# DRIVETOK: 3D DRIVING SCENE TOKENIZATION FOR UNIFIED MULTI-VIEW RECONSTRUCTION AND UNDERSTANDING

**Anonymous authors**
Paper under double-blind review

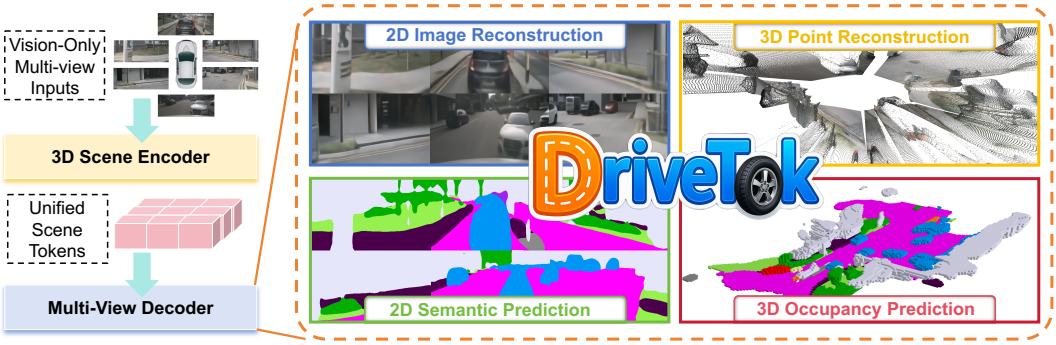



Figure 1: **Overview of DriveTok.** Multi-view images are processed by a 3D scene encoder to produce unified scene tokens, independent of camera layout and resolution. A spatial-aware multi-view decoder renders predictions in both image and occupancy spaces. Through joint multi-task training, our scene tokens encode rich textural, semantic, and geometric information.

## ABSTRACT

Visual tokenization for autonomous driving must represent multi-view scenes compactly while preserving the information needed for reconstruction and scene understanding. However, most existing tokenizers are designed for monocular and 2D scenes, leading to inefficiency and inter-view inconsistency when applied to high-resolution multi-view driving scenes. To address this, we propose DriveTok, an efficient 3D driving scene tokenizer for unified multi-view reconstruction and understanding. DriveTok first obtains semantically rich visual features from vision foundation models and then transforms them into the scene tokens with 3D deformable cross-attention. For decoding, we employ a multi-view transformer to reconstruct multi-view features from the scene tokens and use multiple heads to obtain RGB, depth, and semantic reconstructions. We also add a 3D head directly on the scene tokens for 3D semantic occupancy prediction for better spatial awareness. With the multiple training objectives, DriveTok learns unified scene tokens that integrate semantic, geometric, and textural information for efficient multi-view tokenization. Extensive experiments demonstrate that the scene tokens from DriveTok perform well on image reconstruction, semantic segmentation, depth prediction, and 3D occupancy prediction tasks.

## 1 INTRODUCTION

Autonomous driving is undergoing a shift from a perception-centric pipeline toward a reasoning-based pipeline for complex driving scenes (Mao et al., 2023; Shao et al., 2024; Xu et al., 2024). This transition is largely driven by vision-language-action models (VLAs) (Wang et al., 2002; Zhou et al., 2025a; Jiang et al., 2025; Zhou et al., 2025b; Li et al., 2025b) and world models (Hu et al., 2023a; Wang et al., 2024a; Zhao et al., 2025; Wang et al., 2024b; Zheng et al., 2024b; Li et al., 2025b), which are increasingly applied to enhance the cognitive capabilities of autonomous systems.

A central challenge in this evolution is how to represent the sensor inputs with comprehensive (i.e., low-level information for reconstruction) yet also semantically rich (i.e., high-level information for



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.

understanding) representations. This motivates a visual representation that preserves both image detail and 3D scene structure. However, most existing visual tokenizers focus on monocular and generic-domain image reconstruction and usually tokenize inputs into per-image 2D patch tokens. They cannot capture the spatial 3D information that is vital for driving. Also, autonomous vehicles are typically equipped with surrounding high-resolution cameras. Per-image tokenization leads to a large set of tokens and thus inefficiency for the subsequent large models.

To address these challenges, we propose DriveTok, an efficient 3D scene tokenizer tailored for autonomous driving systems, as shown in Figure 1. DriveTok utilizes a unified scene representation to transform inputs from multiple high-resolution cameras into a fixed number of geometry-aware scene tokens. Semantic-rich visual features are first extracted using a pre-trained vision foundation model, then mapped to 3D space through 3D deformable cross-attention. For decoding, we adopt a spatial-aware multi-view transformer to combine the encoded scene tokens with camera-specific view tokens and uses a visibility mask to restrict scene–view attention to regions observed by each camera. The updated view tokens are decoded into RGB, depth, and semantic maps, while a separate 3D head predicts semantic occupancy directly from the scene tokens. These multi-task objectives enable DriveTok to learn unified scene tokens that encode semantic, geometric, and textural information. Extensive experiments validate the effectiveness of these tokens across image reconstruction, semantic segmentation, depth prediction, and 3D occupancy prediction tasks.

## 2 Related Work

**Visual Tokenizers.** Discrete tokenizers such as VQ-VAE (Van Den Oord et al., 2017) and VQ-GAN (Esser et al., 2021) compress images into learned codebook indices, while transformer variants improve scalability and sequence modeling (Yu et al., 2021; 2024). Continuous tokenizers instead optimize latent features for reconstruction, generation, or alignment with foundation models (Wu et al., 2025; Tong et al., 2025; Yao et al., 2025; Sargent et al., 2025). These methods are typically trained on generic and single-image data. When applied independently to every driving camera, they duplicate overlapping content and provide neither a shared 3D frame nor guaranteed cross-view consistency. DriveTok retains the useful reconstruction objective of image tokenizers but moves the latent representation into a unified metric scene space.

**Driving-scene Representations.** Depth estimators recover local structure from images (Zhou et al., 2017; Godard et al., 2019; Ranftl et al., 2020; 2021), while BEV methods explicitly lift multi-camera evidence into a common coordinate system (Philion & Fidler, 2020; Huang et al., 2021; Li et al., 2023; 2024). Subsequent methods employ tri-perspective view (Huang et al., 2023), sparse 3D Gaussians (Huang et al., 2024b; 2025), or superquadrics (Zuo et al., 2025b) to better represent 3D structure. Such representations are effective for perception, but most are optimized for a particular output. Recent 3D reconstruction methods also learn geometric representations (Wang et al., 2025a; Lin et al., 2025; Zuo et al., 2025a; 2026), but their view-level representations scale with the observations and are not explicitly optimized as a compact scene-token interface.

**Geometry-aware Driving Tokenizers.** BEV-VAE (Chen & Zhao, 2025) encodes multi-view images into a BEV-aligned latent for spatially consistent generation, and subsequent methods (Ivanovic et al., 2025) use a triplane representation can reduce the number of camera tokens used by driving policies. In contrast, our DriveTok introduces a unified scene tokenization framework designed for multi-view driving scenes, which jointly preserves RGB appearance, metric depth, image semantics, and semantic occupancy within a fixed scene-token sequence. Its joint training over image-space and 3D-space objectives leads to semantic-rich scene tokens that better align with downstream vision-language models for holistic understanding and reasoning in autonomous driving scenes.

## 3 Proposed Approach

### 3.1 3D Driving Scene Tokenization

Conventional image tokenizers typically operate in the 2D image space, compressing individual images $I_i \in \mathbb{R}^{H\times W\times 3}$ into latent or discrete sequences $Z_i \in \mathbb{R}^{L\times d}$ optimized for image-level reconstruction or generation, Where $L$ is the length of sequence and $d$ is the embedding dimension. However, when applied to autonomous driving, this per-image formulation is fundamentally limited.

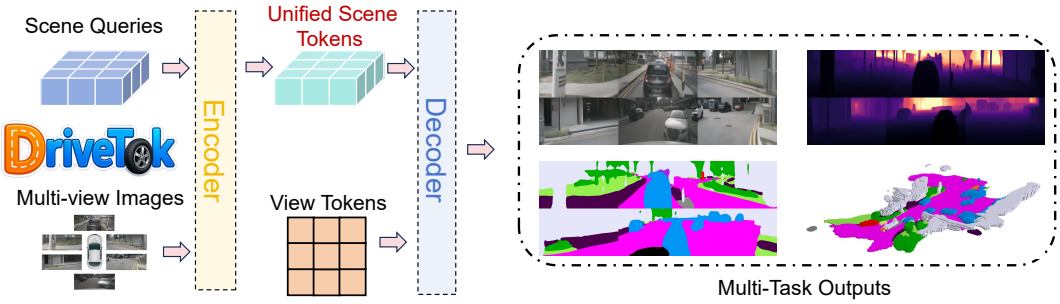



Figure 2: **Framework of DriveTok.** DriveTok processes multi-view images and scene queries through an encoder-decoder architecture to produce unified scene tokens and generate diverse autonomous driving scene reconstruction and understanding outputs.

Multi-camera inputs $\{I_i\}_{i=1}^N$ are processed independently, neglecting their shared 3D structure and resulting in view-inconsistent tokens $\{Z_i\}_{i=1}^N$ that lack spatial alignment and semantic grounding. Moreover, tokenizing each high-resolution image separately incurs computational cost proportional to $\mathcal{O}(N \cdot H \cdot W)$ and redundant representations in overlapping regions, making it inefficient and poorly scalable for dense, multi-view driving scenes.

To address these limitations, we propose a 3D driving scene tokenization framework as shown in Figure 2, which lifts multi-view image features into a unified 3D space to generate compact scene tokens $B$. Image features extracted from a vision foundation model are projected into a global scene grid $\mathcal{Q}$ using geometry-aware sampling, where information from all camera views is aggregated. These scene tokens are spatially consistent, resolution-agnostic, and encode a combination of semantic, geometric, and visual cues, enhanced through our joint multi-task training strategy.

### 3.2 Vision Foundation Model as Scene Encoder

Prior work (Zheng et al., 2025) shows that building an image tokenizer on top of a pretrained vision foundation model leads to stronger semantic representations. Such semantics are useful for scene understanding. Accordingly, we adopt a pretrained vision foundation model as our image encoder to ensure strong generalization and semantic richness in the extracted features.

Specifically, given surround-view images $\{I_i\}_{i=1}^N$ at time $t$, we extract semantically and texturally rich per-camera features using a pretrained vision foundation backbone and an FPN (Lin et al., 2017), producing $F_i = \text{ImageEnc}(I_i) \in \mathbb{R}^{H_f \times W_f \times C}$, which serve as the basis for generating the unified scene representation. We then map these features into a fixed scene grid $\mathcal{Q} = \{q_{ij} | i = 1, ..., W_b, j = 1, ..., H_b\}$ using a scene query lifter followed by BEVFormer-style multi-scale deformable attention (Li et al., 2024), which leverages the intrinsic/extrinsic of each camera to sample informative image regions for every scene query. Concretely, let $X = (x, y, z, 1)^\top$ denote a 3D point in the LiDAR/ego frame associated with $q_{ij}$ and height bin $z$. For camera i with pose $[R_i|t_i]$ and intrinsics $K_i$ , the 3D-to-2D projection is

$$\tilde{\mathbf{u}}_i = K_i \, [\, R_i \mid t_i \,] \, X, \quad u_i = \frac{\tilde{u}_{i,0}}{\tilde{u}_{i,2}}, v_i = \frac{\tilde{u}_{i,1}}{\tilde{u}_{i,2}}. \tag{1}$$

We aggregate features at sub-pixel locations via bilinear sampling $\phi(F_i, u_i, v_i)$. The scene representation for cell $q_{ij}$ is obtained by deformable cross-attention over $K$ learned offsets per camera:

$$\mathbf{b}_{ij} = \sum_{i=1}^{N} \sum_{k=1}^{K} \alpha_{i,k}^{ij} \phi(F_i, u_{i,k}^{ij}, v_{i,k}^{ij}), \tag{2}$$

where $\alpha_{i,k}^{ij}$ are attention weights depending on the content features and geometry.

Because the scene grid size $(H_b, W_b)$ is fixed (e.g., $H_b$=128, $W_b$=128), the number of resulting scene tokens $N_b = H_b \times W_b$ is decoupled from both the number of cameras $N$ and the input resolution $H \times W$, yielding a consistent token budget across rigs and image sizes. Finally, we enrich scene queries with positional encodings over the physical range to encode metric coordinates. The result is the geometry-aware scene tokens $B \in \mathbb{R}^{H_b \times W_b \times C_b}$ that preserve high-level semantics from the foundation backbone while remaining camera- and resolution-agnostic.

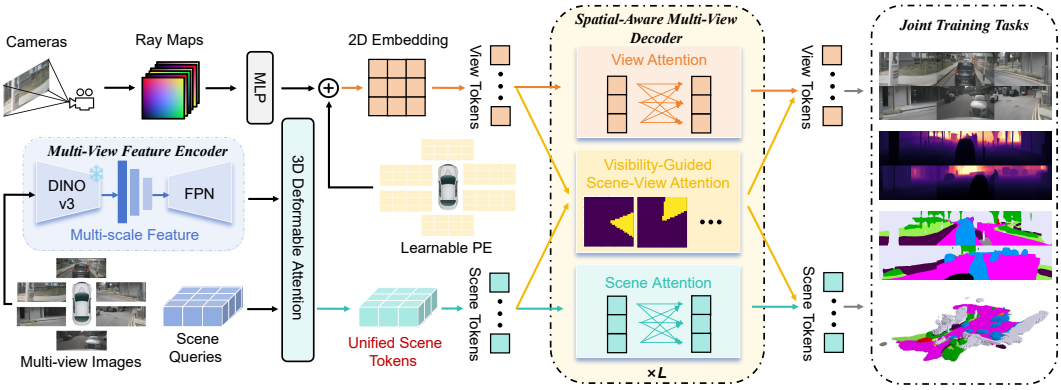



Figure 3: **Architecture of DriveTok.** Surround-view images are encoded by a 3D scene encoder. View tokens, augmented with learned 2D positions and Plücker-ray embeddings, interact with scene tokens through a spatial-aware multi-view transformer with visibility-guided attention. Joint training uses image reconstruction, depth prediction, semantic prediction, occupancy prediction, and latent semantic regularization.

### 3.3 Spatial-Aware Multi-View Decoder

To enable structured interaction between the unified scene tokens and per-camera view tokens, we introduce the spatial-aware multi-view transformer. Unlike conventional dense cross-attention that treats all token pairs equally, our transformer employs the visibility constraint to ensure that only physically visible scene regions interact with the corresponding camera views.

Specifically, our spatial-aware multi-view transformer establishes visibility-guided scene-view attention between the scene tokens and the view tokens within a standard ViT (Dosovitskiy, 2020) framework. Each scene token represents an ego-centric spatial cell, while each view token is initialized from its 2D positional embedding $\mathbf{E}$ plus a Plücker embedding $\mathbf{P}$ of the corresponding viewing ray in the ego frame:

$$\mathbf{v}_{i,p} = \mathbf{E}^{\text{2D}}_{i,p} \; + \; \mathbf{P}\big(\ell_{i,p}\big), \tag{3}$$

where $i$ indexes the camera and $p$ indexes the image patch. The Plücker ray $\ell_{i,p} = (\hat{d}, m)$ encodes the ray that starts at the camera center and passes through patch $p$:

$$\ell_{i,p} = \text{PlückerRay}\big(K_i, R_i, t_i, \tilde{\mathbf{u}}_{i,p}\big), \tag{4}$$

$$\mathbf{P}\big(\ell_{i,p}\big) = \text{MLP}\big([\hat{\mathbf{d}}_{i,p}; \, \mathbf{m}_{i,p}]\big). \tag{5}$$

By injecting $\ell_{i,p}$, we distinguish tokens from different viewpoints that may share similar 2D appearance and provides a camera-aware geometric prior, which improves cross-view disambiguation and alignment with the 3D space.

To ensure physically meaningful communication between these two token types, we introduce a visibility-based attention mask $M$ that masks out impossible scene-view correspondences. Let the scene tokens be

$$\mathbf{B} = [\mathbf{b}_1, \mathbf{b}_2, \ldots, \mathbf{b}_{N_b}] \in \mathbb{R}^{N_b \times D}, \tag{6}$$

and the concatenated view tokens from all cameras be

$$\mathbf{V} = [\mathbf{v}_{1,1}, \ldots, \mathbf{v}_{i,p}, \ldots, \mathbf{v}_{N,N_p}] \in \mathbb{R}^{N_v \times D}, \tag{7}$$

where $N$ is the number of camera views, $N_p = H_p \times W_p$ is the number of image patches per view, $N_b$ and $N_v$ denote the total numbers of scene and view tokens, respectively.

For each camera $i$ and scene cell $j$, we precompute a binary visibility mask $M$:

$$M_{i,j} = \begin{cases} 1, & \text{if scene cell } j \text{ is visible in camera } i, \\ 0, & \text{otherwise.} \end{cases} \tag{8}$$

This mask is applied symmetrically to both view→scene and scene→view directions, while the scene→scene and view→view directions remain unmasked.

Within each ViT block, the visibility-aware multi-head self-attention is computed as

$$\mathrm{Attn}(\mathbf{Q}, \mathbf{K}, \mathbf{V}, \mathbf{M}) = \mathrm{softmax}\left(\frac{\mathbf{Q}\mathbf{K}^\top}{\sqrt{d}} \odot \mathbf{M}\right)\mathbf{V}. \tag{9}$$

After passing through $L$ stacked transformer layers, the updated token sequence becomes

$$[\mathbf{B}', \mathbf{V}'] = \mathrm{Transformer}_L([\mathbf{B}, \mathbf{V}], \mathbf{M}). \tag{10}$$

This architecture enables bi-directional, geometry-consistent information exchange. Scene tokens aggregate view-specific details from the cameras that observe them, while view tokens are enriched with scene-level spatial context reflecting the actual 3D layout of the driving scene. The resulting unified scene tokens maintain both semantic and geometric consistency across all camera views.

### 3.4 Unified Reconstruction and Understanding

We illustrate the overall architecture of our model in Figure 3. DriveTok first lifts features from all cameras into shared 3D scene tokens, then updates scene and view tokens through a spatially aware transformer to support unified image reconstruction and 3D understanding.

We jointly train the scene tokens with three image-space tasks, namely RGB reconstruction, depth prediction, and semantic prediction, along with 3D occupancy prediction and latent semantic regularization. The image-space heads decode updated view tokens with DPT-style decoders (Ranftl et al., 2021), while a 3D head predicts semantic occupancy from scene tokens.

**Overall Training Objective.** The overall training objectives are as follows.

$$\mathcal{L}_{\text{total}} = \lambda_{\text{rgb}}\mathcal{L}_{\text{rgb}} + \lambda_{\text{depth}}\mathcal{L}_{\text{depth}} + \lambda_{\text{sem}}\mathcal{L}_{\text{sem}} + \lambda_{\text{occ}}\mathcal{L}_{\text{occ}} + \lambda_{\text{reg}}\mathcal{L}_{\text{reg}}. \tag{11}$$

The RGB head reconstructs each camera image using pixel, perceptual, and adversarial supervision. We align dense MoGe-2 pseudo depth (Wang et al., 2025c) to sparse projected LiDAR measurements and supervise the depth head with a robust pixel loss and an image-gradient term. Projected LiDAR Segmentation labels provide sparse per-pixel supervision for the image-space semantic head. The 3D head predicts voxel-wise semantic occupancy, encouraging the scene tokens to encode geometry and class information. We additionally project semantic occupancy labels onto the scene grid and supervise the scene tokens directly. We provide more details in Section A.4.

## 4 Experiments

### 4.1 Experiment Setups

**Implementation Details.** We train DriveTok on nuScenes (Caesar et al., 2020) using six surround-view cameras with images resized to $256 \times 704$. The model combines a pretrained DINOv3-ViTB backbone (Siméoni et al., 2025), a BEV scene encoder, and a ViT-Base multi-view decoder. We supervise depth, image semantics, and 3D occupancy using metric-aligned pseudo depth, projected LiDARSeg labels, and SurroundOcc annotations (Wei et al., 2023b), respectively. The model configuration, label construction, and training schedule are detailed in Section A.5.

**Evaluation and Metrics.** We evaluate all tasks on the nuScenes validation set, reporting PSNR and SSIM for reconstruction, AbsRel and $\delta < 1.25$ for depth, and IoU and mIoU for occupancy. The baseline settings and full evaluation protocol are provided in Section A.5.

### 4.2 Multi-View Reconstruction

We compared DriveTok with representative image tokenizers to assess whether our method encodes multi-view inputs into unified scene tokens that preserve fine-grained appearance information. As shown in Table 1, our method achieves performance comparable to all baselines across all six cameras, demonstrating that DriveTok can be adapted to multi-view inputs in autonomous driving. Additionally, we present qualitative results of our method in Figure 4. DriveTok reconstructs textures and preserves cross-view consistency at overlapping fields of view. These observations highlight the advantage of adopting a unified scene representation for surround-view scenarios.

Table 1: **Comparison of image tokenizers on nuScenes dataset.** ‡ denotes VQGAN was trained on Imagenet (Deng et al., 2009), † indicates trained on OpenImages (Kuznetsova et al., 2020). Tok. denotes the tokenization stage only (w/o decoder).

| Method | Input | PSNR↑ | SSIM↑ | Latency (ms)↓ | Memory (MB)↓ |
|---|---|---|---|---|---|
| VQGAN‡ (Esser et al., 2021) | 256×256 | 24.41 | 0.674 | 63.31 | 1440.7 |
| VQGAN† (Esser et al., 2021) | 256×256 | 26.77 | 0.756 | 63.34 | 1440.7 |
| VQGAN†,* (Esser et al., 2021) | 256×256 | 27.04 | 0.758 | 63.31 | 1440.7 |
| ViT-VQGAN (Yu et al., 2021) | 256×256 | 26.96 | 0.763 | **59.23** | **1354.4** |
| ViT-VQGAN* (Yu et al., 2021) | 256×256 | 26.43 | 0.741 | **59.23** | **1354.4** |
| FlowMo-Lo (Sargent et al., 2025) | 256×256 | 23.11 | 0.696 | 248.04 | 8434.8 |
| FlowMo-Hi (Sargent et al., 2025) | 256×256 | **27.91** | **0.793** | 259.70 | 8436.6 |
| VGD (Lin et al., 2025) | 6×114×228 | 23.87 | **0.802** | – | – |
| **DriveTok** | 6×114×228 | **28.32** | 0.769 | – | – |
| BEV-VAE (Chen & Zhao, 2025) | 6×256×256 | 26.32 | 0.746 | 183.81 | **3780.68** |
| **DriveTok** | 6×256×256 | 27.13 | 0.723 | **21.86** | 3957.95 |
| **DriveTok** | 6×256×704 | **27.89** | **0.747** | 87.96 | 4423.73 |

Table 2: **Monocular depth references on nuScenes.** All models are evaluated at pixels with projected LiDAR ground truth; monocular baselines are used zero-shot.

| Method | Input | absrel ↓ | $\delta < 1.25$ ↑ | latency ↓ | Peak Memory ↓ |
|---|---|---|---|---|---|
| UniDepthV2-B (Piccinelli et al., 2025) | 256 × 704 | 0.52 | 0.30 | **62.45 ms** | **2194.65 MB** |
| UniDepthV2-L (Piccinelli et al., 2025) | 256 × 704 | 0.40 | 0.31 | 100.45 ms | 3902.32 MB |
| DepthPro (Bochkovskii et al., 2024) | 256 × 704 | 0.26 | 0.47 | 5179.93 ms | 30559.55 MB |
| Metric3D-V2 (Hu et al., 2024) | 256 × 704 | 0.28 | 0.69 | 403.49 ms | 2982.53 MB |
| **DriveTok** | 6 × 256 × 704 | **0.08** | **0.93** | 383.53 ms | 8562.64 MB |

Table 3: **Multi-view depth prediction on nuScenes.** Metrics are computed at pixels with projected LiDAR ground truth.

| Method | absrel ↓ | $\delta < 1.25$ ↑ | Method | absrel ↓ | $\delta < 1.25$ ↑ |
|---|---|---|---|---|---|
| SurroundDepth (Wei et al., 2023a) | 0.28 | 0.66 | Dist4D (Guo et al., 2025) | 0.39 | 0.58 |
| R3D3 (Schmied et al., 2023) | 0.25 | 0.73 | OmniNWM (Li et al., 2025a) | 0.23 | 0.81 |
| SelfOcc (Huang et al., 2024a) | 0.23 | 0.75 | **DriveTok** | **0.08** | **0.93** |

### 4.3 Scene Understanding

**Depth Prediction.** We evaluated the effectiveness of DriveTok in learning depth-aware representations that capture both metric scale and scene geometry from multi-view inputs. We compared DriveTok with representative monocular depth estimation methods and multi-view depth predictors. As shown in Table 2 and Table 3, DriveTok achieves the lowest AbsRel and the highest $\delta < 1.25$ among all compared methods. These results demonstrate that the unified scene tokens effectively capture both local geometry and global scene structure. Moreover, we further project the predicted surround-view depth into the world coordinate system to obtain 3D reconstructed point clouds, as illustrated in Figure 4, verifying the superiority of our method in maintaining geometry consistency.

**3D Occupancy Prediction.** We further evaluated our DriveTok by testing its capability on occupancy prediction task, which directly measures how well the learned tokens capture geometric and semantic structure of the driving scene. Specifically, We compared our DriveTok against recent occupancy predition models. The quantitative results summarized in Table 4 show that our DriveTok achieves performance comparable to state-of-the-art methods under the same setting, highlighting its superior ability to learn rich spatial and semantic information from multi-view inputs.

**Semantic Prediction.** To further verify whether our scene tokens encode semantic information, we conducted qualitative visualizations on the semantic prediction task, as shown in Figure 4. We do not provide quantitative reults due to the lack of dense ground-truth labels. Trained only with sparse LiDARSeg labels, DriveTok successfully predicts coherent semantic layouts that align well with object semantics and scene geometry. The predictions remain consistent across adjacent viewpoints, indicating that our scene tokens capture strong cross-view semantic correspondences. DriveTok reliably differentiates complex scene elements such as sidewalks, vegetation, and distant structures.

Table 4: **3D semantic occupancy prediction results on nuScenes.** The highest value in each column is bold, and the second highest distinct value is underlined. Tied values share the same rank.

| Method | IoU | mIoU | barrier | bicycle | bus | car | const. veh. | motorcycle | pedestrian | traffic cone | trailer | truck | drive. suf. | other flat | sidewalk | terrain | manmade | vegetation |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| MonoScene (Cao & De Charette, 2022) | 23.96 | 7.31 | 4.03 | 0.35 | 8.00 | 8.04 | 2.90 | 0.28 | 1.16 | 0.67 | 4.01 | 4.35 | 27.72 | 5.20 | 15.13 | 11.29 | 9.03 | 14.86 |
| BEVFormer (Li et al., 2024) | 30.50 | 16.75 | 14.22 | 6.58 | 23.46 | 28.28 | 8.66 | 10.77 | 6.64 | 4.05 | 11.20 | 17.78 | 37.28 | 18.00 | 22.88 | 22.17 | 13.80 | 22.21 |
| TPVFormer (Huang et al., 2023) | 30.86 | 17.10 | 15.96 | 5.31 | 23.86 | 27.32 | 9.79 | 8.74 | 7.09 | 5.20 | 10.97 | 19.22 | 38.87 | 21.25 | 24.26 | 23.15 | 11.73 | 20.81 |
| OccFormer (Zhang et al., 2023) | 31.39 | 19.03 | 18.65 | 10.41 | 23.92 | 30.29 | 10.31 | 14.19 | 13.59 | 10.13 | 12.49 | 20.77 | 38.78 | 19.79 | 24.19 | 22.21 | 13.48 | 21.35 |
| SurroundOcc (Wei et al., 2023b) | 31.49 | **30.30** | **20.59** | 11.68 | **28.06** | **30.86** | 10.70 | 15.14 | **14.09** | **12.06** | **14.38** | **22.26** | 37.29 | 23.70 | 24.49 | 22.77 | 14.89 | 21.86 |
| GaussianFormer (Huang et al., 2024b) | 29.83 | 19.10 | 19.52 | 11.26 | 26.11 | 29.78 | 10.47 | 13.83 | 12.58 | 8.67 | 12.74 | 21.57 | 39.63 | 23.28 | 24.46 | 22.99 | 9.59 | 19.12 |
| GaussianFormer-2 (Huang et al., 2025) | 30.56 | 20.02 | 20.15 | 12.99 | 27.61 | 30.23 | 11.19 | 15.31 | 12.64 | 9.63 | 13.31 | **22.26** | 39.68 | 23.47 | 25.62 | 23.20 | 12.25 | 20.73 |
| QuadricFormer (Zuo et al., 2025b) | 31.22 | 20.12 | 19.58 | **13.11** | 27.27 | 29.64 | **11.25** | **16.26** | 12.65 | 9.15 | 12.51 | 21.24 | 40.20 | 24.34 | 25.69 | 24.24 | 12.95 | 21.86 |
| **DriveTok** | **33.32** | 20.06 | 20.04 | 9.81 | 26.11 | 27.87 | 11.14 | 13.44 | 9.52 | 8.65 | 12.67 | 20.44 | **41.79** | **25.31** | **27.68** | **26.37** | **16.59** | **23.57** |

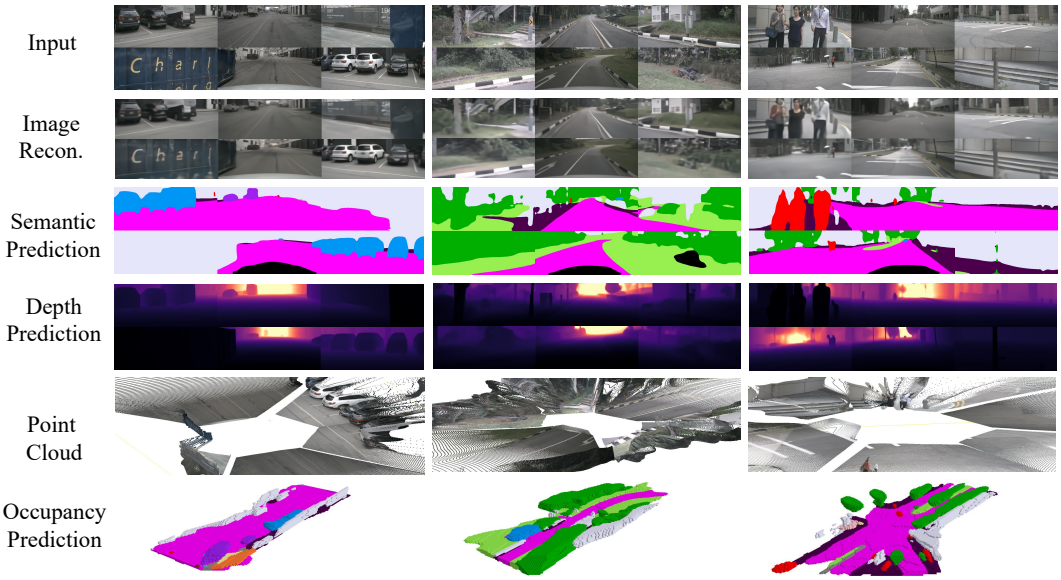



Figure 4: **DriveTok predictions across tasks.** RGB, depth, semantic, points, and occupancy outputs decoded from the same scene representation show consistent geometry across the surround views.

### 4.4 Downstream Forecasting and Planning

To test whether the representation remains useful beyond its pretraining heads, we use DriveTok as the unified visual tokenizer in a driving world model, denoted DriveTok-World. Table 5 reports camera-only 4D occupancy forecasting. DriveTok-World improves average semantic mIoU by 4.03 points and average geometric IoU by 16.66 points over the strongest listed baseline while running at approximately 4 FPS. The same representation also supports motion planning as shown in Table 6, where it obtains the lowest average L2 error and competitive collision rate.

### 4.5 Experimental Analysis

**Latent-Space Visualization.** We visualize the BEV feature maps with principal component analysis, the ground-truth labels used for semantic regularization, and the decoded versus ground-truth images in Figure 5. The proposed semantic regularization successfully injects explicit semantic structure into the BEV latent space, avoiding the radial patterns common in conventional perception methods.

**Joint Task Training.** To validate the effectiveness of our joint task training strategy, we trained the scene tokens under different combinations of training objectives and then fine-tuned the occupancy head and measured the performance of occupancy prediction. The results in Table 7 show that the image reconstruction task encourages the scene tokens to retain low-level texture information from the input images, while the depth and semantic prediction tasks enable the scene tokens to capture rich 3D spatial structure and semantic cues, as reflected in the occupancy results. Finally, although incorporating the occupancy prediction task degrades the image reconstruction quality, it further enhances the spatial and semantic information encoded in the scene tokens.

Table 5: **Camera-only 4D occupancy forecasting.**

| Method | Semantic mIoU↑ | | | | Geometric IoU↑ | | | | FPS↑ |
|---|---|---|---|---|---|---|---|---|---|
| | 1s | 2s | 3s | Avg. | 1s | 2s | 3s | Avg. | |
| DOME-F (Gu et al., 2024) | 24.12 | 17.41 | 13.24 | 18.25 | 35.18 | 27.90 | 23.44 | 28.84 | – |
| $I^2$-World-STC (Liao et al., 2025) | 21.67 | 18.78 | 16.47 | 18.97 | 30.55 | 28.76 | 26.99 | 28.77 | 4.21 |
| DTT (Xu et al., 2025) | 24.87 | 18.30 | 15.63 | 19.60 | 38.98 | 37.45 | 31.89 | 36.11 | – |
| COME-F (Shi et al., 2025) | 26.56 | 21.73 | 18.49 | 22.26 | 48.08 | 43.84 | 40.28 | 44.07 | – |
| **DriveTok-World** | **30.90** | **25.70** | **22.44** | **26.29** | **64.02** | **60.59** | **57.75** | **60.73** | ∼4.0 |

Table 6: **Camera-only motion planning.** L2 and collision metrics are reported at 1, 2, and 3 seconds.

| Method | L2↓ | | | | Collision↓ | | | | FPS↑ |
|---|---|---|---|---|---|---|---|---|---|
| | 1s | 2s | 3s | Avg. | 1s | 2s | 3s | Avg. | |
| UniAD (Hu et al., 2023b) | 0.48 | 0.96 | 1.65 | 1.03 | 0.05 | 0.17 | **0.71** | **0.31** | 1.8 |
| GenAD (Zheng et al., 2024a) | 0.36 | 0.83 | 1.55 | 0.91 | 0.06 | 0.23 | 1.00 | 0.43 | – |
| OmniDrive (Wang et al., 2025d) | 0.40 | 0.80 | **1.32** | 0.84 | 0.04 | 0.46 | 2.32 | 0.94 | – |
| UniUGP (Lu et al., 2025) | 0.58 | 1.14 | 1.95 | 1.23 | 0.01 | 0.19 | 0.81 | 0.33 | – |
| **DriveTok-World** | **0.23** | **0.65** | 1.46 | **0.78** | **0.00** | **0.08** | 1.09 | 0.39 | ∼6.0 |

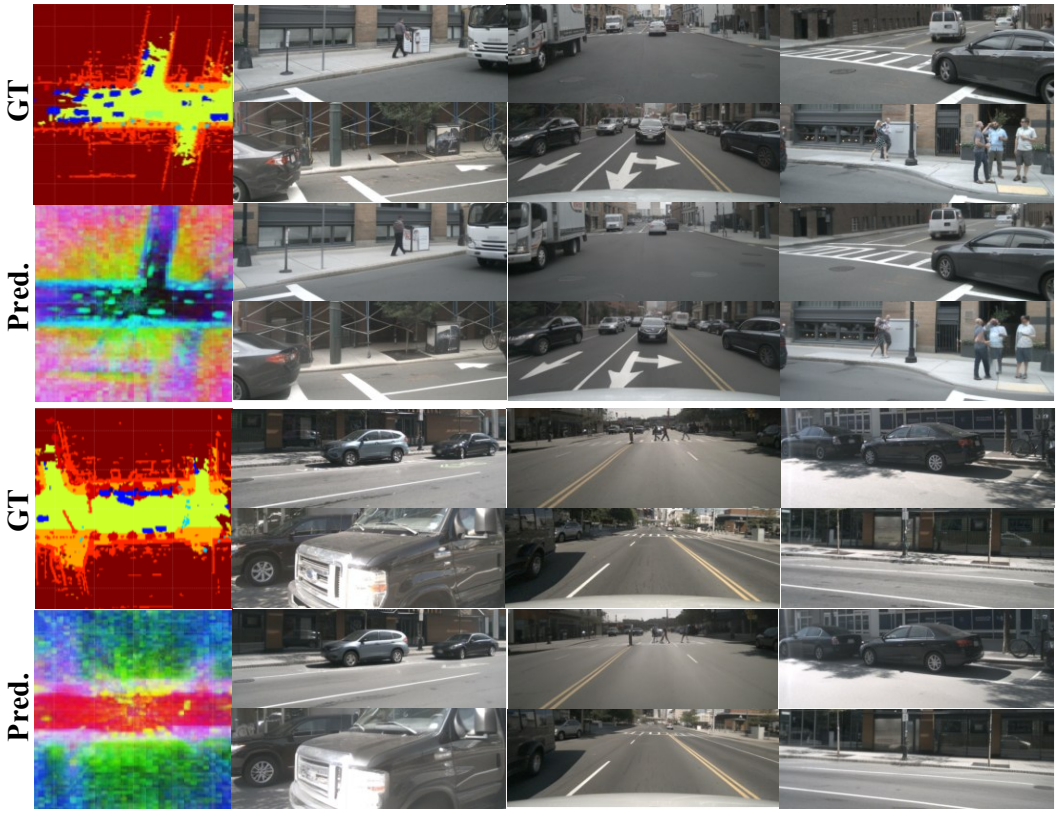



Figure 5: **Visualizations of BEV scene tokens and images.** We visualize BEV features with PCA together with the semantic-regularization targets and decoded versus ground-truth images. The PCA projection shows that the scene tokens model semantic scene structure as well as image texture, without the radial patterns observed in conventional BEV features.

**Visibility-Guided Attention.** We conducted an ablation study in which we removed the visibility mask and allowed each scene token to attend to all view tokens unconditionally. We first pretrained the scene tokens using only the image reconstruction task. After pretraining, we froze all components except the occupancy head and fine-tuned solely on the 3D occupancy prediction task. As illustrated in Table 8, removing visibility-guided attention eliminates the local geometric inductive bias, causing scene tokens to overfit image textures rather than learn meaningful spatial structure. In contrast, visibility-guided attention serves as an effective regularizer that prevents overfitting and enforces physically valid interactions, thereby encouraging the model to encode spatial information.

Table 7: **Ablation study on the joint task training strategy.** Different combinations of reconstruction, depth, semantic, and occupancy objectives reveal a trade-off between texture quality and geometry-semantic understanding, where stronger multi-task supervision improves perferormance.

| Setting | Tasks | | | | Texture | | Geometry & Semantics | |
|---|---|---|---|---|---|---|---|---|
| | Recon. | Depth | Sem. | Occ. | PSNR↑ | SSIM↑ | IoU↑ | mIoU↑ |
| Image Tokenizer | ✓ | | | | **28.84** | **0.770** | 12.81 | 3.84 |
| Geometry-Aware Model | ✓ | ✓ | | | 28.52 | 0.764 | 20.72 | 9.36 |
| Scene-Understanding Model | ✓ | ✓ | ✓ | | 28.06 | 0.756 | 24.05 | 11.68 |
| Unified Model | ✓ | ✓ | ✓ | ✓ | 27.89 | 0.747 | **33.32** | **20.06** |

Table 8: **Ablation on visibility-guided attention.** Removing visibility guidance reduces 3D occupancy prediction performance.

| Setting | Texture | | Geo. & Sem. | |
|---|---|---|---|---|
| | PSNR↑ | SSIM↑ | IoU↑ | mIoU↑ |
| w/o Geo-Attn | **29.98** | **0.798** | 5.32 | 0.59 |
| **w/ Geo-Attn** | 28.84 | 0.770 | **12.81** | **3.84** |

Table 9: **Ablation of view-token positional cues.** All variants are evaluated at the same training checkpoint.

| Embedding | PSNR↑ | SSIM↑ | IoU↑ | mIoU↑ |
|---|---|---|---|---|
| 2D only | 24.69 | 0.638 | 31.98 | 19.39 |
| Plücker only | 23.67 | 0.609 | 32.21 | 19.47 |
| **2D + Plücker** | **25.24** | **0.667** | **32.96** | **19.53** |

Table 10: **Comparison with a VGGT-based tokenizer (Wang et al., 2025a).** Both representations use the same task heads and supervision.

| Method | PSNR↑ | SSIM↑ | IoU↑ | mIoU↑ |
|---|---|---|---|---|
| VGGT-based | 25.45 | 0.703 | 25.08 | 8.60 |
| **DriveTok** | **27.89** | **0.747** | **33.32** | **20.06** |

Table 11: **Zero-shot depth on PandaSet.** Models are trained without PandaSet.

| Method | AbsRel↓ | $\delta < 1.25$↑ |
|---|---|---|
| Metric3D-v2 (Hu et al., 2024) | 0.481 | 0.130 |
| DVGT (Zuo et al., 2025a) | 0.405 | 0.321 |
| **DriveTok** | **0.289** | **0.626** |

**View-Token Positional Cues.** The 2D positional embedding encodes image-plane locality, whereas the Plücker embedding identifies the camera ray associated with each patch. At the same controlled checkpoint, combining them improves both appearance reconstruction and occupancy over either cue alone (Table 9). The gain is consistent across PSNR, SSIM, mIoU, and IoU, indicating that image-plane structure and ray geometry provide complementary information.

**Representation Architecture.** General-purpose geometry encoders often retain dense view-level features rather than constructing a compact scene-aligned sequence. To isolate this representation choice from the training objectives, we equip a VGGT encoder (Wang et al., 2025a) with the same reconstruction and occupancy heads and train it with the same supervision as DriveTok. Table 10 shows a 2.44 dB PSNR gap and an 11.46-point mIoU gap in favor of DriveTok. The controlled comparison supports aggregating multi-view evidence into a compact metric scene grid instead of passing dense view features directly to the task heads.

**Cross-Dataset Generalization.** DriveTok uses camera calibration explicitly. A zero-shot occupancy comparison across datasets with different rigs and voxel protocols would therefore mix representation transfer with changes in calibration, labels, and evaluation grids. We instead evaluate zero-shot depth on PandaSet (Xiao et al., 2021), which uses a different six-camera rig. 3D occupancy models trained and evaluated separately under the OpenScene protocol are reported in Section D.2. On PandaSet, the nuScenes-trained DriveTok reduces AbsRel from 0.405 to 0.289 relative to DVGT and raises $\delta < 1.25$ from 0.321 to 0.626 (Table 11).

## 5 Conclusion

In this work, we introduced DriveTok, a unified scene tokenizer for multi-view driving scene reconstruction and understanding. DriveTok produces compact scene tokens that simultaneously encode texture, geometry, and semantics via the joint task training strategy. Our spatial-aware multi-view decoder, equipped with visibility-guided attention, enables geometry-consistent interaction between scene tokens and visual tokens. Extensive experiments on nuScenes (Caesar et al., 2020) demonstrate that DriveTok achieves competitive image reconstruction quality and state-of-the-art 3D semantic occupancy performance. Moreover, DriveTok attains the best results on the metric depth prediction task, validating the effectiveness of our tokenization framework.

### AI USE STATEMENT

In this work, we used generative AI tools for language polishing and for guidance on figure layout and presentation, to improve the readability of this paper. We have not used generative AI tools for any tasks with required disclosure, including generating synthetic datasets, developing theoretical models or conceptual frameworks, formulating mathematical claims, assisting in the writing of proofs, proposing or refining hypotheses, designing or providing feedback on research methodology or experiments, implementing methods, cleaning and reformatting datasets, supporting qualitative and thematic data analysis, or interpreting results. The remaining required disclosure tasks are not applicable to this work. We have reviewed all AI-assisted work. We take responsibility for the final content of this work, including text, claims, or artifacts produced with the aid of generative AI.

### ETHICS STATEMENT

This work does not involve human subjects, sensitive data, or privacy concerns. All datasets used are publicly available, and we foresee no harmful applications or conflicts of interest arising from this research.

### REPRODUCIBILITY STATEMENT

We have made efforts to ensure the reproducibility of our work. The architecture and training details of drivetok are fully described in main content. All datasets used in our experiments are publicly available. The code and checkpoint of our model will be revealed in the article upon acceptance.

### AUTHOR CONTRIBUTIONS

Not disclosed during double-blind peer-review process.

### ACKNOWLEDGMENTS

Not disclosed during double-blind peer-review process.

## REFERENCES

Aleksei Bochkovskii, AmaÃĢl Delaunoy, Hugo Germain, Marcel Santos, Yichao Zhou, Stephan R Richter, and Vladlen Koltun. Depth pro: Sharp monocular metric depth in less than a second. *arXiv preprint arXiv:2410.02073*, 2024.

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 *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 11621–11631, 2020.

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

Zeming Chen and Hang Zhao. Bev-vae: Multi-view image generation with spatial consistency for autonomous driving. *arXiv preprint arXiv:2507.00707*, 2025.

Tri Dao. Flashattention-2: Faster attention with better parallelism and work partitioning. *arXiv preprint arXiv:2307.08691*, 2023.

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

Alexey Dosovitskiy. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.

Patrick Esser, Robin Rombach, and Bjorn Ommer. Taming transformers for high-resolution image synthesis. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 12873–12883, 2021.

Andreas Geiger, Philip Lenz, Christoph Stiller, and Raquel Urtasun. Vision meets robotics: The kitti dataset. *The international journal of robotics research*, 32(11):1231–1237, 2013.

Clément Godard, Oisin Mac Aodha, Michael Firman, and Gabriel J Brostow. Digging into self-supervised monocular depth estimation. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 3828–3838, 2019.

Songen Gu, Wei Yin, Bu Jin, Xiaoyang Guo, Junming Wang, Haodong Li, Qian Zhang, and Xiaoxiao Long. DOME: Taming diffusion model into high-fidelity controllable occupancy world model. *arXiv preprint arXiv:2410.10429*, 2024.

Jiazhe Guo, Yikang Ding, Xiwu Chen, Shuo Chen, Bohan Li, Yingshuang Zou, Xiaoyang Lyu, Feiyang Tan, Xiaojuan Qi, Zhiheng Li, et al. Dist-4d: Disentangled spatiotemporal diffusion with metric depth for 4d driving scene generation. *arXiv preprint arXiv:2503.15208*, 2025.

Anthony Hu, Lloyd Russell, Hudson Yeo, Zak Murez, George Fedoseev, Alex Kendall, Jamie Shotton, and Gianluca Corrado. Gaia-1: A generative world model for autonomous driving. *arXiv preprint arXiv:2309.17080*, 2023a.

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 Transactions on Pattern Analysis and Machine Intelligence*, 2024.

Yihan Hu, Jiazhi Yang, Li Chen, Keyu Li, Chonghao Sima, Xizhou Zhu, Siqi Chai, Senyao Du, Tianwei Lin, Wenhai Wang, Lewei Lu, Xiaosong Jia, Qiang Liu, Jifeng Dai, Yu Qiao, and Hongyang Li. Planning-oriented autonomous driving. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023b.

Junjie Huang, Guan Huang, Zheng Zhu, Yun Ye, and Dalong Du. Bevdet: High-performance multi-camera 3d object detection in bird-eye-view. *arXiv preprint arXiv:2112.11790*, 2021.

Yuanhui Huang, Wenzhao Zheng, Yunpeng Zhang, Jie Zhou, and Jiwen Lu. Tri-perspective view for vision-based 3d semantic occupancy prediction. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 9223–9232, 2023.

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*, 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 *European Conference on Computer Vision*, pp. 376–393. Springer, 2024b.

Yuanhui Huang, Amonnut Thammatadatrakoon, Wenzhao Zheng, Yunpeng Zhang, Dalong Du, and Jiwen Lu. Gaussianformer-2: Probabilistic gaussian superposition for efficient 3d occupancy prediction. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 27477–27486, 2025.

Boris Ivanovic, Cristiano Saltori, Yurong You, Yan Wang, Wenjie Luo, and Marco Pavone. Efficient multi-camera tokenization with triplanes for end-to-end driving. *arXiv preprint arXiv:2506.12251*, 2025.

Anqing Jiang, Yu Gao, Zhigang Sun, Yiru Wang, Jijun Wang, Jinghao Chai, Qian Cao, Yuweng Heng, Hao Jiang, Yunda Dong, et al. Diffvla: Vision-language guided diffusion planning for autonomous driving. *arXiv preprint arXiv:2505.19381*, 2025.

Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C Berg, Wan-Yen Lo, et al. Segment anything. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 4015–4026, 2023.

Alina Kuznetsova, Hassan Rom, Neil Alldrin, Jasper Uijlings, Ivan Krasin, Jordi Pont-Tuset, Shahab Kamali, Stefan Popov, Matteo Malloci, Alexander Kolesnikov, et al. The open images dataset v4: Unified image classification, object detection, and visual relationship detection at scale. *International journal of computer vision*, 128(7):1956–1981, 2020.

Bohan Li, Zhuang Ma, Dalong Du, Baorui Peng, Zhujin Liang, Zhenqiang Liu, Chao Ma, Yueming Jin, Hao Zhao, Wenjun Zeng, et al. Omninwm: Omniscient driving navigation world models. *arXiv preprint arXiv:2510.18313*, 2025a.

Yingyan Li, Shuyao Shang, Weisong Liu, Bing Zhan, Haochen Wang, Yuqi Wang, Yuntao Chen, Xiaoman Wang, Yasong An, Chufeng Tang, et al. Drivevla-w0: World models amplify data scaling law in autonomous driving. *arXiv preprint arXiv:2510.12796*, 2025b.

Yinhao Li, Zheng Ge, Guanyi Yu, Jinrong Yang, Zengran Wang, Yukang Shi, Jianjian Sun, and Zeming Li. Bevdepth: Acquisition of reliable depth for multi-view 3d object detection. In *Proceedings of the AAAI conference on artificial intelligence*, volume 37, pp. 1477–1485, 2023.

Zhiqi Li, Wenhai Wang, Hongyang Li, Enze Xie, Chonghao Sima, Tong Lu, Qiao Yu, and Jifeng Dai. Bevformer: learning bird's-eye-view representation from lidar-camera via spatiotemporal transformers. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2024.

Zhimin Liao, Ping Wei, Ruijie Zhang, Shuaijia Chen, Haoxuan Wang, and Ziyang Ren. $I^2$-World: Intra-inter tokenization for efficient dynamic 4d scene forecasting. *arXiv preprint arXiv:2507.09144*, 2025.

Junhong Lin, Kangli Wang, Shunzhou Wang, Songlin Fan, Ge Li, and Wei Gao. VGD: Visual geometry gaussian splatting for feed-forward surround-view driving reconstruction. *arXiv preprint arXiv:2510.19578*, 2025.

Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. Feature pyramid networks for object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2117–2125, 2017.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017.

Hao Lu, Ziyang Liu, Guangfeng Jiang, Yuanfei Luo, Sheng Chen, Yangang Zhang, and Ying-Cong Chen. UniUGP: Unifying understanding, generation, and planning for end-to-end autonomous driving. *arXiv preprint arXiv:2512.09864*, 2025.

Jiageng Mao, Yuxi Qian, Junjie Ye, Hang Zhao, and Yue Wang. Gpt-driver: Learning to drive with gpt. *arXiv preprint arXiv:2310.01415*, 2023.

Jonah Philion and Sanja Fidler. Lift, splat, shoot: Encoding images from arbitrary camera rigs by implicitly unprojecting to 3d. In *European conference on computer vision*, pp. 194–210. Springer, 2020.

Luigi Piccinelli, Christos Sakaridis, Yung-Hsu Yang, Mattia Segu, Siyuan Li, Wim Abbeloos, and Luc Van Gool. Unidepthv2: Universal monocular metric depth estimation made simpler. *arXiv preprint arXiv:2502.20110*, 2025.

René Ranftl, Katrin Lasinger, David Hafner, Konrad Schindler, and Vladlen Koltun. Towards robust monocular depth estimation: Mixing datasets for zero-shot cross-dataset transfer. *IEEE transactions on pattern analysis and machine intelligence*, 44(3):1623–1637, 2020.

René Ranftl, Alexey Bochkovskiy, and Vladlen Koltun. Vision transformers for dense prediction. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 12179–12188, 2021.

Kyle Sargent, Kyle Hsu, Justin Johnson, Li Fei-Fei, and Jiajun Wu. Flow to the mode: Mode-seeking diffusion autoencoders for state-of-the-art image tokenization. *arXiv preprint arXiv:2503.11056*, 2025.

Aron Schmied, Tobias Fischer, Martin Danelljan, Marc Pollefeys, and Fisher Yu. R3d3: Dense 3d reconstruction of dynamic scenes from multiple cameras. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 3216–3226, 2023.

Hao Shao, Yuxuan Hu, Letian Wang, Guanglu Song, Steven L Waslander, Yu Liu, and Hongsheng Li. Lmdrive: Closed-loop end-to-end driving with large language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 15120–15130, 2024.

Yining Shi, Kun Jiang, Qiang Meng, Ke Wang, Jiabao Wang, Wenchao Sun, Tuopu Wen, Mengmeng Yang, and Diange Yang. COME: Adding scene-centric forecasting control to occupancy world model. *arXiv preprint arXiv:2506.13260*, 2025.

Oriane Siméoni, Huy V Vo, Maximilian Seitzer, Federico Baldassarre, Maxime Oquab, Cijo Jose, Vasil Khalidov, Marc Szafraniec, Seungeun Yi, Michaël Ramamonjisoa, et al. Dinov3. *arXiv preprint arXiv:2508.10104*, 2025.

Shengbang Tong, David Fan, Jiachen Li, Yunyang Xiong, Xinlei Chen, Koustuv Sinha, Michael Rabbat, Yann LeCun, Saining Xie, and Zhuang Liu. Metamorph: Multimodal understanding and generation via instruction tuning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 17001–17012, 2025.

Aaron Van Den Oord, Oriol Vinyals, et al. Neural discrete representation learning. *Advances in neural information processing systems*, 30, 2017.

Hang-Sheng Wang, Xinping Zhu, Li-Shiuan Peh, and Sharad Malik. Orion: A power-performance simulator for interconnection networks. In *35th Annual IEEE/ACM International Symposium on Microarchitecture, 2002.(MICRO-35). Proceedings.*, pp. 294–305. IEEE, 2002.

Jianyuan Wang, Minghao Chen, Nikita Karaev, Andrea Vedaldi, Christian Rupprecht, and David Novotny. VGGT: Visual geometry grounded transformer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2025a.

Ruicheng Wang, Sicheng Xu, Cassie Dai, Jianfeng Xiang, Yu Deng, Xin Tong, and Jiaolong Yang. Moge: Unlocking accurate monocular geometry estimation for open-domain images with optimal training supervision. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 5261–5271, 2025b.

Ruicheng Wang, Sicheng Xu, Yue Dong, Yu Deng, Jianfeng Xiang, Zelong Lv, Guangzhong Sun, Xin Tong, and Jiaolong Yang. Moge-2: Accurate monocular geometry with metric scale and sharp details. *arXiv preprint arXiv:2507.02546*, 2025c.

Shihao Wang, Zhiding Yu, Xiaohui Jiang, Shiyi Lan, Min Shi, Nadine Chang, Jan Kautz, Ying Li, and Jose M. Alvarez. OmniDrive: A holistic vision-language dataset for autonomous driving with counterfactual reasoning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2025d.

Xiaofeng Wang, Zheng Zhu, Guan Huang, Xinze Chen, Jiagang Zhu, and Jiwen Lu. Drivedreamer: Towards real-world-drive world models for autonomous driving. In *European conference on computer vision*, pp. 55–72. Springer, 2024a.

Yuqi Wang, Jiawei He, Lue Fan, Hongxin Li, Yuntao Chen, and Zhaoxiang Zhang. Driving into the future: Multiview visual forecasting and planning with world model for autonomous driving. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 14749–14759, 2024b.

Yi Wei, Linqing Zhao, Wenzhao Zheng, Zheng Zhu, Yongming Rao, Guan Huang, Jiwen Lu, and Jie Zhou. Surrounddepth: Entangling surrounding views for self-supervised multi-camera depth estimation. In *Conference on robot learning*, pp. 539–549. PMLR, 2023a.

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*, pp. 21729–21740, 2023b.

Chengyue Wu, Xiaokang Chen, Zhiyu Wu, Yiyang Ma, Xingchao Liu, Zizheng Pan, Wen Liu, Zhenda Xie, Xingkai Yu, Chong Ruan, et al. Janus: Decoupling visual encoding for unified multimodal understanding and generation. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 12966–12977, 2025.

Pengchuan Xiao, Zhenlei Shao, Steven Hao, Zishuo Zhang, Xiaolin Chai, Judy Jiao, Zesong Li, Jian Wu, Kai Sun, Kun Jiang, Yunlong Wang, and Diange Yang. PandaSet: Advanced sensor suite dataset for autonomous driving. In *IEEE International Intelligent Transportation Systems Conference*, 2021.

Haoran Xu, Peixi Peng, Guang Tan, Yiqian Chang, Yisen Zhao, and Yonghong Tian. Delta-triplane transformers as occupancy world models. *arXiv preprint arXiv:2503.07338*, 2025.

Zhenhua Xu, Yujia Zhang, Enze Xie, Zhen Zhao, Yong Guo, Kwan-Yee K Wong, Zhenguo Li, and Hengshuang Zhao. Drivegpt4: Interpretable end-to-end autonomous driving via large language model. *IEEE Robotics and Automation Letters*, 2024.

Jingfeng Yao, Bin Yang, and Xinggang Wang. Reconstruction vs. generation: Taming optimization dilemma in latent diffusion models. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 15703–15712, 2025.

Jiahui Yu, Xin Li, Jing Yu Koh, Han Zhang, Ruoming Pang, James Qin, Alexander Ku, Yuanzhong Xu, Jason Baldridge, and Yonghui Wu. Vector-quantized image modeling with improved vqgan. *arXiv preprint arXiv:2110.04627*, 2021.

Qihang Yu, Mark Weber, Xueqing Deng, Xiaohui Shen, Daniel Cremers, and Liang-Chieh Chen. An image is worth 32 tokens for reconstruction and generation. *Advances in Neural Information Processing Systems*, 37:128940–128966, 2024.

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*, pp. 9433–9443, 2023.

Guosheng Zhao, Xiaofeng Wang, Zheng Zhu, Xinze Chen, Guan Huang, Xiaoyi Bao, and Xingang Wang. Drivedreamer-2: Llm-enhanced world models for diverse driving video generation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pp. 10412–10420, 2025.

Anlin Zheng, Xin Wen, Xuanyang Zhang, Chuofan Ma, Tiancai Wang, Gang Yu, Xiangyu Zhang, and Xiaojuan Qi. Vision foundation models as effective visual tokenizers for autoregressive image generation. *arXiv preprint arXiv:2507.08441*, 2025.

Wenzhao Zheng, Ruiqi Song, Xianda Guo, Chenming Zhang, and Long Chen. GenAD: Generative end-to-end autonomous driving. In *European Conference on Computer Vision*, 2024a.

Wenzhao Zheng, Zetian Xia, Yuanhui Huang, Sicheng Zuo, Jie Zhou, and Jiwen Lu. Doe-1: Closed-loop autonomous driving with large world model. *arXiv preprint arXiv:2412.09627*, 2024b.

Tinghui Zhou, Matthew Brown, Noah Snavely, and David G Lowe. Unsupervised learning of depth and ego-motion from video. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1851–1858, 2017.

Xingcheng Zhou, Xuyuan Han, Feng Yang, Yunpu Ma, and Alois C Knoll. Opendrivevla: Towards end-to-end autonomous driving with large vision language action model. *arXiv preprint arXiv:2503.23463*, 2025a.

Zewei Zhou, Tianhui Cai, Seth Z Zhao, Yun Zhang, Zhiyu Huang, Bolei Zhou, and Jiaqi Ma. Autovla: A vision-language-action model for end-to-end autonomous driving with adaptive reasoning and reinforcement fine-tuning. *arXiv preprint arXiv:2506.13757*, 2025b.

Sicheng Zuo, Zixun Xie, Wenzhao Zheng, Shaoqing Xu, Fang Li, Shengyin Jiang, Long Chen, Zhi-Xin Yang, and Jiwen Lu. DVGT: Driving visual geometry transformer. *arXiv preprint arXiv:2512.16919*, 2025a.

Sicheng Zuo, Wenzhao Zheng, Xiaoyong Han, Longchao Yang, Yong Pan, and Jiwen Lu. Quadricformer: Scene as superquadrics for 3d semantic occupancy prediction. *arXiv preprint arXiv:2506.10977*, 2025b.

Sicheng Zuo, Zixun Xie, Wenzhao Zheng, Shaoqing Xu, Fang Li, Hanbing Li, Long Chen, Zhi-Xin Yang, and Jiwen Lu. DVGT-2: Vision-geometry-action model for autonomous driving at scale. *arXiv preprint arXiv:2604.00813*, 2026.

# Appendix

## A Architecture and Implementation Details

This section details the scene encoder, multi-view decoder, task heads, visibility-guided attention, and training objectives.

### A.1 3D Scene Encoder

**Visual Foundation Model and Feature Sizes.** The 3D scene encoder is responsible for transforming multi-view image observations into unified scene tokens used by the reconstruction and prediction heads. We adopt a pretrained DINOv3-ViTB (Siméoni et al., 2025) backbone, followed by an FPN (Lin et al., 2017) to build a multi-scale pyramid from surround-view images. Concretely, for an input of 256×704, the backbone generates three multi-scale feature maps. The FPN fuses these features and outputs four feature levels to the BEV lifting stage. The BEV module expects a shared embedding channel (we set $C_{bev} = 256$) and four feature levels, consistent with our encoder configuration.

**Scene Encoding.** We define the BEV space as a uniform 3D grid spanning $[-51.2, 51.2] \times [-51.2, 51.2] \times [-5.0, 3.0]$ meters in the real world. The BEV encoder uses a $128 \times 128$ feature grid and samples eight heights within this range; the occupancy head later resamples its predictions to a $200 \times 200 \times 16$ voxel grid. These coordinates are fed into the BEV positional encoder, where each cell center is encoded using sinusoidal Fourier features and then linearly projected to the BEV feature dimension. This yields the learnable positional embeddings added to the BEV queries before entering the transformer.

The scene encoder receives (i) the BEV queries with positional embeddings and (ii) multi-view image features from four pyramid levels. Before fusion, we attach learned camera embeddings and pyramid-level embeddings to each feature map. The image features are then flattened across views and pyramid levels, and the associated spatial shapes and level start indices are provided to the deformable attention module, which produces the final scene tokens.

### A.2 Spatial-Aware Multi-View Decoder

To mitigate the computational overhead of the multi-view decoder when dealing with long sequences, we first patchify the input BEV features of size $128 \times 128 \times 256$ following the ViT paradigm (Dosovitskiy, 2020). This process yields scene tokens with a reduced spatial resolution of $32 \times 32$ and an embedding dimension of 768. Subsequently, our spatial-aware multi-view decoder is a ViT-Base (Dosovitskiy, 2020) run on a concatenated token sequence: $[\text{CLS}]||\text{BEV}(32 \times 32)||\text{VIEW}(6 \times 16 \times 44)$.

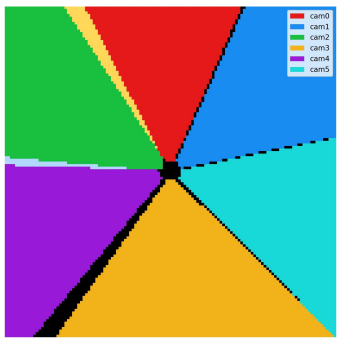



Figure 6: **Visualization of visibility-guided mask.** Each color corresponds to the region in the BEV plane that is visible from a particular camera, forming six wedge-shaped sectors that meet at the ego-vehicle center. The black region at the center corresponds to areas not visible to any camera. In addition, any colors outside these six primary sectors represent boundary zones where multiple camera frustums overlap.

**View Tokens.** View tokens are constructed by starting with a learned mask token for each patch, to which a 2D view positional grid and a learned per-camera embedding are added. For each patch center, Plücker rays are computed using the camera intrinsics and extrinsics, processed with a small MLP, and fused into the view token through a linear projection.

**Scene and View Attention.** $Scene \leftrightarrow Scene$ (BEV self-attention) is always enabled to promote global spatial context in BEV, while $View \leftrightarrow View$ (image self-attention) is likewise always enabled across all cameras and patches to encourage cross-view appearance priors.

**Visibility-Guided Scene↔View Attention.** $Scene \leftrightarrow View$ is selectively enabled through a visibility-guided mask: we first compute a boolean tensor $M \in \{0, 1\}^{B \times N_c \times N_b}$ by projecting BEV cell centers into each camera, marking a cell as visible if any height slice lies in

front of the camera and falls within the image. The mask adds a bias of $-\infty$ to scene–view logits when $M_{i,j} = 0$, giving those pairs zero attention weight in both directions; all other logits remain unchanged. The resulting camera coverage is visualized in Figure 6. The visualization verifies that the resulting visibility mask captures single-view coverage and non-visible regions, thereby enforcing geometry-consistent sparsity for visibility-guided $Scene \leftrightarrow View$ attention.

### A.3 Joint Task Heads

**RGB Reconstruction Head.** The ViewTokenDecoder adopts a DPT (Ranftl et al., 2021) architecture. Tokens are reshaped into the patch grid and projected to 256 channels with lightweight deblocking. A single-scale feature pyramid is constructed through strided convolutions, yielding four levels $\{256, 512, 1024, 1024\}$ that are refined in a top-down manner using DPT FeatureFusionBlocks. The fused representation is progressively upsampled with additional deblocking to recover full resolution, and a final convolutional head produces the 3-channel RGB output.

**Depth Prediction Head.** The DepthAdaptor applies a residual MLP to the view tokens and predicts per-camera affine parameters (a, b) to correct monocular scale and shift, after which a second ViewTokenDecoder produces a single-channel depth map using a Softplus activation. The final aligned depth is computed as $a \cdot \text{pred} + b$.

**Semantics Prediction Head.** A third ViewTokenDecoder with the same architecture is employed for the auxiliary prediction task, producing 17 output channels. Its output remains as raw logits, and the ignore label (255) is excluded during training.

**Occupancy Prediction Head.** The OccAdaptor maps BEV tokens into a 2D feature plane using LayerNorm and a linear projection, followed by lightweight depthwise–pointwise convolutions to produce a 256-channel feature map at $128^2$. The OccHead upsamples this map with a stride-aligned transposed-convolution stack, while a height-conditioning module provides FiLM parameters across eight vertical slices to lift the 2D features into a 3D volume. Optional 3D refinement blocks further process the volume, and a final $1 \times 1 \times 1$ classifier predicts 18 classes. The resulting dense volume is then resampled to the target $200 \times 200 \times 16$ occupancy grid.

### A.4 Training Objectives

We utilize a joint task training strategy to enrich the unified scene tokens with texture, semantics, and 3D geometry. We adopt three image-space tasks: image reconstruction, depth prediction, and semantic prediction, as well as an occupancy prediction task and semantic regularization for scene tokens. The image-space tasks utilize DPT (Ranftl et al., 2021) decoders that upsample per-camera view tokens to the native image resolution, while the occupancy prediction task employs a convolutional head to generate occupancy logits from scene tokens. The semantic regularization term explicitly injects semantic information into the scene tokens, promoting semantic structure learning in the latent space.

**Image Reconstruction.** Given the updated view tokens $V$, the DPT head predicts per-camera RGB images $\hat{\mathbf{I}}$. The reconstruction loss combines pixel-level fidelity, perceptual similarity, and adversarial training:

$$\mathcal{L}_{\text{rgb}} = \lambda_{\text{pix}} \|\hat{\mathbf{I}} - \mathbf{I}\|_1 + \lambda_{\text{perc}} \mathcal{L}_{\text{LPIPS}} + \lambda_{\text{adv}} \mathcal{L}_{\text{GAN}}. \tag{12}$$

Here $\|\cdot\|_1$ denotes the L1 distance evaluated over valid pixels, $\mathcal{L}_{\text{LPIPS}}$ is the learned perceptual image patch similarity loss, and $\mathcal{L}_{\text{GAN}}$ is the adversarial loss computed via a discriminator following VQGAN (Esser et al., 2021). In our implementation, we set the balancing weights to $\lambda_{\text{pix}} = 1.0$, $\lambda_{\text{perc}} = 1.0$, and $\lambda_{\text{adv}} = 0.3$.

**Depth Prediction.** Public driving datasets (Caesar et al., 2020; Geiger et al., 2013) rarely provide dense per-pixel depth ground truth. The sparsity limits pixel-wise supervision, encourages texture copying artifacts, and leaves monocular depth with unresolved metric scale. We first obtain dense pseudo depth for each pixel using MoGe-2 (Wang et al., 2025c), which predicts metric-scale monocular geometry. We then project LiDAR points onto each image plane to obtain sparse metric samples that anchor the pseudo labels. To reconcile the two, we estimate a global scale and shift per image using ROE (Wang et al., 2025b), which down-weights outliers from occlusions and sensor noise and produces stable alignment. The full annotation procedure is described in Section B. The aligned pseudo depth $d^{\text{align}}$ serves as the supervision target for a DPT-style depth head. The depth head predicts $\hat{d}$ and is trained with a Charbonnier term and a gradient consistency regularizer that preserves

sharp boundaries:

$$\mathcal{L}_{\text{depth}} = \sqrt{(\hat{d} - d^{\text{align}})^2 + \epsilon^2} + \gamma_\nabla \sum_{a \in \{x,y\}} \sqrt{(\nabla_a \hat{d} - \nabla_a d^{\text{align}})^2 + \epsilon^2}. \tag{13}$$

Here $\epsilon = 0.001$ is the Charbonnier constant, $\gamma_\nabla$ weights the gradient-consistency term, and $\nabla_a$ denotes the discrete image gradient along direction $a \in \{x, y\}$. The summation aggregates horizontal and vertical components, and the loss is applied at valid pixels.

**Semantic Prediction.** We obtain semantic supervision by projecting LiDARSeg labels onto the image plane, which produces sparse yet precise per-pixel ground truth. The DPT decoder outputs per-pixel logits, and we use cross-entropy with an ignore mask so that only pixels covered by the LiDARSeg projection contribute to the loss. Let $\Omega_{\text{sem}}$ denote pixels whose labels are not 255.

$$\mathcal{L}_{\text{sem}} = \frac{1}{|\Omega_{\text{sem}}|} \sum_{p \in \Omega_{\text{sem}}} \text{CE}(\hat{S}_p, Y_p). \tag{14}$$

Here $\hat{S}$ are the logits from the decoder, $Y$ are sparse projected labels, and the ignore label masks pixels without supervision.

**Occupancy Prediction.** To further enhance the geometric and semantic expressiveness of our unified scene representation, we incorporate an additional occupancy prediction objective (Huang et al., 2023) during training. This task complements the image-space objectives by encouraging the scene tokens to reason about the 3D structure of the driving environment. Specifically, the scene tokens $B$ are decoded by a 3D occupancy head $\mathcal{D}_{\text{occ}}$ to predict voxel-wise semantic occupancy logits $\hat{O} \in \mathbb{R}^{X \times Y \times Z \times C}$. The occupancy prediction is optimized with a combination of cross-entropy and Lovász-Softmax losses:

$$\mathcal{L}_{\text{occ}} = \mathcal{L}_{\text{CE}}(\hat{O}, O) + \lambda_{\text{Lovász}} \mathcal{L}_{\text{Lovász}}(\hat{O}, O), \tag{15}$$

where $O$ denotes the ground-truth occupancy labels and $\lambda_{\text{Lovász}} = 0.2$. Jointly optimizing the occupancy and image-space tasks enables the scene tokens to capture richer 3D spatial and semantic relationships, thereby facilitating more holistic scene understanding.

**Semantic Regularization.** Conventional perception methods in autonomous driving apply supervision only on the model outputs, neglecting the semantic structure of the underlying scene representation. Apart from image-space and 3D occupancy supervision, we further regularize the scene tokens by aligning them with explicit semantic information to avoid structure corruption in the latent space. To elaborate, we first derive ground-truth semantic labels $Y_B$ for scene tokens by projecting semantic occupancy labels to the latent space with importance weighting for different semantic classes. We then regularize the scene-token semantic logits $\hat{S}_B$ with cross-entropy and Lovász-Softmax losses, using the same form as Equation (15) in the latent space:

$$\mathcal{L}_{\text{reg}} = \mathcal{L}_{\text{CE}}(\hat{S}_B, Y_B) + \lambda_{\text{Lovász}} \mathcal{L}_{\text{Lovász}}(\hat{S}_B, Y_B). \tag{16}$$

**Total Training Objective.** The overall training loss is defined in the main text. We set $\lambda_{\text{rgb}} = 10.0$ and $\lambda_{\text{occ}} = 5.0$, while $\lambda_{\text{depth}} = 0.2$, $\lambda_{\text{sem}} = 0.1$, and $\lambda_{\text{reg}} = 3.0$. We think the main contribution of our method is the formulation of this unified scene tokenizer, paving the way for new generative paradigms for autonomous driving. For some components (e.g., BEV queries), we employ simple yet effective designs to instantiate our formulation. We adopt new designs (e.g., visibility-aware attention) when existing techniques cannot well fit.

### A.5 Experimental Setup and Evaluation Protocol

**Dataset and supervision.** We train and evaluate on the nuScenes (Caesar et al., 2020) training and validation splits, respectively, using all six surround-view cameras with images resized to $256 \times 704$. We generate dense depth targets with MoGe-2 (Wang et al., 2025c) and align them to projected sparse LiDAR depth using the robust scale–shift fit described in Section B. We project nuScenes-LiDARSeg annotations onto each camera and assign the ignore label 255 to unlabeled pixels. For 3D occupancy supervision, we use semantic occupancy labels from SurroundOcc (Wei et al., 2023b).

**Model and training configuration.** The pretrained DINOv3-ViTB backbone (Siméoni et al., 2025) uses patch size 16 and produces a four-level feature pyramid with channel widths

Table 12: **Average runtime and GPU memory usage of each DriveTok module during forward inference.**

| Module | Avg. time (ms) | Peak GPU memory (MB) |
|---|---|---|
| Image Backbone | 13.56 | 3747.24 |
| FPN | 2.01 | 3771.46 |
| BEV Encoder | 72.39 | 4423.73 |
| Spatial-Aware Decoder | 225.85 | 6476.74 |
| RGB Head | 68.70 | 8456.09 |
| Depth Head | 69.72 | 8562.64 |
| Sem Head | 69.18 | 8652.19 |
| Occ Head | 150.35 | 7802.53 |
| **Scene Tokenization** | 87.96 | 4423.73 |
| **Total** | 671.76 | 8652.19 |

$\{768, 768, 768, 768\}$. A four-level FPN feeds the BEVFormer-style module, which uses a $128 \times 128$ BEV grid. The BEV region spans $[-51.2, -51.2, -5.0, 51.2, 51.2, 3.0]$ meters in the LiDAR coordinate frame, with eight vertical bins for visibility checks. The spatial-aware multi-view transformer follows a ViT-Base configuration; DPT-style decoders produce the image-space outputs, and a convolutional head predicts 3D occupancy. The model has approximately 280M trainable parameters. We optimize with AdamW (Loshchilov & Hutter, 2017) using a learning rate of $1 \times 10^{-4}$, weight decay of 0.01, and $\beta = (0.9, 0.999)$. We apply global gradient clipping at 35.0 and a cosine learning-rate schedule with warmup. Training runs for approximately 400k iterations on eight NVIDIA A800 GPUs using FlashAttention-2 (Dao, 2023) and BFloat16 precision.

**Evaluation protocol.** We report peak signal-to-noise ratio (PSNR) and structural similarity (SSIM) for image reconstruction. For depth, we project LiDAR points into each camera, mask pixels without LiDAR measurements, and report absolute relative error (AbsRel) and the fraction of valid pixels satisfying $\delta < 1.25$. The monocular methods in Table 2 are zero-shot references, whereas the surround-view methods in Table 3 are trained on nuScenes. We use sparse LiDARSeg labels to train semantic prediction and present qualitative results in Figure 4. Unlike approaches that use dense pseudo-labels from segmentation models such as SAM (Kirillov et al., 2023), this task is intended to enrich the scene tokens with semantic information. We evaluate occupancy on SurroundOcc (Wei et al., 2023b) with IoU and mIoU, following GaussianFormer (Huang et al., 2024b).

### A.6 Inference Time and Memory Usage of Modules

To evaluate the efficiency of DriveTok, we measured the average runtime and memory consumption of each module during forward inference on a single NVIDIA A800 80GB GPU, as shown in Table 12. Scene tokenization takes 87.96 ms per scene and reaches 4423.73 MB peak GPU memory.

## B Dense Depth Annotation

Public driving datasets such as nuScenes (Caesar et al., 2020) provide only sparse LiDAR depth, which is insufficient for supervising dense, per-pixel geometry. We obtain dense metric depth by combining monocular predictions with sparse LiDAR supervision. Specifically, MoGe-2 (Wang et al., 2025c) produces a dense pseudo depth map $\tilde{d}$. Sparse LiDAR points are projected onto the image plane to provide metric samples $d^{\text{lidar}}$. We then align $\tilde{d}$ to metric scale using a robust affine fit (ROE):

$$(a^\star, b^\star) = \arg\min_{a,b} \sum_{i \in \Omega} \rho\Big(a\tilde{d}_i + b - d_i^{\text{lidar}}\Big), \tag{17}$$

yielding the final aligned dense depth

$$d_i^{\text{align}} = a^\star \tilde{d}_i + b^\star. \tag{18}$$

This aligned depth serves as the supervision target for training.

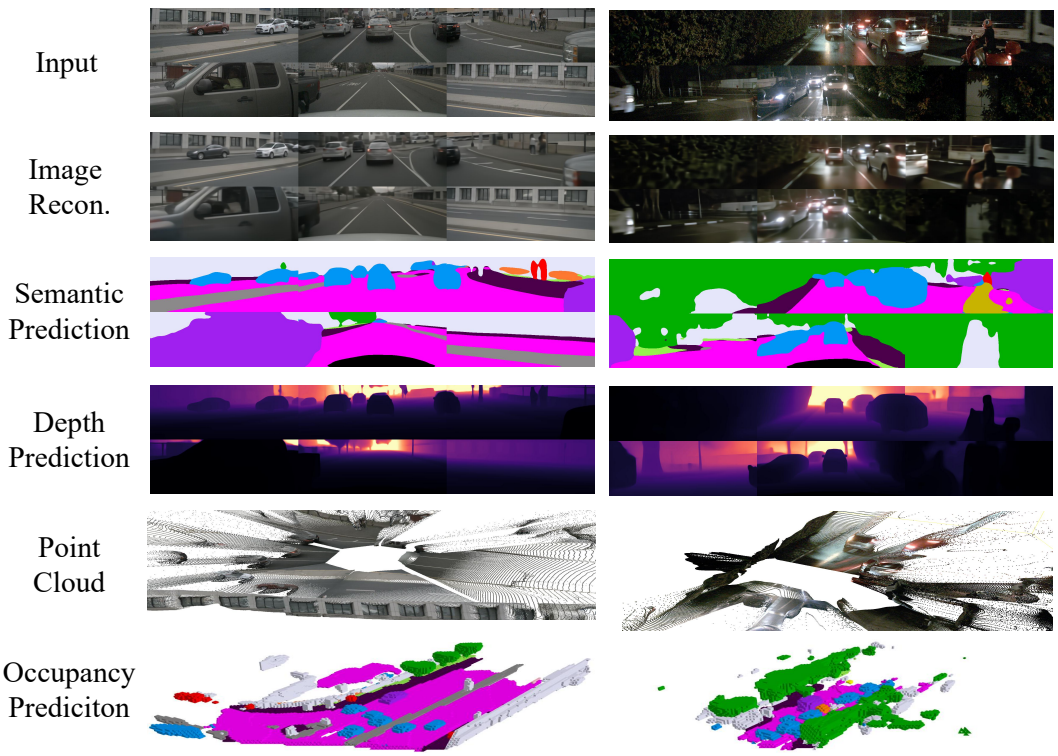



Figure 7: **Visualizations of DriveTok in different tasks.** Each column corresponds to a different autonomous driving scenario, including challenging night-time scenes.

Table 13: Backbone sensitivity. All listed baselines use ResNet-101. DriveTok is evaluated with ResNet-101 and DINOv3 at the lower input resolution of $256 \times 704$. Dashes denote metrics not reported for task-specific occupancy methods.

| Method | Backbone | Input | IoU↑ | mIoU↑ | PSNR↑ | SSIM↑ | AbsRel↓ | $\delta < 1.25$↑ |
|---|---|---|---|---|---|---|---|---|
| BEVFormer (Li et al., 2024) | ResNet-101 | $896 \times 1600$ | 30.50 | 16.75 | – | – | – | – |
| GaussianFormer (Huang et al., 2024b) | ResNet-101 | $896 \times 1600$ | 29.83 | 19.10 | – | – | – | – |
| GaussianFormer-2 (Huang et al., 2025) | ResNet-101 | $896 \times 1600$ | 30.56 | 20.02 | – | – | – | – |
| QuadricFormer (Zuo et al., 2025b) | ResNet-101 | $896 \times 1600$ | 31.22 | **20.12** | – | – | – | – |
| **DriveTok** | ResNet-101 | $256 \times 704$ | **32.23** | 18.83 | 25.94 | 0.698 | 0.100 | 0.880 |
| **DriveTok** | DINOv3 | $256 \times 704$ | **33.32** | **20.06** | **27.89** | **0.747** | **0.080** | **0.930** |

# C More Visualization Results

We now provide more visualization results, including challenging night-time scenarios as shown in Figure 7, to better demonstrate the robustness of our approach. We also provide a demo video in the supplementary material. These visualizations show that DriveTok can generate coherent 2D and 3D predictions across diverse tasks from our unified scene tokens.

# D Additional Ablations and Analysis

## D.1 Backbone Sensitivity

Table 13 replaces DINOv3 with ResNet-101 while retaining the DriveTok architecture. At $256 \times 704$, the ResNet-101 variant obtains the highest occupancy IoU among the listed ResNet-101 methods, although its mIoU is below the strongest specialized occupancy baselines. Replacing DINOv3 reduces all four DriveTok task metrics, with the largest relative change in AbsRel. The comparison separates the contribution of the scene-token architecture from the benefit of the foundation-model backbone: the former remains competitive under the conventional backbone, while the latter improves appearance, depth, and occupancy together.

### D.2 Semantic Occupancy on OpenScene

To complement the zero-shot PandaSet depth evaluation in Section 4.5, we train and evaluate occupancy models separately under the OpenScene protocol. On OpenScene, DriveTok improves occupancy mIoU from 27.10 to 39.50 over QuadricFormer under the same dataset protocol as shown in Table 14. Together, these evaluations test two distinct forms of generalization: zero-shot transfer of metric depth across camera rigs and occupancy learning on a larger dataset with a different annotation protocol.

Table 14: Semantic occupancy on OpenScene. Both methods are trained and evaluated with the same dataset protocol.

| Method | mIoU↑ |
|---|---|
| QuadricFormer (Zuo et al., 2025b) | 27.10 |
| **DriveTok** | **39.50** |