to render pointmaps and SAM2-like features for each $\mathbf{T}_j$, conditioned on reconstruction outputs. Rendering frames are used only for loss computation.

Tokenization. We merge the global pointmaps $\{\mathbf{P}_{i,1}^{\text{global}}\}_{i=1}^{N_{rec}}$ into a single point cloud $\mathbf{P}^{\text{global}}$ in the reference-frame coordinate system. Projecting $\mathbf{P}^{\text{global}}$ into $\{\mathbf{T}_j\}_{j=1}^{N_{rnd}}$ yields N_{rnd} xyz-images and point-to-pixel correspondences, enabling 2D projection of SAM2-like features and confidence maps into each novel view. Each modality image is processed by an MLP; the results are concatenated and linearly projected to form novel-view tokens $\{\mathbf{X}_j\}_{j=1}^{N_{rnd}}$. RoPE is used for positional encoding universally.

Rendering. Because reconstruction frames cover the scene only partially, projected novel views contain missing areas and projection artifacts. The rendering transformers learn to complete missing geometry and correct projection errors, producing denser pointmaps.

The rendering encoder $\widetilde{\mathcal{E}}$ contains 6 transformer blocks, processing novel-view token representations $\mathbf{X}$ to predict encoder tokens. During training, we distill knowledge from the large reconstruction encoder $\mathcal{E}$ (24 transformer blocks) to the small rendering encoder $\widetilde{\mathcal{E}}$ through the $\mathcal{L}_{\text{enc}}$ loss (defined in Sec. 3.4). This helps facilitate the optimization process by providing an auxiliary supervision signal, encouraging the rendering encoder to mimic the tokens produced by the larger teacher reconstruction encoder.

The rendering decoder $\widetilde{\mathcal{D}}$ has the same architecture as the reconstruction decoder $\mathcal{D}$ and is initialized from its weights. We also introduce two learnable task tokens $\widetilde{t}_{\text{g}}$ and $\widetilde{t}_{\text{s}}$, initialized from t_{g} and t_{s}. The scene memory $\mathbf{M}$ obtained from the reconstruction stage remained fixed (*cf*. Eq. (1)) and is used by $\widetilde{\mathcal{D}}$ to render the final set of outputs. During decoding, $\widetilde{\mathcal{D}}$ applies cross-attention between decoder tokens and the memory tokens in $\mathbf{M}$, making possible reference to the whole reconstructed scene. Intuitively, the explicit reconstruction outputs from the previous stage guides the rendering, while the implicit memory provides supporting information to correct and complete the scene. *Segmentation Forcing* is also applied to regularize novel-view predictions.

In summary, output of the rendering stage is written:

$$\widetilde{\mathcal{D}}(\mathbf{M}, \widetilde{\mathcal{E}}(\{\mathbf{X}\}_{j=1}^{N_{rnd}})) = \{\widetilde{\mathbf{F}}_j, \widetilde{\mathbf{P}}_{j,1}^{\text{global}}, \widetilde{\mathbf{P}}_{j,j}^{\text{local}}, \widetilde{\mathbf{C}}_j\}_{j=1}^{N_{rnd}}. \quad (2)$$

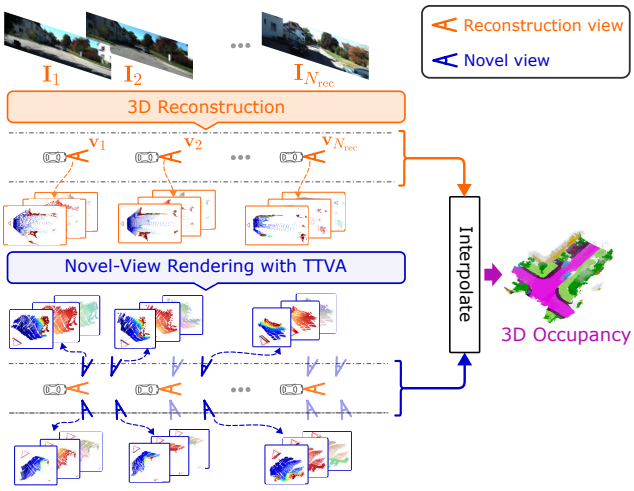


Figure 3. **OccAny inference** undergoes two stages: (i) 3D reconstruction to retrieve N_{rec} pointmaps with predicted camera poses $\{\mathbf{v}_i\}_{i=1}^{N_{rec}}$, and (ii) novel-view rendering with TTVA sampled along the trajectory of $\{\mathbf{v}_i\}_{i=1}^{N_{rec}}$. 3D occupancy is obtained by aggregating all pointmaps and voxelizing them with trilinear interpolation.

3.3. OccAny Inference

We first retrieve the reconstructed pointmaps, SAM2-like features and the registered camera poses of all N_{rec} input frames from the *3D Reconstruction* stage. We then randomly sample novel views around the predicted camera trajectory $\{\mathbf{v}_i\}_{i=1}^{N_{rec}}$ (*cf*. Eq. (1)) and pass them through the *Novel View Rendering* (NVR) stage to infer the novel-view pointmaps and segmentation features (*cf*. Eq. (2)). The final 3D occupancy is obtained by aggregating all pointmaps from both stages and voxelizing them into a dense grid via trilinear interpolation. The inference protocol is visualized in Fig. 3. OccAny is versatile and can predict 3D occupancy for either sequential, monocular, or surround-view inputs. Predicted SAM2-like features can be directly used for segmentation.

NVR Inference. Thanks to NVR, we can use arbitrary views at test-time to help infer occlusion; this strategy is coined Test-time View Augmentation (TTVA). We first position novel camera views uniformly every ρ_{fwd} meters along a straight path along the trajectory of predicted poses $\{\mathbf{v}_i\}_{i=1}^{N_{rec}}$. At each of those N_{fwd} sampled positions, we vary horizontal viewing angles $\Phi{=}\{0, \pm\phi\}$ and shift the camera by a lateral amount of $\pm\rho_{\text{lat}}$. Fig. 6 illustrates the NVR setups.

Segmentation w/ SAM2-like features. We apply Grounded SAM2 [44] pipeline by feeding the first frame to GroundingDINO [35] and obtain candidate bounding boxes of all semantic classes of interest. We then use the pretrained prompt decoder of SAM2 to prompt OccAny's predicted SAM2-like features with the obtained bounding boxes, resulting in dense semantic masks for the first frame. Semantic masks are then propagated through the entire scene with SAM2 video tracking. Finally we assign the predicted occupancy voxels with predicted semantic classes.

3.4. Training Losses

Both stages are trained using the same set of losses, *i.e.* global- and local- pointmap loss $\mathcal{L}_{\text{glo}}$, $\mathcal{L}_{\text{loc}}$, and Segmentation Forcing loss $\mathcal{L}_{\text{forcing}}$, with the exception of the rendering encoder distillation loss $\mathcal{L}_{\text{enc}}$, which is applied only in the rendering stage. We only describe common losses in the reconstruction stage for brevity.

Pointmap Losses $\mathcal{L}_{\text{glo}}$, $\mathcal{L}_{\text{loc}}$. The loss weights the difference between the predicted pointmap $\mathbf{P}_{i,1}^{\text{global}}$ and ground truth $\mathbf{P}_{i,1}^{*}$ using the predicted confidence map $\mathbf{C}_i$ [3]:

$$\mathcal{L}_{\text{glo}} = \frac{1}{|s|} \sum_{i=1}^{N_{rec}} \left\| \mathbf{C}_i \odot \left(\mathbf{P}_{i,1}^{\text{global}} - \mathbf{P}_{i,1}^{*} \right) \right\|_1 - \alpha \log(\mathbf{C}_i),$$

where $\odot$ denotes element-wise multiplication with channel-wise broadcasting, and α controls the regularization strength, and s is the normalization scale [3, 58]. The local pointmap loss $\mathcal{L}_{\text{loc}}$ is formulated identically.

Geometry-aware Segmentation Forcing Loss $\mathcal{L}_{\text{forcing}}$. We employ a Mean Squared Error (MSE) loss. We use the same confidence map $\mathbf{C}$ in pointmap losses above to weight the MSE error:

$$\mathcal{L}_{\text{forcing}} = \frac{1}{H'W'} \sum_{i=1}^{N_{rec}} \left\| \mathbf{C}_i \odot (\mathbf{F}_i - \mathbf{F}_i^{*}) \right\|_2^2, \qquad (3)$$

where N_{rec} is the number of reconstruction frames. Since $\mathbf{C}$ represents the geometry confidence learned by the pointmap head, our weighting forces the network to focus on high-confidence areas and ignore low-confidence ones like sky. We note that $\mathcal{L}_{\text{forcing}}$ does not update the confidence head.

Encoder Distillation Loss $\mathcal{L}_{\text{enc}}$. This loss distills knowledge from the larger teacher reconstruction encoder $\mathcal{E}$ (24 layers) to the smaller student rendering encoder $\widetilde{\mathcal{E}}$ (6 layers). It minimizes the squared L2 distance between the output tokens from both encoders. Given the output tokens from the rendering encoder $\widetilde{\mathcal{E}}(\{\mathbf{X}_j\}_{j=1}^{N_{rnd}})$ and the reconstruction encoder $\mathcal{E}(\{\widetilde{\mathbf{I}}_j\}_{j=1}^{N_{rnd}})$, the loss is written as:

$$\mathcal{L}_{\text{enc}} = \sum_{j=1}^{N_{rec}} \left\| \mathcal{E}(\widetilde{\mathbf{I}}_j) - \widetilde{\mathcal{E}}(\mathbf{X}_j) \right\|_2^2,$$

where $\{\widetilde{\mathbf{I}}_j\}_{j=1}^{N_{rec}}$ are the novel-view images.

4. Experiments

Training. OccAny is trained on a mixture of five urban datasets, using images from all cameras and projected LiDAR pointmap as ground truth: Waymo [51], DDAD [17], PandaSet [66], VKITTI2 [2], and ONCE [38].

In the reconstruction stage, we initialize with MUSt3R [3], freeze the encoder $\mathcal{E}$ and only train the decoder $\mathcal{D}$ for 3D reconstruction. Input frames are resized to 512-width with varying aspect ratios. We sample training sequences with minimum length N=6 and maximum length N=10. Frames are sampled at 2Hz in all datasets.

In the rendering stage, we initialize $\widetilde{\mathcal{D}}$ with the pretrained weights of $\mathcal{D}$. We keep the same sequence length $N \in [6, 10]$, and randomly select among those N_{rnd} frames as rendering views; the remaining $N_{rec} = N - N_{rnd}$ are used for reconstruction. The first frame serves as reference and it is always part of the reconstruction set.

Evaluation. We evaluate the generalization of OccAny on two out-of-domain benchmarks: SemanticKITTI [1] and Occ3D-NuScenes [54], detailed in Sec. A.1.

We use three evaluation settings:

- *Sequence:* a sequence of 5 frames coming from a single camera on SemanticKITTI and Occ3D-NuScenes,
- *Monocular:* a single input frame on SemanticKITTI,
- *Surround-view:* all surrounding frames at a single timestep on Occ3D-NuScenes.

NVR inference. In the *Sequence* and *Surround-view* settings, we use the augmentation strategy TTVA with $N_{\text{fwd}} = 10$, forward shift ρ_{fwd} of 3 m, and lateral shift ρ_{lat} of 2 m. In the *Monocular* setting, we sample denser and use $N_{\text{fwd}} = 50$, forward shift ρ_{fwd} of 1 m, lateral shift ρ_{lat} of 2 m. All settings use horizontal angle ϕ of $\{0°, \pm 60°\}$.

Baselines. We compare OccAny against four strong baselines: MUSt3R [3], CUT3R [58], VGGT [56], AnySplat [26], and Depth Anything 3 (DA3) [33]. Among them, CUT3R is trained only in the online setting. AnySplat is an VGGT extension with Gaussian Splatting [27] for novel view synthesis and for improving geometric consistency. MUSt3R and CUT3R output metric-scale pointmaps, whereas VGGT and AnySplat produce scale-invariant pointmaps. To resolve the scale ambiguity of VGGT and AnySplat, we calibrate their depth predictions with Metric3Dv2 [19] using their predicted camera intrinsics; those two variants are presented as VGGT† and AnySplat†. For DA3, we use DA3-LARGE to estimate global point map and DA3METRIC-LARGE for metric scaling. Since AnySplat and CUT3R support novel-view synthesis, we also apply our proposed TTVA strategy to improve those, referred to as CUT3R* and AnySplat*†. All models are tested on the same input resolution, with a very slight difference depending on the patch-size.

For reference, we also report published results from vision-based self-supervised occupancy models trained *in-domain*, which are heavily biased to dataset-specific characteristics especially camera intrinsics and extrinsics. We compare against self-supervised methods as both do not require in-domain 3D ground-truth for training. However, OccAny is completely zero-shot while self-supervised methods are trained on in-domain calibrated data.

Metrics. Similar to [5, 21], we use the standard 3D occupancy metrics Precision, Recall, and Intersection over Union

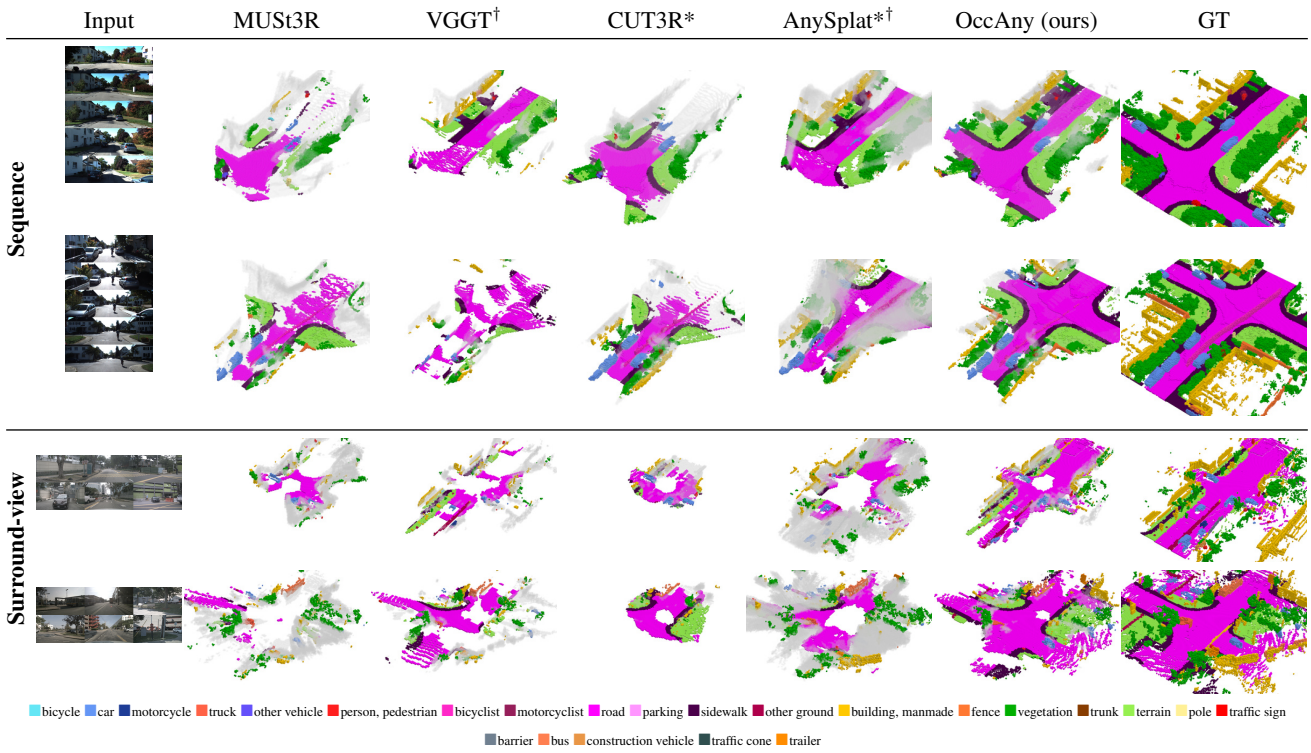


Figure 4. **Occupancy predictions** of OccAny and baselines on a sequence and a surround view. We visualize here predicted voxels. For qualitative analysis, we overlay the semantic ground-truth colors on predicted voxels to better highlight class-wise gains. False positive voxels are painted in gray without any overlayed color. Compared to baselines, our occupancy predictions are denser and more accurate.

Method	Venue	Semantic KITTI				Occ3D-NuScenes			
		Res.	Prec.	Rec.	IoU	Res.	Prec.	Rec.	IoU
MUSt3R [3]	CVPR'25	512x160	18.38	25.58	11.97	512x288	19.27	28.60	13.01
CUT3R [58]	CVPR'25	512x160	25.72	21.11	13.11	512x288	24.69	16.57	11.01
CUT3R* [58]	CVPR'25	512x160	27.05	27.92	15.93	512x288	29.44	30.50	17.62
VGGT† [56]	CVPR'25	518x168	36.35	22.62	15.20	518x294	38.34	26.23	18.45
AnySplat† [26]	TOG'25	518x168	18.22	35.62	11.67	518x294	26.67	36.93	18.33
AnySplat*† [26]	TOG'25	518x168	14.53	**47.48**	12.39	518x294	24.42	**42.48**	18.35
DA3 [33]	ICLR'26	518x168	26.37	28.13	15.76	518x294	**51.25**	23.64	19.30
$\text{OccAny}_{\text{base}}$	–	512x160	**43.38**	20.37	16.09	512x288	48.09	20.97	17.10
OccAny	–	512x160	36.79	46.70	**25.91**	512x288	36.09	40.39	**23.55**

*: use TTVA †: scaled with Metric3Dv2 [19].
$\text{OccAny}_{\text{base}}$: w/o Segmentation Forcing & Novel-view Rendering.

Table 1. **Sequence setting.** Occupancy prediction on SemanticKITTI and Occ3D-NuScenes.

(IoU) to assess geometry quality; mean IoU (mIoU) is used for semantic segmentation. Following open-vocabulary LiDAR semantic segmentation works [15, 41, 46], we also report performance on super classes, denoted as mIoU^{sc}. This helps evaluate results at a coarser semantic level, alleviating the impact of "prompting and text-to-image alignment" limitations [41] especially on semantically confusing classes, *e.g.*, "car" *vs*. "other-vehicle".

4.1. Main results

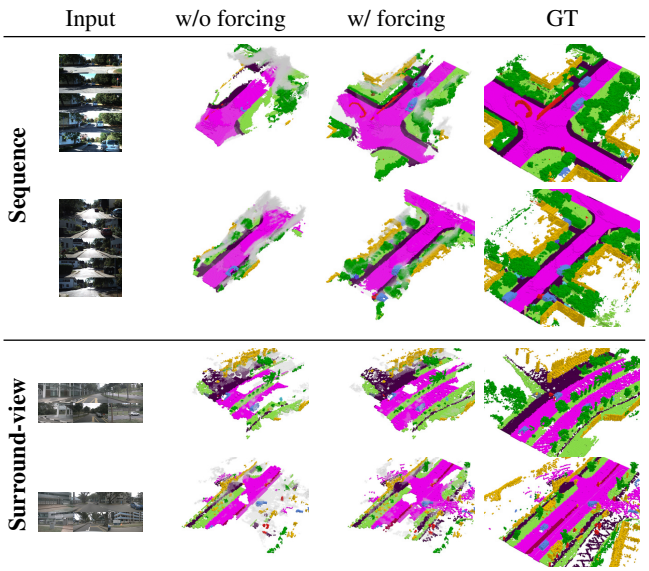
Sequence. In the *Sequence* setting (Tab. 1), OccAny surpasses all other zero-shot baselines. On SemanticKITTI, it reaches 25.91% IoU, surpassing the nearest baseline

Test	Method	Venue	Res.	Prec.	Rec.	IoU
in-domain	MonoScene [5]	CVPR'22	1220x370	13.15	40.22	11.18
	SceneRF [6]	ICCV'23	1220x370	17.28	40.96	13.84
	SelfOcc [21]	CVPR'24	1220x370	34.83	37.31	21.97
	Splatter Image [52]	CVPR'24	1220x370	11.30	53.93	10.30
	Hi-Gaussian [67]	ICCV'25	1220x370	17.39	59.72	15.56
	OccNeRF [73]	TIP'25	1220x370	35.25	39.27	22.81
out-of-domain	MUSt3R [3]	CVPR'25	512x160	15.29	12.24	7.29
	CUT3R [58]	CVPR'25	512x160	33.32	8.64	7.37
	CUT3R* [58]	CVPR'25	512x160	33.47	17.59	13.03
	VGGT† [56]	CVPR'25	518x168	25.59	14.49	10.19
	AnySplat† [26]	TOG'25	518x168	17.97	20.39	10.56
	AnySplat*† [26]	TOG'25	518x168	14.61	**35.21**	11.52
	DA3 [33]	ICLR'26	518x168	23.98	14.54	9.95
	$\text{OccAny}_{\text{base}}$	–	512x160	41.24	14.49	12.01
	OccAny	–	512x160	**45.64**	33.66	**24.03**

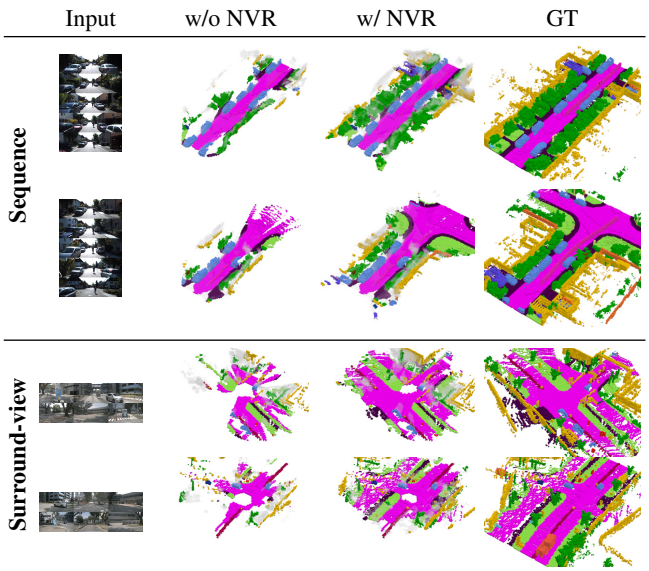
*: use TTVA †: scaled with Metric3Dv2 [19].
$\text{OccAny}_{\text{base}}$: w/o Segmentation Forcing & Novel-view Rendering.

Table 2. **Monocular setting.** Occupancy results with Monocular input on SemanticKITTI following [6, 21]. Results for MonoScene and Splatter Image are taken from [6, 67].

(CUT3R*) by roughly 10 points. A similar trend is observed on Occ3D-NuScenes, where OccAny achieves 23.55% IoU, significantly outperforming baselines; of note, some baselines are already enhanced with post-hoc metric scaling and, if applicable, TTVA. This demonstrates OccAny's ability to effectively complete geometry from limited-view sequence without in-domain training, thanks to *Segmentation Forcing*



(a) Segmentation Forcing



(b) Novel-View Rendering

Figure 5. **Qualitative ablation** shows the gains from *Segmentation Forcing* and *Novel-View Rendering*. Voxel colorization follows Fig. 4. The two proposed strategies significantly improve the density and the accuracy of occupancy predictions.

Test	Method	Venue	Res.	Prec.	Rec.	IoU
in-domain	SelfOcc [21]	CVPR'24	800x450	–	–	45.01
	OccNeRF [73]	TIP'25	672x336	57.20	55.47	39.20
	DistillNeRF [57]	NeuRIPS'24	400x228	–	–	29.11
	SimpleOcc [13]	TIV'24	672x336	41.91	64.02	33.92
	GaussTR [25]	CVPR'25	896x504	–	–	45.19
out-of-domain	MUSt3R [3]	CVPR'25	512x288	20.79	28.29	13.61
	CUT3R [58]	CVPR'25	512x288	32.19	7.93	6.79
	CUT3R* [58]	CVPR'25	512x288	40.60	26.73	19.21
	VGGT† [56]	CVPR'25	518x294	41.56	28.64	20.42
	AnySplat† [26]	TOG'25	518x294	29.35	40.80	20.59
	AnySplat*† [26]	TOG'25	518x294	24.52	57.65	20.78
	DA3 [33]	ICLR'26	518x294	53.26	23.75	19.65
	$\text{OccAny}_{\text{base}}$	–	512x288	**59.58**	21.19	18.53
	OccAny	–	512x288	45.04	**58.54**	**34.15**

*: use TTVA †: scaled with Metric3Dv2 [19].

$\text{OccAny}_{\text{base}}$: w/o Segmentation Forcing & Novel-view Rendering.

Table 3. **Surround-view setting**. More results are in Tab.7 (Sec. B.2)

Method	Venue	Semantic KITTI sequence			Occ3D-NuScenes surround-view		
		Res.	mIoU	mIoU^{sc}	Res.	mIoU	mIoU^{sc}
MUSt3R [3] + SAM2 [43]	CVPR'25	512x160	3.22	5.96	512x288	2.43	3.84
CUT3R [58] + SAM2 [43]	CVPR'25	512x160	4.15	6.72	512x288	2.40	2.75
CUT3R* [58] + SAM2 [43]	CVPR'25	512x160	4.53	8.18	512x288	3.06	3.99
VGGT† [56] + SAM2 [43]	CVPR'25	518x294	3.47	6.76	518x294	4.39	6.49
AnySplat† [26]+ SAM2 [43]	TOG'25	518x168	3.37	6.83	518x294	3.96	5.97
AnySplat*† [26]+ SAM2 [43]	TOG'25	518x168	3.86	7.51	518x294	4.44	6.51
DA3 [33]+ SAM2 [43]	ICLR'26	518x168	4.92	9.56	518x294	4.55	6.29
OccAny w/o forcing + SAM2 [43]	–	512x160	6.83	12.01	512x288	6.17	8.96
OccAny	–	512x160	**7.28**	**13.53**	512x288	**6.66**	**10.32**

*: use TTVA †: scaled with Metric3Dv2 [19].

Table 4. **Semantic Occupancy Prediction** with GSAM2 [44].

and *Novel-View Rendering*. The $\text{OccAny}_{\text{base}}$ variant, which is equivalent to fine-tuning MUSt3R on our datasets, was trained without the two proposed strategies and obtained only marginal improvements over baselines.

Wrong metric reasoning leads to voxels predicted outside of the scene, significantly degrading the performance. The scale-invariant design of VGGT and AnySplat is not well-suited for the occupancy task, unlike OccAny with metric prediction by design. The Gaussian Splatting of AnySplat, while favorable for synthesizing compelling images, produces lots of geometric artifacts, thereby hallucinating lots of noises and harming geometry prediction. Fig. 4 visualizes the occupancy results.

Monocular. In the more challenging *Monocular* setting on SemanticKITTI (Tab. 2), OccAny demonstrates remarkable generalization. It achieves 24.03% IoU, outperforming all other zero-shot baselines by significant margins (*e.g.*, +11.00% IoU over CUT3R* w/ TTVA). Notably, it significantly surpasses several in-domain self-supervised methods like SceneRF (+10.19%); OccAny even surpasses self-supervised SOTAs SelfOcc (+2.06%) and OccNeRF (+1.22%), despite never been trained on SemanticKITTI.

Surround-view. In the *Surround-view* setting on Occ3D-NuScenes Tab. 3, OccAny maintains its lead among zero-shot methods with 34.15% IoU, and achieves better performance than some in-domain approaches like DistillNeRF/SimpleOcc, yet remains behind more recent methods.

Semantic Occupancy. We further evaluate 3D semantic occupancy (Tab. 4) by applying Grounded SAM2 pipeline directly on OccAny's segmentation features. OccAny achieves the highest mIoU and mIoU^{sc} across both datasets, compared to baselines using a separated SAM2 model to produce segmentation features. The comparison with the variant "OccAny w/o forcing + SAM2" confirms that our *Segmentation Forcing* strategy leads to a unified and simpler solution to

Method	Semantic KITTI sequence						Occ3D-NuScenes surround-view					
	Res.	Pre.	Rec.	IoU	mIoU	mIoU^{sc}	Res.	Pre.	Rec.	IoU	mIoU	mIoU^{sc}
OccAny	512x160	36.79	46.70	25.91	**7.28**	**13.53**	512x288	**45.04**	**58.54**	**34.15**	6.66	10.32
OccAny+	512x160	**38.12**	**49.14**	**27.33**	6.48	13.30	512x288	46.38	54.66	33.49	**7.20**	**11.50**

Table 5. **Changing the base foundation models** used in OccAny to DA3 [33] and SAM3 [8] results in the **OccAny+** variant.

NVR	$\mathcal{L}_{\text{forcing}}$	geo-aware	$t_g + t_s$	$\mathcal{L}_{\text{enc}}$	SemKITTI seq. IoU	SemKITTI seq. Δ IoU	SemKITTI single IoU	SemKITTI single Δ IoU
✗					19.64	-6.27	11.55	-12.48
	✗				24.23	-1.68	21.73	-2.30
		✗			24.88	-1.03	23.02	-1.01
			✗		25.04	-0.87	22.67	-1.36
				✗	24.99	-0.92	23.46	-0.57
OccAny					25.91	—	24.03	—

Table 6. **Ablation results** on SemanticKitti. The "geo-aware" stands for applying geometry confidence maps $\mathbf{C}$ in the segmentation forcing loss (*cf*. Eq. (3)).

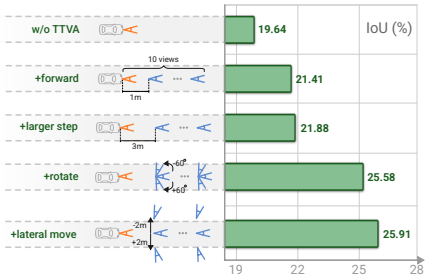


Figure 6. **Ablating NVR inference** on SemanticKITTI

better predict geometry and segmentation.

Impact of base foundation models. We change the foundation models used in OccAny to DA3 [33] and SAM3 [8], resulting in the **OccAny+** variant, detailed in Sec. A.3. Tab. 5 and Sec. B show that OccAny benefits from advances in generic foundation models, while being *independently and orthogonally* effective for occupancy prediction.

4.2. Analysis

Method ablation. Tab. 6 analyzes the contribution of each proposed component. Removing Test-Time View Augmentation (TTVA) causes the most significant drop (-6.27% in sequence- and -12.47% in monocular setting), highlighting its critical role in geometry completion. The rendering-specific losses $\mathcal{L}_{\text{Enc}}$, geometry-aware $\mathcal{L}_{\text{forcing}}$, and the task tokens also consistently contribute to the final performance, proving their effectiveness. Fig. 5 shows gains brought by Segmentation Forcing and Novel-view Rendering (TTVA).

NVR inference. We ablate NVR inference in Fig. 6. Starting from the baseline without TTVA, adding simple forward movement helps complete distant geometry ($+1.83\%$). Introducing rotations and lateral shifts further helps complete the geometry by resolving occlusions from diverse views, improving IoU by $+4.15\%$ and resulting in the final 25.91%.

Promptable segmentation feature. We visualize the segmentation features of OccAny using PCA, as shown in Fig. 7. Low-resolution features appear to cluster semantically similar regions, while high-resolution features seem to capture fine details like boundaries and textures, both helping regularize and improve occupancy prediction (*cf*. Fig. 5 & Tab. 6).

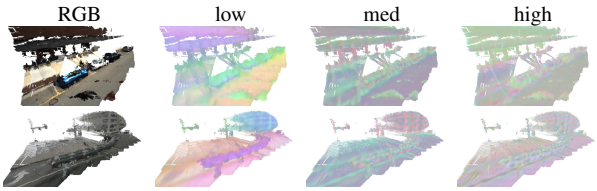


Figure 7. **PCA visualization of our segmentation features** of multi-view sequences. Low-resolution features capture high-level semantics (*e.g.*, separating cars, buildings, and roads), while high-resolution features capture low-level details such as boundaries and textures. Features remain consistent across different views.

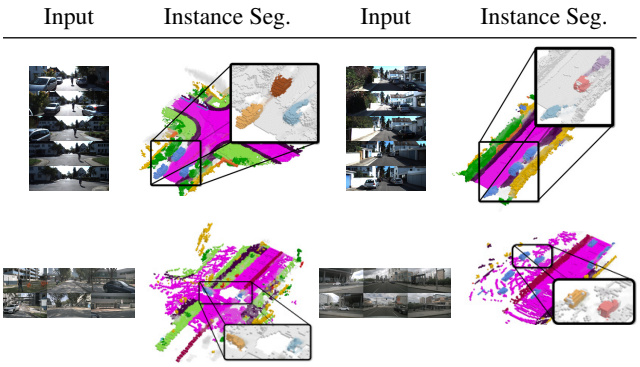


Figure 8. **Instance segmentation** of cars with OccAny's features.

Similar to SAM2, our segmentation features remain spatially and temporally consistent. This consistency enables instance segmentation via prompting with object instances detected by Grounding DINO. In Fig. 8, we show some qualitative results when performing instance segmentation directly on our segmentation features.

5. Conclusion

We propose for the first time a generalized 3D occupancy network, called OccAny, that is trained once and perform zero-shot inference on arbitrary out-of-domain sequential, monocular and surround-view unposed data. With the proposed Segmentation Forcing and Novel-View Rendering strategies, OccAny outperforms generic visual-geometry foundation models on occupancy prediction. OccAny surpasses several in-domain self-supervised models, while remaining behind more recent ones. Our work introduces a novel framework for occupancy prediction prioritizing scalability and generalization, paving the way toward the next generation of versatile and generalized occupancy networks. The gap to fully-supervised in-domain performance remains substantial, leaving room for future improvements in this direction.

Acknowledgment. This work was granted access to the HPC resources of IDRIS under the allocations AD011014102R2, AD011013540R1 made by GENCI. We acknowledge EuroHPC Joint Undertaking for awarding the project ID EHPC-REG-2025R01-032 access to Karolina, Czech Republic.

References

[1] Jens Behley, Martin Garbade, Andres Milioto, Jan Quenzel, Sven Behnke, Cyrill Stachniss, and Juergen Gall. Semantickitti: A dataset for semantic scene understanding of lidar sequences. In *ICCV*, 2019. 2, 5

[2] Yohann Cabon, Naila Murray, and Martin Humenberger. Virtual kitti 2. In *arXiv*, 2020. 5

[3] Yohann Cabon, Lucas Stoffl, Leonid Antsfeld, Gabriela Csurka, Boris Chidlovskii, Jerome Revaud, and Vincent Leroy. Must3r: Multi-view network for stereo 3d reconstruction. In *CVPR*, 2025. 2, 3, 5, 6, 7

[4] Holger Caesar, Varun Bankiti, Alex H. Lang, Sourabh Vora, Venice Erin Liong, Qiang Xu, Anush Krishnan, Yu Pan, Giancarlo Baldan, and Oscar Beijbom. nuscenes: A multimodal dataset for autonomous driving. In *CVPR*, 2020. 1

[5] Anh-Quan Cao and Raoul de Charette. Monoscene: Monocular 3d semantic scene completion. In *CVPR*, 2022. 1, 2, 5, 6

[6] Anh-Quan Cao and Raoul de Charette. Scenerf: Self-supervised monocular 3d scene reconstruction with radiance fields. In *ICCV*, 2023. 1, 2, 6

[7] Anh-Quan Cao, Angela Dai, and Raoul de Charette. Pasco: Urban 3d panoptic scene completion with uncertainty awareness. In *CVPR*, 2024. 2

[8] Nicolas Carion, Laura Gustafson, Yuan-Ting Hu, Shoubhik Debnath, Ronghang Hu, Didac Suris, Chaitanya Ryali, Kalyan Vasudev Alwala, Haitham Khedr, Andrew Huang, Jie Lei, Tengyu Ma, Baishan Guo, Arpit Kalla, Markus Marks, Joseph Greer, Meng Wang, Peize Sun, Roman Rädle, Triantafyllos Afouras, Effrosyni Mavroudi, Katherine Xu, Tsung-Han Wu, Yu Zhou, Liliane Momeni, Rishi Hazra, Shuangrui Ding, Sagar Vaze, Francois Porcher, Feng Li, Siyuan Li, Aishwarya Kamath, Ho Kei Cheng, Piotr Dollár, Nikhila Ravi, Kate Saenko, Pengchuan Zhang, and Christoph Feichtenhofer. Sam 3: Segment anything with concepts. In *ICLR*, 2026. 8

[9] Loick Chambon, Eloi Zablocki, Alexandre Boulch, Mickael Chen, and Matthieu Cord. Gaussrender: Learning 3d occupancy with gaussian rendering. In *CVPR*, 2025. 2

[10] Angel X Chang, Thomas Funkhouser, Leonidas Guibas, Pat Hanrahan, Qixing Huang, Zimo Li, Silvio Savarese, Manolis Savva, Shuran Song, Hao Su, et al. Shapenet: An information-rich 3d model repository. *arXiv*, 2015. 1

[11] Christopher Choy, JunYoung Gwak, and Silvio Savarese. 4d spatio-temporal convnets: Minkowski convolutional neural networks. In *CVPR*, 2019. 1

[12] Angela Dai, Angel X Chang, Manolis Savva, Maciej Halber, Thomas Funkhouser, and Matthias Nießner. Scannet: Richly-annotated 3d reconstructions of indoor scenes. In *CVPR*, 2017. 1

[13] Wanshui Gan, Ningkai Mo, Hongbin Xu, and Naoto Yokoya. A comprehensive framework for 3d occupancy estimation in autonomous driving. *IEEE TIV*, 2024. 7

[14] Wanshui Gan, Fang Liu, Hongbin Xu, Ningkai Mo, and Naoto Yokoya. Gaussianocc: Fully self-supervised and efficient 3d occupancy estimation with gaussian splatting. In *ICCV*, 2025. 1, 2

[15] Simon Gebraad, Andras Palffy, and Holger Caesar. Leap: Consistent multi-domain 3d labeling using foundation models. In *ICRA*, 2025. 6

[16] Andreas Geiger, Philip Lenz, and Raquel Urtasun. Are we ready for autonomous driving? the kitti vision benchmark suite. In *CVPR*, 2012. 1

[17] Vitor Guizilini, Rares Ambrus, Sudeep Pillai, Allan Raventos, and Adrien Gaidon. 3d packing for self-supervised monocular depth estimation. In *CVPR*, 2020. 5

[18] Adrian Hayler, Felix Wimbauer, Dominik Muhle, Christian Rupprecht, and Daniel Cremers. S4c: Self-supervised semantic scene completion with neural fields. In *3DV*, 2024. 2

[19] Mu Hu, Wei Yin, Chi Zhang, Zhipeng Cai, Xiaoxiao Long, Hao Chen, Kaixuan Wang, Gang Yu, Chunhua Shen, and Shaojie Shen. Metric3d v2: A versatile monocular geometric foundation model for zero-shot metric depth and surface normal estimation. *IEEE TPAMI*, 2024. 5, 6, 7

[20] Yuanhui Huang, Wenzhao Zheng, Yunpeng Zhang, Jie Zhou, and Jiwen Lu. Tri-perspective view for vision-based 3d semantic occupancy prediction. In *CVPR*, 2023. 1, 2

[21] Yuanhui Huang, Wenzhao Zheng, Borui Zhang, Jie Zhou, and Jiwen Lu. Selfocc: Self-supervised vision-based 3d occupancy prediction. In *CVPR*, 2024. 1, 2, 5, 6, 7

[22] Yuanhui Huang, Wenzhao Zheng, Yunpeng Zhang, Jie Zhou, and Jiwen Lu. Gaussianformer: Scene as gaussians for vision-based 3d semantic occupancy prediction. In *ECCV*, 2024. 2

[23] Wonbong Jang, Philippe Weinzaepfel, Vincent Leroy, Lourdes Agapito, and Jerome Revaud. Pow3r: Empowering unconstrained 3d reconstruction with camera and scene priors. In *CVPR*, 2025. 2

[24] Aleksandar Jevtić, Christoph Reich, Felix Wimbauer, Oliver Hahn, Christian Rupprecht, Stefan Roth, and Daniel Cremers. Feed-forward scenedino for unsupervised semantic scene completion. In *ECCV*, 2025. 1, 2

[25] 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 *CVPR*, 2025. 1, 2, 7

[26] Lihan Jiang, Yucheng Mao, Linning Xu, Tao Lu, Kerui Ren, Yichen Jin, Xudong Xu, Mulin Yu, Jiangmiao Pang, Feng Zhao, Dahua Lin, and Bo Dai. Anysplat: Feed-forward 3d gaussian splatting from unconstrained views. *ACM TOG*, 2025. 2, 5, 6, 7

[27] Bernhard Kerbl, Georgios Kopanas, Thomas Leimkühler, and George Drettakis. 3d gaussian splatting for real-time radiance field rendering. *ACM TOG*, 2023. 1, 5

[28] Vincent Leroy, Yohann Cabon, and Jerome Revaud. Grounding image matching in 3d with mast3r. In *ECCV*, 2024. 2

[29] Bohan Li, Yasheng Sun, Xin Jin, Wenjun Zeng, Zheng Zhu, Xiaoefeng Wang, Yunpeng Zhang, James Okae, Hang Xiao, and Dalong Du. Stereoscene: Bev-assisted stereo matching empowers 3d semantic scene completion. In *IJCAI*, 2024. 1

[30] Samuel Li, Pujith Kachana, Prajwal Chidananda, Saurabh Nair, Yasutaka Furukawa, and Matthew Brown. Rig3r: Rig-aware conditioning for learned 3d reconstruction. In *NeurIPS*, 2025. 2

[31] Yiming Li, Zhiding Yu, Christopher Choy, Chaowei Xiao, Jose M Alvarez, Sanja Fidler, Chen Feng, and Anima Anandkumar. Voxformer: Sparse voxel transformer for camera-based 3d semantic scene completion. In *CVPR*, 2023. 1, 2

[32] Yiming Li, Sihang Li, Xinhao Liu, Moonjun Gong, Kenan Li, Nuo Chen, Zijun Wang, Zhiheng Li, Tao Jiang, Fisher Yu, Yue Wang, Hang Zhao, Zhiding Yu, and Chen Feng. Sscbench: A large-scale 3d semantic scene completion benchmark for autonomous driving. In *IROS*, 2024. 2

[33] Haotong Lin, Sili Chen, Jun Hao Liew, Donny Y. Chen, Zhenyu Li, Guang Shi, Jiashi Feng, and Bingyi Kang. Depth anything 3: Recovering the visual space from any views. *arXiv*, 2025. 5, 6, 7, 8

[34] Haisong Liu, Haiguang Wang, Yang Chen, Zetong Yang, Jia Zeng, Li Chen, and Limin Wang. Fully sparse 3d panoptic occupancy prediction. In *ECCV*, 2024. 2

[35] Shilong Liu, Zhaoyang Zeng, Tianhe Ren, Feng Li, Hao Zhang, Jie Yang, Qing Jiang, Chunyuan Li, Jianwei Yang, Hang Su, et al. Grounding dino: Marrying dino with grounded pre-training for open-set object detection. In *ECCV*, 2024. 4

[36] 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 *CVPR*, 2024. 2

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

[38] Jiageng Mao, Minzhe Niu, Chenhan Jiang, Hanxue Liang, Jingheng Chen, Xiaodan Liang, Yamin Li, Chaoqiang Ye, Wei Zhang, Zhenguo Li, et al. One million scenes for autonomous driving: Once dataset. In *NeurIPS*, 2021. 5

[39] R. Marcuzzi, L. Nunes, E.A. Marks, L. Wiesmann, T. Läbe, J. Behley, and C. Stachniss. SfmOcc: Vision-Based 3D Semantic Occupancy Prediction in Urban Environments. *RA-L*, 2025. 2

[40] 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. *Commun. ACM*, 2021. 1

[41] Aljoša Ošep, Tim Meinhardt, Francesco Ferroni, Neehar Peri, Deva Ramanan, and Laura Leal-Taixé. Better call sal: Towards learning to segment anything in lidar. In *ECCV*, 2024. 6

[42] Charles R Qi, Hao Su, Kaichun Mo, and Leonidas J Guibas. Pointnet: Deep learning on point sets for 3d classification and segmentation. In *CVPR*, 2017. 1

[43] Nikhila Ravi, Valentin Gabeur, Yuan-Ting Hu, Ronghang Hu, Chaitanya Ryali, Tengyu Ma, Haitham Khedr, Roman Rädle, Chloe Rolland, Laura Gustafson, Eric Mintun, Junting Pan, Kalyan Vasudev Alwala, Nicolas Carion, Chao-Yuan Wu, Ross Girshick, Piotr Dollár, and Christoph Feichtenhofer. Sam 2: Segment anything in images and videos. In *ICLR*, 2025. 3, 7

[44] Tianhe Ren, Shilong Liu, Ailing Zeng, Jing Lin, Kunchang Li, He Cao, Jiayu Chen, Xinyu Huang, Yukang Chen, Feng Yan, et al. Grounded sam: Assembling open-world models for diverse visual tasks. In *arXiv*, 2024. 4, 7

[45] Luis Roldão, Raoul de Charette, and Anne Verroust-Blondet. Lmscnet: Lightweight multiscale 3d semantic completion. In *3DV*, 2020. 2

[46] Nermin Samet, Gilles Puy, and Renaud Marlet. Losc: Lidar open-voc segmentation consolidator. In *3DV*, 2026. 6

[47] Yiang Shi, Tianheng Cheng, Qian Zhang, Wenyu Liu, and Xinggang Wang. Occupancy as set of points. In *ECCV*, 2024. 2

[48] Sophia Sirko-Galouchenko, Alexandre Boulch, Spyros Gidaris, Andrei Bursuc, Antonin Vobecky, Patrick Pérez, and Renaud Marlet. Occfeat: Self-supervised occupancy feature prediction for pretraining bev segmentation networks. In *CVPR*, 2024. 2

[49] Shuran Song, Fisher Yu, Andy Zeng, Angel X Chang, Manolis Savva, and Thomas Funkhouser. Semantic scene completion from a single depth image. In *CVPR*, pages 1746–1754, 2017. 2

[50] Edgar Sucar, Zihang Lai, Eldar Insafutdinov, and Andrea Vedaldi. Dynamic point maps: A versatile representation for dynamic 3d reconstruction. In *ICCV*, 2025. 2

[51] Pei Sun, Henrik Kretzschmar, Xerxes Dotiwalla, Aurelien Chouard, Vijaysai Patnaik, Paul Tsui, James Guo, Yin Zhou, Yuning Chai, Benjamin Caine, et al. Scalability in perception for autonomous driving: Waymo open dataset. In *CVPR*, 2020. 5

[52] Stanislaw Szymanowicz, Chrisitian Rupprecht, and Andrea Vedaldi. Splatter image: Ultra-fast single-view 3d reconstruction. In *CVPR*, 2024. 6

[53] Hugues Thomas, Charles R Qi, Jean-Emmanuel Deschaud, Beatriz Marcotegui, François Goulette, and Leonidas J Guibas. Kpconv: Flexible and deformable convolution for point clouds. In *ICCV*, 2019. 1

[54] 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. In *NeurIPS*, 2023. 2, 5

[55] Hengyi Wang and Lourdes Agapito. 3d reconstruction with spatial memory. In *3DV*, 2025. 2

[56] Jianyuan Wang, Minghao Chen, Nikita Karaev, Andrea Vedaldi, Christian Rupprecht, and David Novotny. Vggt: Visual geometry grounded transformer. In *CVPR*, 2025. 2, 5, 6, 7

[57] Letian Wang, Seung Wook Kim, Jiawei Yang, Cunjun Yu, Boris Ivanovic, Steven L. Waslander, Yue Wang, Sanja Fidler, Marco Pavone, and Peter Karkus. Distillnerf: Perceiving 3d scenes from single-glance images by distilling neural fields and foundation model features. In *NeurIPS*, 2024. 2, 7

[58] Qianqian Wang, Yifei Zhang, Aleksander Holynski, Alexei A Efros, and Angjoo Kanazawa. Continuous 3d perception model with persistent state. In *CVPR*, 2025. 2, 5, 6, 7

[59] Shuzhe Wang, Vincent Leroy, Yohann Cabon, Boris Chidlovskii, and Jerome Revaud. Dust3r: Geometric 3d vision made easy. In *CVPR*, 2024. 2

[60] Yu Wang and Chao Tong. H2gformer: Horizontal-to-global voxel transformer for 3d semantic scene completion. In *AAAI*, 2024. 1

[61] Yi Wei, Linqing Zhao, Wenzhao Zheng, Zheng Zhu, Jie Zhou, and Jiwen Lu. Surroundocc: Multi-camera 3d occupancy prediction for autonomous driving. In *ICCV*, 2023. 1

[62] Felix Wimbauer, Nan Yang, Christian Rupprecht, and Daniel Cremers. Behind the scenes: Density fields for single view reconstruction. In *CVPR*, 2023. 1, 2

[63] Felix Wimbauer, Weirong Chen, Dominik Muhle, Christian Rupprecht, and Daniel Cremers. Anycam: Learning to recover camera poses and intrinsics from casual videos. In *CVPR*, 2025. 2

[64] Xiaoyang Wu, Li Jiang, Peng-Shuai Wang, Zhijian Liu, Xihui Liu, Yu Qiao, Wanli Ouyang, Tong He, and Hengshuang Zhao. Point transformer v3: Simpler faster stronger. In *CVPR*, 2024. 1

[65] Zhaoyang Xia, Youquan Liu, Xin Li, Xinge Zhu, Yuexin Ma, Yikang Li, Yuenan Hou, and Yu Qiao. Scpnet: Semantic scene completion on point cloud. In *CVPR*, 2023. 2

[66] Pengchuan Xiao, Zhenlei Shao, Steven Hao, Zishuo Zhang, Xiaolin Chai, Judy Jiao, Zesong Li, Jian Wu, Kai Sun, Kun Jiang, et al. Pandaset: Advanced sensor suite dataset for autonomous driving. In *ITSC*, 2021. 5

[67] Binjian Xie, Pengju Zhang, Hao Wei, and Yihong Wu. Higaussian: Hierarchical gaussians under normalized spherical projection for single-view 3d reconstruction. In *ICCV*, 2025. 2, 6

[68] Yujie Xue, Huilong Pi, Jiapeng Zhang, Yunchuan Qin, Zhuo Tang, Kenli Li, and Ruihui Li. Sdformer: Vision-based 3d semantic scene completion via sam-assisted dual-channel voxel transformer. In *ICCV*, 2025. 1

[69] Jianing Yang, Alexander Sax, Kevin J. Liang, Mikael Henaff, Hao Tang, Ang Cao, Joyce Chai, Franziska Meier, and Matt Feiszli. Fast3r: Towards 3d reconstruction of 1000+ images in one forward pass. In *CVPR*, 2025. 2

[70] Jiawei Yao, Chuming Li, Keqiang Sun, Yingjie Cai, Hao Li, Wanli Ouyang, and Hongsheng Li. Ndc-scene: Boost monocular 3d semantic scene completion in normalized device coordinates space. In *ICCV*, 2023. 1, 2

[71] Baijun Ye, Minghui Qin, Saining Zhang, Moonjun Gong, Shaoting Zhu, Hao Zhao, and Hang Zhao. Gs-occ3d: Scaling vision-only occupancy reconstruction with gaussian splatting. In *ICCV*, 2025. 2

[72] Zhu Yu, Runmin Zhang, Jiacheng Ying, Junchen Yu, Xiaohai Hu, Lun Luo, Si-Yuan Cao, and Hui-liang Shen. Context and geometry aware voxel transformer for semantic scene completion. In *NeurIPS*, 2024. 1

[73] Chubin Zhang, Juncheng Yan, Yi Wei, Jiaxin Li, Li Liu, Yansong Tang, Yueqi Duan, and Jiwen Lu. Occnerf: Advancing 3d occupancy prediction in lidar-free environments. *IEEE TIP*, 2025. 2, 6, 7

[74] Junyi Zhang, Charles Herrmann, Junhwa Hur, Varun Jampani, Trevor Darrell, Forrester Cole, Deqing Sun, and Ming-Hsuan Yang. Monst3r: A simple approach for estimating geometry in the presence of motion. In *ICLR*, 2025. 2

[75] Yunpeng Zhang, Zheng Zhu, and Dalong Du. Occformer: Dual-path transformer for vision-based 3d semantic occupancy prediction. In *ICCV*, 2023. 2

[76] Xiaoyu Zhou, Jingqi Wang, Yongtao Wang, Yufei Wei, Nan Dong, and Ming-Hsuan Yang. Autoocc: Automatic open-ended semantic occupancy annotation via vision-language guided gaussian splatting. In *ICCV*, 2025. 2

[77] Sicheng Zuo, Wenzhao Zheng, Xiaoyong Han, Longchao Yang, Yong Pan, and Jiwen Lu. Quadricformer: Scene as superquadrics for 3d semantic occupancy prediction. *NeurIPS*, 2025. 2

[78] Lojze Zust, Yohann Cabon, Juliette Marrie, Leonid Antsfeld, Boris Chidlovskii, Jerome Revaud, and Gabriela Csurka. Panst3r: Multi-view consistent panoptic segmentation. In *ICCV*, 2025. 2