Generative Lane Topology Reasoning via Autoregressive Model with Geometry Prior

Jiahui Fu[1], Zehao Huang, Han Li[1], Naiyan Wang, and Si Liu[1]

Institute of Artificial Intelligence, Beihang University
{jiahuifu, lihan0620, liusi}@buaa.edu.cn
{zehaohuang18, winsty}@gmail.com

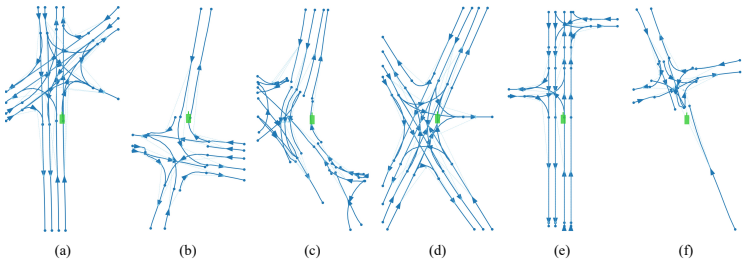

Fig. 1: Samples of predicted lane graphs. Given predicted lane graphs, human intuition can readily distinguish realistic structures from unrealistic ones without visual input. Test your intuition.[1]

Abstract. Lane topology reasoning aims to construct a lane graph from onboard sensor observations. Existing methods follow a detection and association paradigm that treats each lane instance independently, leading to geometric inconsistency at connected endpoints and incomplete graphs due to visual occlusions. To address these issues, we propose TopoGPT, a generative framework that learns the geometry prior from typical lane graph structures through autoregressive sequence modeling. Specifically, we construct a large-scale map dataset comprising 3.3M scenes. For each lane graph, a lane tokenizer serializes it into discrete tokens, while a scene context encoder converts it into a rasterized image and extracts global features as scene tokens. We pre-train an autoregressive lane sequence transformer via scene-conditioned next-token prediction, endowing the model with the geometry prior over lane graph structures. Building upon this prior, a perception adapter aligns BEV features from multi-view images with the pre-trained scene condition, transferring the learned geometry prior to sensor-based lane graph prediction. On the OpenLane-V2 benchmark, TopoGPT outperforms existing methods by an average of +6.4 on lane-level and +11.6 on point-level metrics, and produces geometrically consistent and structurally complete lane graphs. Our project page is available at `https://buaa-colalab.github.io/topogpt_page`.

Keywords: Autonomous driving · Lane Topology · Generative Models

[1] (a,d,e) from our predictions w/ geometry prior; (b,c,f) from methods w/o geometry prior.

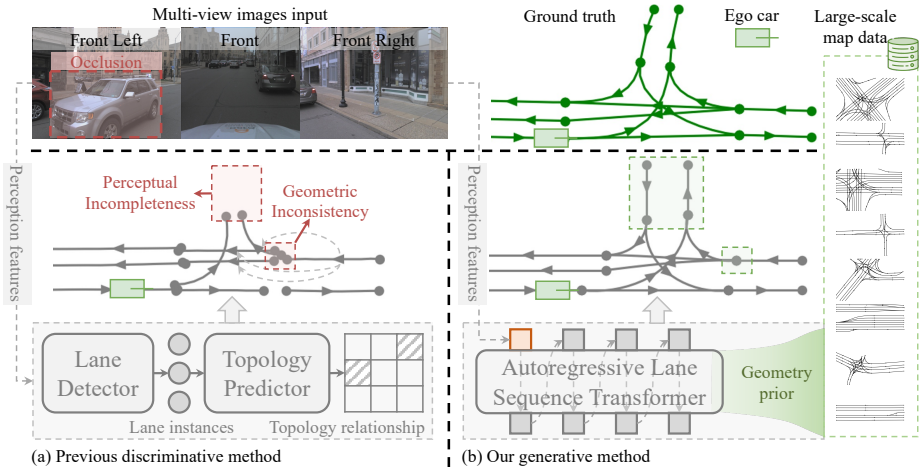

Fig. 2: Given multi-view images: (a) Previous discriminative methods suffer from endpoints misalignment and missed lanes under occlusion. (b) Our generative model learns the geometry prior from large-scale map data and predicts contiguous and complete lane graphs conditioned on image-derived perception tokens.

1 Introduction

Lane topology reasoning [43,56] constructs a vectorized lane graph from onboard sensor observations by jointly detecting centerlines and inferring their topological relations (*i.e.*, predecessor/successor). This converts raw perception into a structured, navigable representation for autonomous driving.

However, existing methods [9, 10, 24, 49] typically decouple lane detection from topology prediction in a first-detect-then-associate paradigm. As shown in Fig. 2 (a), a lane detector first predicts lane instances, after which a topology predictor estimates dense pairwise relationships. This two-stage discriminative design inevitably introduces two issues. (1) *Geometric Inconsistency*: Endpoints of topologically connected centerlines are often misaligned, as the decoupled topology stage cannot retroactively refine centerline geometry. (2) *Perceptual Incompleteness*: The lane graph is bounded by detector recall, which degrades under occlusion and limited sensing range. The subsequent association stage cannot recover missed instances, yielding a fragmented graph.

In contrast to generic object detection, where instances are often spatially independent, centerlines in a lane graph are governed by strict topological constraints inherent to road layout. As illustrated in Fig. 1, humans can readily judge whether a lane graph is structurally realistic even without surrounding camera images, owing to intrinsic geometry priors about common road structures. This motivates us to investigate whether a model can be endowed with a similar capability, enabling predictions that conform to such a prior distribution. To this end, we introduce the **G**enerative **P**re-trained **T**ransformer for Lane **Topo**logy (**TopoGPT**), which models lane graphs as a global structure rather

than independent instances. Leveraging millions of open-source map scenes, we pre-train TopoGPT to learn the geometry prior and capture global structures of lane graphs. Because map-only data does not require accurately synchronized images and map annotations, it can be collected from diverse HD map sources at a much larger scale than sensor-map paired data. Building upon this learned prior, we fine-tune the model to produce geometrically consistent and structurally complete predictions conditioned on online sensor observations.

As illustrated in Fig. 2 (b), we design an Autoregressive Lane Sequence Transformer (ALST) to learn the geometry prior over lane graphs via generative sequence modeling. To serialize a lane graph, a Bézier-based lane tokenizer converts each centerline into a group of discrete lane tokens. These groups are then concatenated according to a predefined spatial order, forming a complete lane token sequence that can directly reconstruct the original lane graph. Furthermore, we rasterize each lane graph into a pseudo-image and use a scene context encoder to extract global features as scene tokens. To support large-scale pre-training, we develop a dedicated data-processing pipeline that standardizes lane graphs from diverse data sources into a unified representation, resulting in a dataset of 3.3M scenes, on which we pre-train ALST via next lane token prediction conditioned on scene tokens. After pre-training, we fine-tune it on sensor-map paired data. In particular, a Bird's-Eye-View (BEV) encoder extracts BEV tokens from multi-view images, and a perception adapter projects them into the same condition token space as the scene tokens. Meanwhile, we apply LoRA-based fine-tuning to adapt ALST to BEV token conditioning. Benefiting from the geometry prior learned during pre-training, our method produces geometrically continuous connections between lane instances, allowing accurate topology prediction via a simple distance-based endpoint association criterion. Moreover, even under occlusion, since each lane token is generated conditioned on both perception tokens and preceding lane tokens, the model can infer reasonable topology in occluded regions to complete missing lanes consistent with the lane graph structure. We benchmark TopoGPT on the OpenLane-V2 [43] dataset against state-of-the-art discriminative methods. On subset A, TopoGPT consistently outperforms prior methods in spatial localization accuracy (reflected by +6.4 mean gain in the lane-level metric) and geometric structure realism (reflected by +11.6 mean gain in the point-level metric). Our contributions are as follows:

- We propose a generative lane topology reasoning framework that models lane topology as a global structure and predicts lane graphs autoregressively.
- We leverage large-scale map data to learn the geometry prior over lane graphs, which can be transferred to benefit perception.
- We achieve substantial improvements over previous discriminative methods, producing spatially accurate and structurally realistic lane graphs.

2 Related Work

Lane Topology Reasoning. Lane topology reasoning aims to detect lanes and infer their connectivity. Existing methods mainly fall into two paradigms.

(1) DETR-style frameworks [3, 7, 10, 12, 21–25, 31, 32, 39, 47, 49, 51] use learnable queries to detect lane instances and predict topological relations. Representative designs include interaction modeling by graph convolutional networks [24], distance-to-relation mapping [9], and endpoint-aware reasoning [10]. (2) Sequential graph generation methods [28, 29, 37, 50, 52] formulate lane topology construction as autoregressive generation. They either serialize lanes into a token sequence and generate it with a sequence-to-sequence transformer [28, 29], or incrementally grow the graph by adding vertices or edges [50]. Although these methods use sequence modeling, they focus on sophisticated graph traversal to determine the serialization order and predict lane sequences directly from perceptual inputs. In contrast, we focus on learning the geometry prior of lane graph structures from large-scale map data to facilitate lane topology reasoning.

Map Perception with Prior. Many studies incorporate prior information to enhance online map perception for autonomous driving. Most approaches derive priors from Standard Definition (SD) maps [19, 30, 34, 36, 38, 53, 54], satellite imagery [11, 53], and trajectories [17]. For instance, SMERF [30] encodes SD maps as polyline sequences and fuses map priors via cross-attention. SatForHDMap [11] introduces a hierarchical fusion module that refines onboard sensor features by aligning them with satellite representations. TrajTopo [17] leverages crowdsourced trajectories, representing them as rasterized heatmaps and vectorized instance tokens, and incorporates them as priors for online mapping. Unlike these methods that rely on extra inputs, we pre-train on large-scale map data to internalize a geometry prior into the model parameters.

Generative Model for Map. Generative models have been explored for map construction. VectorMapNet [27] uses a polyline generator to produce map elements. Several studies [8, 18, 20, 44, 55] leverage sensor inputs within a diffusion framework to generate rasterized maps. MapDiffusion [33] adopts diffusion to model the distribution of possible vectorized maps. LaneDiffusion [46] further employs diffusion models at the BEV feature level. These generative methods help enforce global structural consistency in map perception.

3 Method

3.1 Overview

Problem Formulation. Given surround-view images captured by multiple cameras mounted on a vehicle, the lane topology reasoning task aims to detect centerlines $\mathbf{L} = \{l_i \mid i = 1, 2, \ldots, M\}$ in the BEV space and reason about their topology relationships $\mathbf{E} \subseteq \mathbf{L} \times \mathbf{L}$. Each centerline l_i is represented as a directed ordered sequence of points $\mathbf{P} = [p_1, p_2, \ldots, p_K]$, where $p = (x, y) \in \mathbb{R}^2$ denotes a 2D coordinate in BEV space, and p_1, p_K are the starting and ending points, respectively. The topology relationship captures the connectivity among directed centerlines, forming a directed lane graph. An edge $(l_i, l_j) \in \mathbf{E}$ indicates that l_i is a predecessor of l_j.

Framework Overview. Unlike prior methods that decouple per-centerline detection from topology prediction, we note that lane graphs exhibit a strong

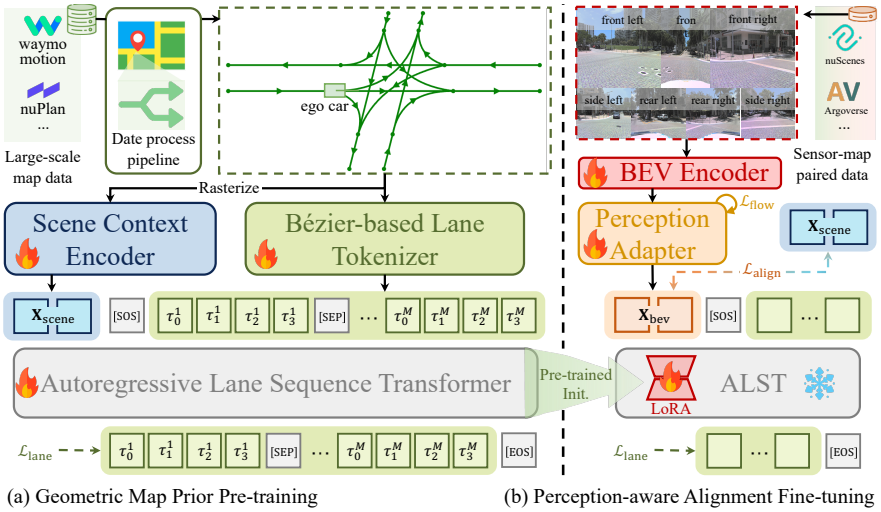

Fig. 3: Framework of TopoGPT. Built on an autoregressive lane sequence transformer, we first perform geometric map-prior pre-training on large-scale map data to learn the geometry prior, where lane token sequences are generated conditioned on scene tokens $\mathbf{X}_{\text{scene}}$. We then conduct perception-aware alignment fine-tuning on sensor-map paired data, initializing from the pre-trained weights to leverage the learned prior, where a BEV encoder extracts BEV tokens $\mathbf{X}_{\text{bev}}$ from multi-view images and aligns them with $\mathbf{X}_{\text{scene}}$, enabling lane graph prediction from sensor data.

geometry prior (*e.g.*, parallel adjacent lanes and regular intersection connectivity). We therefore aim to explicitly model and leverage this geometry prior to produce accurate centerline positions and reasonable topological relationships.

To this end, we employ a generative model with an Autoregressive (AR) architecture. Built upon this framework, we propose a two-stage training strategy, as illustrated in Fig. 3. In the first stage, *Geometric Map-Prior Pre-training*, we exploit scalable map-only data to learn the prior via autoregressive generative training. Concretely, each lane graph is simultaneously serialized into a lane token sequence and rasterized into a pseudo-image to extract scene tokens. The AR model is then trained to autoregressively generate the lane token sequence conditioned on scene tokens, capturing geometric regularities prevalent in real-world road scenes. Further details can be found in Sec. 3.2. In the second stage, *Perception-aware Alignment Fine-tuning*, we adapt the model to limited sensor-map paired data. Benefiting from the learned prior, fine-tuning reduces to a simpler mapping from perception features to the pre-trained condition space. Specifically, BEV features extracted from multi-view images are aligned with the scene tokens used during pre-training. The generative model then conditions on BEV features and produces lane graph predictions that are geometrically realistic and consistent with the perception input. We provide further details in Sec. 3.3. With this two-stage paradigm, we learn the geometry prior from

real-world lane graphs using a generative model. When conditioned on perceptual input, the model exploits the prior to generate precise and geometrically realistic lane predictions, enabling accurate topology reasoning through simple distance-based endpoint association.

3.2 Geometric Map-Prior Pre-training

To learn the geometry prior over lane graphs, we introduce *Geometric Map-Prior Pre-training*, which casts lane topology prediction as an autoregressive sequence generation problem. As illustrated in Fig. 3 (a), we first extract high-quality lane graphs from large-scale HD map sources via a dedicated processing pipeline. To bridge continuous geometry and discrete sequence, we propose a **Lane Tokenizer** that parameterizes each lane with Bézier curves and converts it into an ordered sequence of discrete lane tokens. To provide global context for generation, a **Scene Context Encoder** rasterizes the vectorized lanes, extracts features, and maps them to scene tokens. Finally, a decoder-only **Autoregressive Model** conditions on the scene tokens as a prefix and learns the joint distribution over lane token sequences via standard next-token prediction.

Data Collection. With the large-scale deployment of vehicles equipped with advanced driver-assistance systems, a substantial corpus of HD map data has been accumulated. These map-only sources are not constrained by the availability of synchronized multi-view images, making them substantially easier to scale than sensor-map paired datasets. We aim to learn the distribution over common real-world lane graph topologies from this corpus, which serves as a geometry prior for perception. Specifically, our data processing pipeline consists of three steps. (1) *Scene Retrieval*. We obtain raw map data from large-scale driving-log datasets with HD map annotations (*e.g.*, nuPlan [2] and the Waymo Open Motion Dataset [6]). To reduce geographic redundancy, we subsample ego-poses within each log using a distance threshold. Each sampled ego-pose defines the origin of a local coordinate, and we query centerline elements within a fixed radius from the HD map. (2) *Instance Merging*. Different datasets adopt heterogeneous definitions for HD map elements. Centerline instances may be segmented by semantic transitions (*e.g.*, changes in line-style attributes) or truncated at boundaries (*e.g.*, stop lines). To unify the representation, we aggregate lane instances according to the predecessor-successor relationships. Following SLEDGE [5], we merge adjacent lanes along non-branching paths, effectively removing intermediate instances with a single predecessor and a single successor. After merging, lanes are split only at topology bifurcations such as merges and forks. (3) *Geometric Sampling*. For each lane instance, we uniformly sample K points and use their (x, y) coordinates as the geometric representation $\mathbf{P} \in \mathbb{R}^{K\times 2}$.

Lane Tokenizer. To model the prior distribution of centerlines in map data with an AR model, we design a Bézier-based Lane Tokenizer that converts the centerlines of a scene into discrete lane token representations as shown in Fig. 4. To achieve a compact and unified representation of centerlines, for each centerline instance we fit a discrete coordinate sequence $\mathbf{P}$ with a cubic Bézier curve defined

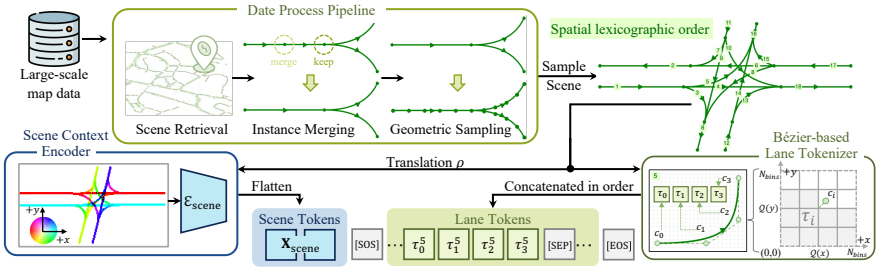

Fig. 4: Token sequence construction. We first build a unified lane graph from large-scale map data through a data processing pipeline. For each sampled scene, we (1) rasterize it and encode with a scene context encoder to obtain scene tokens, (2) convert each lane into a discrete group using a Bézier-based lane tokenizer and then sort all groups by a spatial lexicographical order to form the lane token sequence.

by four ordered control points $\mathbb{C} = \{c_0, c_1, c_2, c_3\}$. This parameterization maps the discrete control points onto a continuous curve of the form:

$$B(t, \mathbb{C}) = \sum_{i=0}^{n} \binom{n}{i} t^i (1-t)^{n-i} c_i, \quad t \in [0, 1], \; n = 3. \tag{1}$$

The control points are obtained by solving a least-squares problem that minimizes the discrepancy between the $B(t, \mathbb{C})$ and $\mathbf{P}$. The curve is strictly constrained to pass through the first and last control points, *i.e.*, $c_0 = p_1$ and $c_3 = p_K$. In this formulation, c_0 and c_3 are directly assigned as the endpoints (*i.e.*, start/end points), ensuring that the global topological structure and connectivity are preserved. The intermediate control points c_1 and c_2 serve as shape descriptors that capture the local geometry of the lane. We then quantize the continuous coordinates (x, y) of each control point into discrete vocabulary indices. Inspired by Pix2seq [4], we define a quantization function $\mathcal{Q}(v, v_{\max})$ that maps a continuous value $v \in [0, v_{\max}]$ to an integer in the range $[1, N_{\text{bins}}]$:

$$\mathcal{Q}(v, v_{\max}) = \left\lfloor \frac{v}{v_{\max}} \cdot (N_{\text{bins}} - 1) \right\rfloor + 1. \tag{2}$$

With this definition, we uniformly discretize both axes of the BEV range, with spatial extents X and Y respectively, into N_{bins} bins, so that four control points $\mathbb{C}$ can be represented as a sequence of discrete token indices as:

$$\mathbf{t} = \{\tau_i \mid \tau_i = \mathcal{Q}(y_i, Y) \cdot N_{\text{bins}} + \mathcal{Q}(x_i, X), \quad i \in \{0, 1, 2, 3\}\}. \tag{3}$$

Furthermore, to establish a deterministic ordering among centerlines, we apply a spatial lexicographic sort [40] that prioritizes the $[x_{\min}, y_{\min}, x_{\max}, y_{\max}]$ coordinates sequentially. This spatial sorting organizes lanes in canonical left-to-right (and then bottom-to-top) order. The discretized token group $\mathbf{t}^j$ of each centerline l_j is then arranged according to this order. We insert a separator token `<SEP>` between adjacent centerlines in the sorted order, and prepend/append

special tokens <SOS> and <EOS> to mark the beginning and end of the sequence, respectively. Through the above process, the centerlines can be converted into a discrete token sequence as:

$$\mathbf{T} = [\texttt{<SOS>},\ \ldots,\ \texttt{<SEP>},\ \underbrace{\tau_0^j, \tau_1^j, \tau_2^j, \tau_3^j}_{\mathbf{t}^j},\ \ldots,\ \texttt{<SEP>},\ \underbrace{\tau_0^M, \tau_1^M, \tau_2^M, \tau_3^M}_{\mathbf{t}^M},\ \texttt{<EOS>}], \tag{4}$$

where M denotes the number of centerlines in the scene, and τ_i^j represents the quantized token of the i-th control point of the j-th centerline.

Scene Context Encoder. Unconditional generation of complex lane topologies lacks explicit spatial constraints, making the model prone to drifting away from the scene distribution. To mitigate this issue, we introduce globally informative scene tokens as the condition. Motivated by the map encoder used in motion planners [5], we define a vector-to-raster translation ρ that converts the raw vectorized centerline set $\mathbf{L} \in \mathbb{R}^{M \times K \times 2}$ into a rasterized scene image $\mathbf{I}_l \in \mathbb{R}^{X_i \times Y_i \times 2}$. Specifically, for each pixel traversed by a polyline segment, we assign a two-dimensional direction vector $\Delta = [dx, dy]$ pointing to its successor point, while all remaining pixels are set to background values (*i.e.*, $[0, 0]$) as in Fig. 4. This translation preserves geometric orientation and topological connectivity of the lane graph. To extract compact features from the high-resolution rasterized image, we use a convolutional neural network encoder $\mathcal{E}_{\text{scene}}$, which downsamples $\mathbf{I}_l$ into a feature map $\mathbf{F}_{\text{scene}} \in \mathbb{R}^{X_f \times Y_f \times C}$. To bridge the modality gap between raster features and discrete tokens, we follow a standard vision-language alignment pipeline: we flatten $\mathbf{F}_{\text{scene}}$ along spatial dimensions and project it into the AR model hidden dimension D using a linear layer, followed by LayerNorm to stabilize feature scaling. The projected features form a scene token sequence $\mathbf{X}_{\text{scene}} \in \mathbb{R}^{N_{\text{scene}} \times D}$, which serve as prefix conditions, guiding the AR model to sequentially predict lane tokens $\mathbf{T}$. This process is formulated as:

$$\mathbf{X}_{\text{scene}} = \text{LN}\Big(\text{Proj}\Big(\text{Flatten}\big(\mathcal{E}_{\text{scene}}\big(\rho(\mathbf{L})\big)\big)\Big)\Big). \tag{5}$$

Autoregressive Model. We adopt an AR architecture similar to LLaMA-Gen [41]. Specifically, we reformulate structured lane topology reasoning as a sequence modeling problem and propose the Autoregressive Lane Sequence Transformer (ALST). ALST is based on a decoder-only Transformer [42] $\mathcal{G}_\theta$ with a standard causal mask. On the input side, the scene-token sequence $\mathbf{X}_{\text{scene}}$ is used as a prefix condition to provide scene-level cues for lane token generation, *i.e.*, $\mathbf{T} = \mathcal{G}_\theta(\mathbf{X}_{\text{scene}})$. Notably, we adopt Multimodal Rotary Position Embedding (M-RoPE) from Qwen2-VL [45] for scene tokens to preserve 2D spatial locality after flattening. During generation, the model predicts each next lane token conditioned on the scene context and all previously generated lane tokens. Formally, the conditional probability of the sequence $\mathbf{T}$ is factorized as:

$$p(\mathbf{T} \mid \mathbf{X}_{\text{scene}}) = \prod_{m=1}^{|\mathbf{T}|} p(\tau_m \mid \mathbf{X}_{\text{scene}},\ \tau_{<m}). \tag{6}$$

During training, we adopt a teacher forcing strategy and optimize both the scene context encoder $\mathcal{E}_{\text{scene}}$ and the AR decoder $\mathcal{G}_\theta$ end-to-end by minimizing the Cross-Entropy (CE) loss $\mathcal{L}_{\text{lane}}$ over the predicted lane token sequence.

Training Strategy. During pre-training, we aim for the model to not only generate scene-consistent centerlines, but also capture intrinsic geometry prior reflected in lane token sequences. To this end, we apply the following data augmentations. (1) *Global Augmentation.* To prevent overfitting and improve generalization, we apply the same geometric transformation to both the rasterized scene image and the vectorized centerlines, including random flipping, translation, and rotation. (2) *Condition Degradation.* To prevent the model from overrelying on the scene condition and neglecting inherent geometric structure, we randomly degrade the scene condition during training. Specifically, we randomly remove a subset of lanes from the rasterized scene image, or drop the scene condition entirely for some samples, while always keeping the full information in the lane token sequences. This completion objective trains the model to predict the full lane token sequence under degraded scene conditioning, thereby encouraging it to learn the lane graph topology.

3.3 Perception-aware Alignment Fine-tuning

To leverage the learned geometry prior for lane prediction from sensory inputs, we extract perceptual features through a **BEV Encoder** and align them with scene tokens used during pre-training via a **Perception Adapter** as shown in Fig. 5, thereby enabling lane prediction conditioned on multi-view images.

BEV Encoder. Given multi-view images as input, we first employ a ResNet-50 [15] backbone to extract multi-scale fused 2D feature maps from each view. Following [13], we adopt a bilinear sampling strategy for 2D-to-3D feature lifting. Specifically, a regular coordinate grid, which is spatially aligned with the rasterized scene image, is predefined in the target 3D space, and these 3D coordinates are back-projected onto the 2D feature maps of all cameras. Bilinear interpolation is performed at the projected locations to retrieve local image features, which are subsequently aggregated via weighted averaging to construct a unified 3D feature volume. The height dimension is then collapsed and folded into the channel dimension, yielding the BEV feature map $\mathbf{F}_{\text{bev}} \in \mathbb{R}^{X_f \times Y_f \times C}$.

Perception Adapter. We design a Perception Adapter based on noise-free flow matching to map the BEV features into the scene tokens learned during pre-training. Flow matching is a general generative modeling framework that learns a continuous transport trajectory from a source distribution to a target distribution by solving the ordinary differential equation $\frac{dz_t}{dt} = v_\theta(z_t, t)$, where z_t denotes the evolving sample at continuous time $t \in [0, 1]$ and v_θ is a neural network parameterized velocity field. Recent work [14,26] has demonstrated that, unlike diffusion models that typically assume a Gaussian source distribution, flow matching does not theoretically require a specific parametric form for source distribution. This flexibility enables the model to start from structured feature distributions and learn a direct transport path to the target distribution.

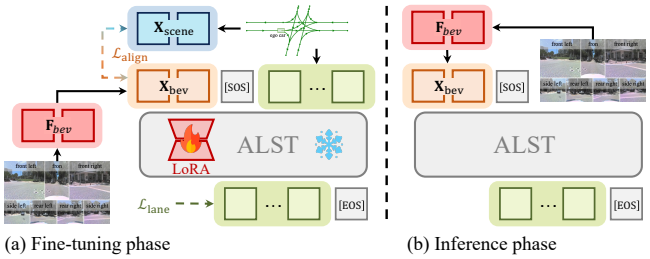

Fig. 5: (a) During fine-tuning, $\mathbf{X}_{\text{bev}}$ is optimized under alignment and lane prediction objectives. (b) During inference, $\mathbf{X}_{\text{bev}}$ is directly used as condition for lane prediction.

Building on this property, we propose a flow-matching-based adapter that evolves BEV features toward the pre-trained scene token distribution. We formulate the procedure as a cross-modality transport problem: the continuous BEV features extracted from multi-view images define the source distribution, while the target distribution is given by the scene tokens produced by a frozen pre-trained scene context encoder. Within the adapter, the v_θ, implemented as a lightweight DiT [35] model, is trained to approximate the ground-truth velocity that connects the source and target distributions along a linear interpolation path. Specifically, we optimize the flow-matching loss $\mathcal{L}_{\text{flow}}$, defined as the Mean Squared Error (MSE) between the predicted velocity and the ground-truth velocity. To mitigate the distribution shift in ALST conditioning from scene tokens during pre-training to BEV tokens during fine-tuning, we execute the flow-matching generation process during fine-tuning to obtain BEV-conditioned tokens. We further introduce a latent alignment loss $\mathcal{L}_{\text{align}}$ to supervise the adapter. During each forward pass, we explicitly unroll the numerical integration steps starting from the $\mathbf{F}_{\text{bev}}$ to obtain the predicted BEV tokens $\mathbf{X}_{\text{bev}} \in \mathbb{R}^{N_{\text{scene}} \times D}$, which are aligned with the corresponding scene tokens using a feature-level MSE loss $\mathcal{L}_{\text{align}}$. Moreover, $\mathbf{X}_{\text{bev}}$ is simultaneously fed to the subsequent ALST as the prefix condition. The CE loss $\mathcal{L}_{\text{lane}}$, identical to that used in pre-training, is computed on the predicted lane token sequence and backpropagated through the unrolled integration steps to update the adapter end-to-end.

Training. We jointly optimize the BEV encoder and the Perception Adapter, while employing Low-Rank Adaptation (LoRA) to fine-tune the pre-trained ALST so that it adapts to the BEV token inputs.

Inference. During inference, to ensure stable predictions, we start with the BEV features $\mathbf{F}_{\text{bev}}$ and obtain the BEV tokens $\mathbf{X}_{\text{bev}}$ through a deterministic ODE integration process (implemented by an Euler solver with discretization steps N_{flow}). Moreover, we use greedy decoding for autoregressive generation. To guarantee that the lane token sequence can be correctly decoded, we perform constrained decoding by forcing the `<SEP>` token at fixed periodic positions. For topology prediction, benefiting from the high geometric consistency at the endpoints of our predicted centerlines, we simply apply a distance threshold $\epsilon = 0.5\,\text{m}$ at endpoints to determine predecessor-successor relationships.

4 Experiments

4.1 Experimental Setup

Datasets. For the pre-training dataset, we extract HD map information from motion datasets including Waymo Motion [6], nuPlan [2], and Argoverse 2 Motion [48], constructing 2.07M, 1.01M, and 0.25M lane graph scenes respectively, yielding a total of 3.3M samples for pre-training. For the fine-tuning dataset, we evaluate our method on the OpenLane-V2 [43], which is divided into two subsets: *subset_A* and *subset_B*, each containing 1,000 scenes captured at 2 Hz with multi-view images and corresponding annotations. We utilize its annotations for lane centerline detection and inter-lane topology reasoning tasks.

Methods under Comparison. We select several recent DETR-based state-of-the-art methods for comparison, including TopoNet [24], TopoMLP [49], TopoLogic [9], and TopoPoint [10]. Notably, these methods typically output far more predictions than the actual number of lanes in a scene (*e.g.*, 200 or 300 candidates) along with associated confidence scores, whereas our method directly predicts all lanes in the scene. To ensure a fair comparison, we analyze the precision-recall curve as shown in Fig. 7 for each competing method and identify the operating point that maximizes the F1-score, using the corresponding predictions as the optimal output for that method.

Evaluation Metric. Based on the predicted centerline graph and the Ground Truth (GT), we employ two categories of evaluation metrics. (1) *Lane-level metrics*: adapted from [43], we report the Mean Precision (M-P) and Mean Recall (M-R) of centerline averaged over multiple distance thresholds, along with the Mean Topology (M-T) similarity between the predicted and the GT. (2) *Point-level metrics*: Following [1, 12, 16, 25], we convert the predicted centerlines and their inter-lane topology into a point-level graph. The structural similarity is then evaluated across two key dimensions: spatial alignment via Graph IoU (G-IoU), Split Detection Accuracy (SDA), and GEO F1 (G-F1), while structural fidelity is assessed using TOPO F1 (T-F1), JTOPO F1 (JT-F1), and Average Path Length Similarity (APLS).

Implementation Details. In the pre-training stage, we set the BEV range to $[\pm 52\,\mathrm{m}, \pm 32\,\mathrm{m}]$ along the X and Y axes respectively, and adopt $N_{\text{bins}} = 64$ to discretize the coordinate grid. The model is trained for 16 epochs on 8 NVIDIA H20 GPUs with a batch size of 128 per GPU. In the fine-tuning stage, we follow [24] and apply the same image-level data augmentation strategy. The model is trained for 24 epochs on 8 NVIDIA H20 GPUs with a batch size of 8 per GPU.

4.2 Quantitative Evaluation

As shown in Tab. 1, our method substantially outperforms existing approaches across multiple metrics on both subsets. On *subset_A*, regarding lane-level metrics, TopoGPT achieves an average +6.4 improvement, with more accurate centerline localization and shape prediction compared to the previous best results (**45.6** vs. 42.7 on M-P, **42.6** vs. 34.0 on M-R) and more reasonable topology

Table 1: Performance comparison on OpenLane-V2. We use their released model checkpoints when available; otherwise, we reproduce the results using the official codebases and adapt the outputs to our metric definitions.

Data	Method	lane-level metric↑			point-level metric↑					
		M-P	M-R	M-T	G-IoU	SDA	G-F1	T-F1	JT-F1	APLS
A	TopoNet [24]	39.7	32.3	10.1	38.8	7.40	45.6	32.3	21.7	9.80
	TopoMLP [49]	40.4	33.1	13.9	40.5	11.3	47.5	34.4	24.4	11.4
	TopoLogic [9]	40.5	33.6	15.5	39.0	11.5	45.8	32.0	21.7	10.4
	TopoPoint [10]	42.7	34.0	16.6	40.6	19.6	47.5	34.5	24.2	17.4
	TopoGPT (Our)	**45.6**	**42.6**	**24.4**	**54.4**	**26.9**	**60.3**	**47.4**	**35.7**	**28.7**
B	TopoNet [24]	33.4	26.3	6.27	31.5	4.20	36.3	24.3	18.1	6.40
	TopoMLP [49]	36.8	28.3	12.5	39.7	9.10	42.3	29.7	23.1	12.2
	TopoLogic [9]	33.9	27.2	11.7	36.7	8.60	40.1	28.0	21.4	10.3
	TopoPoint [10]	35.4	26.8	12.6	31.9	13.2	36.2	25.7	19.6	11.1
	TopoGPT (Our)	**43.1**	**39.2**	**23.4**	**55.4**	**25.8**	**58.9**	**46.3**	**36.8**	**29.5**

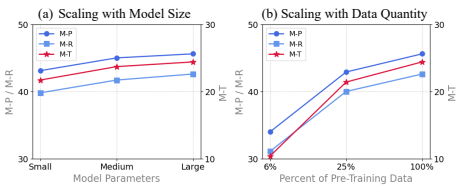

Fig. 6: Performance of the fine-tuned models under different pre-training model sizes and data quantities. As the number of model parameters and the percentage of pre-training data increase, performance consistently improves across all metrics.

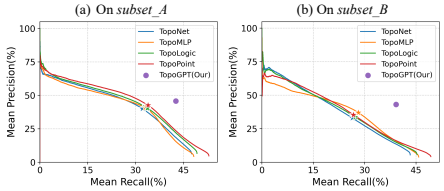

Fig. 7: Precision-Recall curve for lane-level metrics: we plot the Mean Precision *vs*. Mean Recall as the confidence threshold varies. Stars indicate the best F1-score operating points of previous methods, while the circle marks our method.

relationship (**24.4** vs. 16.6 on M-T). The notable +8.6 improvement on Recall further demonstrates that the learned geometry prior can reasonably complete the lane graph. Regarding point-level metrics, TopoGPT achieves an average +11.6 improvement, with more precise geometric structure prediction (**54.4** vs. 40.6 on G-IoU, **26.9** vs. 19.6 on SDA, **60.3** vs. 47.5 on G-F1) and more accurate connectivity inference (**47.4** vs. 34.5 on T-F1, **35.7** vs. 24.4 on JT-F1, **28.7** vs. 17.4 on APLS). On *subset_ B*, the improvements are equally significant.

4.3 Ablation Study

We conduct ablation studies on *subset_ A* and primarily report lane-level metrics.
Scalability of pre-training. During pre-training, TopoGPT demonstrates favorable scalability with respect to both model size and data quantity. (a) *Model size*: Fig. 6 (a) illustrates that as the model scales from Small (24.7M) to Medium (45.9M) to Large (91.3M) parameters, all evaluation metrics improve

Table 2: Ablation study on pre-training strategies, where **G.A.**=Global Augmentation, **C.D.**=Condition Degradation.

G.A.	**C.D.**	M-P↑	M-R↑	M-T↑
		41.0	38.4	21.1
✓		42.6	40.1	22.1
	✓	43.3	40.8	22.6
✓	✓	**45.6**	**42.6**	**24.4**

Table 3: Ablation study on fine-tuning schemes.

	M-P↑	M-R↑	M-T↑
From Scratch	7.4	7.0	0.8
Freeze	44.3	42.1	23.7
LoRA	**45.6**	**42.6**	**24.4**
Full Fine-tune	45.3	42.2	24.3

Table 4: Ablation study on fine-tuning loss designs. `w/o` flow denotes directly using the output features from adapter.

	$\mathcal{L}_{align}$	$\mathcal{L}_{lane}$	M-P↑	M-R↑	M-T↑
`w/o` flow	✓	✓	43.1	39.6	21.6
`w/` flow	✓		42.0	39.5	21.7
		✓	37.2	37.0	19.6
	✓	✓	**45.6**	**42.6**	**24.4**

Table 5: Ablation study on flow-matching steps.

N_{flow}	M-P↑	M-R↑	M-T↑
0	43.1	39.6	21.6
3	44.3	41.0	22.8
6	**45.6**	**42.6**	**24.4**
9	45.4	41.5	23.4

consistently. (b) *Data quantity*: Based on the TopoGPT-L model, the results in Fig. 6 (b) reveal that increasing the amount of pre-training data leads to steady performance gains of the final fine-tuned model, indicating that the geometry prior acquired during pre-training effectively transfers to and benefits the downstream fine-tuning stage.

Training Strategy in pre-training. Tab. 2 validates the effectiveness of the data augmentation strategies introduced in the pre-training stage. Incorporating Global Augmentation (G.A.) and Condition Degradation (C.D.) each independently enhances the model's ability to learn geometry prior, thereby improving various metrics after fine-tuning. Moreover, the two strategies exhibit a synergistic effect, jointly contributing +4.6/+4.2 improvements on M-P/M-R and a +3.3 gain on M-T.

Training Strategy in fine-tuning. We investigate how to leverage the pre-trained ALST weights during the fine-tuning stage. As shown in Tab. 3, *From Scratch* refers to training ALST to predict lane token sequences from BEV features without inheriting pre-trained weights. In this setting, the model struggles to learn such a complex mapping from limited sensor-map paired data, resulting in notably poor results. When inheriting the pre-trained weights, we explore three adaptation schemes: freezing all parameters, LoRA-based fine-tuning, and full-parameter fine-tuning, all of which achieve comparably high performance. We adopt LoRA by default due to its parameter efficiency. Overall, incorporating pre-training yields improvements of +38.2/+35.6/+23.6 on M-P/M-R/M-T, confirming the effectiveness of the geometry prior learned from pre-training.

Loss Terms in fine-tuning. Tab. 4 presents the contribution of each loss term during the fine-tuning. We first establish a baseline adapter without flow match-

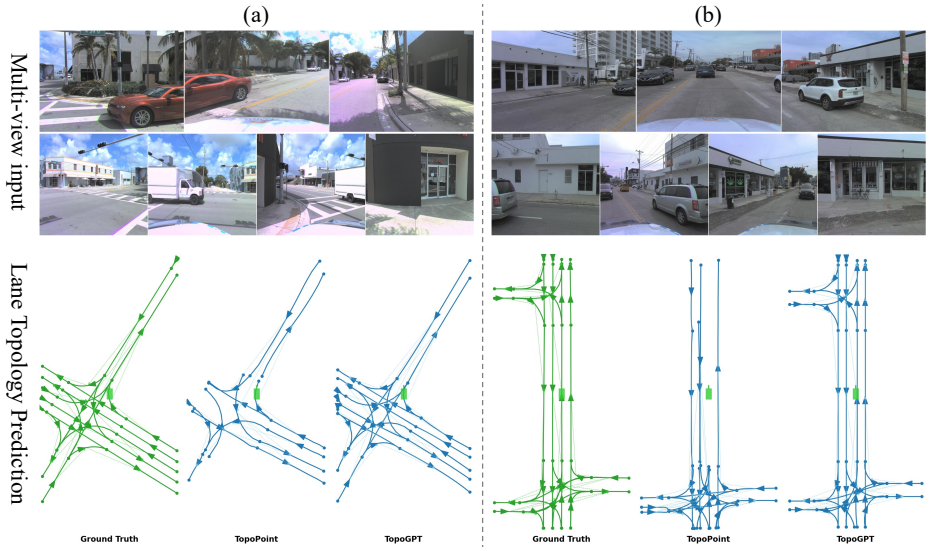

Fig. 8: Qualitative comparison between TopoPoint and our TopoGPT. The first row shows multi-view inputs, and the second row denotes the lane graphs. Dark solid arrows indicate centerlines, while light dashed arrows depict topological relations between centerlines.

ing, where the output features of the DiT model are directly used as $\mathbf{X}_{\text{bev}}$ and aligned with $\mathbf{X}_{\text{scene}}$. This approach already achieves competitive performance, and introducing flow matching to model the transition between the two token distributions further brings +2.5/+3.0/+2.8 improvements on M-P/M-R/M-T. Meanwhile, both $\mathcal{L}_{\text{align}}$ and $\mathcal{L}_{\text{lane}}$ serve critical roles. In particular, the notable accuracy drop upon removing $\mathcal{L}_{\text{lane}}$ underscores the importance of end-to-end optimizing BEV tokens through the lane token prediction loss.

Integration Steps in fine-tuning. Tab. 5 shows that the performance remains consistent across different integration steps N_{flow}. This validates the effectiveness of the flow-based perception adapter.

4.4 Qualitative Results

We present a qualitative comparison between TopoPoint and TopoGPT in Fig. 8. Specifically, we select two traffic scenarios for analysis and visualize the results of centerline detection and topology reasoning. In Fig. 8 (a), for a complex intersection, our TopoGPT produces a structurally complete and reasonably connected topology, whereas TopoPoint fails to predict necessary opposing and connecting centerlines. Meanwhile, the predictions of TopoGPT exhibit better geometric consistency at lane endpoints. In Fig. 8 (b), due to long-range perception limitations and partial occlusion, TopoPoint fails to detect the intersection directly ahead, while our TopoGPT is able to infer the centerlines and their connectivity at the intersection based on visual cues and existing lane structures. In

summary, by leveraging geometry prior from large-scale map data, TopoGPT is capable of predicting complete and coherent lane graphs for entire scenes under the guidance of visual perception.

5 Conclusion

In this paper, we present TopoGPT, a generative framework for lane topology reasoning that models lane graphs via autoregressive sequence prediction. Pre-training on 3.3M scenes of large-scale map data enables the model to learn the geometry prior of lane graphs, which are effectively transferred to perception-conditioned prediction via a flow-based perception adapter. Experiments on OpenLane-V2 demonstrate that TopoGPT substantially outperforms existing discriminative methods on both lane-level and point-level metrics, yielding lane graphs with improved geometric consistency and structural completeness.
Limitations. Our method relies on autoregressive generation, which introduces sequential decoding latency compared to parallel detection methods. Moreover, autoregressive decoding inherently suffers from the risk of error accumulation, where inaccuracies in earlier predictions may propagate to subsequent lane generations due to sequential dependencies. Although global scene context helps mitigate this issue, it cannot be completely eliminated in complex or highly occluded scenarios. Additionally, the geometry prior learned during pre-training may limit generalization to regions with uncommon road layouts.

Acknowledgements

This research is supported in part by the Key Research Program of Hangzhou (No. 2025SZD1A56), the National Natural Science Foundation of China (No. 6246-1160308, U23B2010, and 62576024), the Beijing Natural Science Foundation (No. L231011), the Fundamental Research Funds for the Central Universities (No. 501RCQD2025141003), BeiHang GanWei Project (No. 502GWXM20241410-01), the National Science Foundation Support Projects (No. 62425303), the National Key R&D Program of China (No. 2024YFb4707300), and Young Elite Scientist Sponsorship Program by Beijing Association for Science and Technology.

References

1. Büchner, M., Zürn, J., Todoran, I., Valada, A., Burgard, W.: Learning and aggregating lane graphs for urban automated driving. In: CVPR (2023)
2. Caesar, H., Kabzan, J., Tan, K.S., Fong, W.K., Wolff, E.M., Lang, A.H., Fletcher, L., Beijbom, O., Omari, S.: nuplan: A closed-loop ml-based planning benchmark for autonomous vehicles. arXiv preprint arXiv:2106.11810 (2021)
3. Can, Y.B., Liniger, A., Paudel, D.P., Van Gool, L.: Structured bird's-eye-view traffic scene understanding from onboard images. In: ICCV (2021)

4. Chen, T., Saxena, S., Li, L., Fleet, D.J., Hinton, G.E.: Pix2seq: A language modeling framework for object detection. In: ICLR (2022)
5. Chitta, K., Dauner, D., Geiger, A.: SLEDGE: synthesizing driving environments with generative models and rule-based traffic. In: ECCV (2024)
6. Ettinger, S., Cheng, S., Caine, B., Liu, C., Zhao, H., Pradhan, S., Chai, Y., Sapp, B., Qi, C.R., Zhou, Y., Yang, Z., Chouard, A., Sun, P., Ngiam, J., Vasudevan, V., McCauley, A., Shlens, J., Anguelov, D.: Large scale interactive motion forecasting for autonomous driving : The waymo open motion dataset. In: ICCV (2021)
7. Fu, J., Gao, C., Wang, Z., Yang, L., Wang, X., Mu, B., Liu, S.: Eliminating cross-modal conflicts in bev space for lidar-camera 3d object detection. In: ICRA (2024)
8. Fu, J., Gong, Y., Wang, L., Zhang, S., Zhou, X., Liu, S.: Generative map priors for collaborative bev semantic segmentation. In: CVPR (2025)
9. Fu, Y., Liao, W., Liu, X., Xu, H., Ma, Y., Zhang, Y., Dai, F.: Topologic: An interpretable pipeline for lane topology reasoning on driving scenes. In: NeurIPS (2024)
10. Fu, Y., Liu, X., Li, T., Ma, Y., Zhang, Y., Dai, F.: Topopoint: Enhance topology reasoning via endpoint detection in autonomous driving. In: NeurIPS (2025)
11. Gao, W., Fu, J., Shen, Y., Jing, H., Chen, S., Zheng, N.: Complementing onboard sensors with satellite maps: A new perspective for hd map construction. In: ICRA (2024)
12. Han, Y., Yu, K., Li, Z.: Continuity preserving online centerline graph learning. In: ECCV (2024)
13. Harley, A.W., Fang, Z., Li, J., Ambrus, R., Fragkiadaki, K.: Simple-bev: What really matters for multi-sensor bev perception? In: ICRA (2023)
14. He, J., Yu, Q., Liu, Q., Chen, L.C.: Flowtok: Flowing seamlessly across text and image tokens. In: ICCV (2025)
15. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: CVPR (2016)
16. He, S., Balakrishnan, H.: Lane-level street map extraction from aerial imagery. In: WACV (2022)
17. Jia, P., Luo, Z., Wen, T., Yang, M., Jiang, K., Cui, L., Yang, D.: Enhancing lane segment perception and topology reasoning with crowdsourcing trajectory priors. IEEE RAL (2025)
18. Jia, P., Wen, T., Luo, Z., Yang, M., Jiang, K., Liu, Z., Tang, X., Lei, Z., Cui, L., Zhang, B., et al.: Diffmap: Enhancing map segmentation with map prior using diffusion model. IEEE RAL (2024)
19. Jiang, Z., Zhu, Z., Li, P., Gao, H., Yuan, T., Shi, Y., Zhao, H., Zhao, H.: P-mapnet: Far-seeing map generator enhanced by both sdmap and hdmap priors. IEEE RAL (2024)
20. Le, D.T., Shi, H., Cai, J., Rezatofighi, H.: Diffusion model for robust multi-sensor fusion in 3d object detection and bev segmentation. In: ECCV (2024)
21. Li, H., Gao, Y., Liu, S., Wang, Y., Liu, B., Mu, B.: Unified modeling of lane and lane topology for driving scene reasoning. IEEE TCSVT (2026)
22. Li, H., Huang, S., Xu, L., Gao, Y., Mu, B., Liu, S.: Ratopo: Improving lane topology reasoning via redundancy assignment. In: ACM MM (2025)
23. Li, H., Huang, Z., Wang, Z., Rong, W., Wang, N., Liu, S.: Enhancing 3d lane detection and topology reasoning with 2d lane priors. arXiv preprint arXiv:2406.03105 (2024)
24. Li, T., Chen, L., Wang, H., Li, Y., Yang, J., Geng, X., Jiang, S., Wang, Y., Xu, H., Xu, C., et al.: Graph-based topology reasoning for driving scenes. arXiv preprint arXiv:2304.05277 (2023)

25. Liao, B., Chen, S., Jiang, B., Cheng, T., Zhang, Q., Liu, W., Huang, C., Wang, X.: Lane graph as path: Continuity-preserving path-wise modeling for online lane graph construction. In: ECCV (2024)
26. Liu, Q., Yin, X., Yuille, A.L., Brown, A., Singh, M.: Flowing from words to pixels: A noise-free framework for cross-modality evolution. In: CVPR (2025)
27. Liu, Y., Yuan, T., Wang, Y., Wang, Y., Zhao, H.: Vectormapnet: End-to-end vectorized hd map learning. In: ICML (2023)
28. Lu, J., Nie, M., Zhang, B., Peng, R., Cai, X., Xu, H., Li, H., Wen, F., Zhang, W., Zhang, L.: Translating images to road network: A sequence-to-sequence perspective. IEEE TPAMI (2025)
29. Lu, J., Peng, R., Cai, X., Xu, H., Li, H., Wen, F., Zhang, W., Zhang, L.: Translating images to road network: A non-autoregressive sequence-to-sequence approach. In: ICCV (2023)
30. Luo, K.Z., Weng, X., Wang, Y., Wu, S., Li, J., Weinberger, K.Q., Wang, Y., Pavone, M.: Augmenting lane perception and topology understanding with standard definition navigation maps. In: ICRA (2024)
31. Lv, C., Qi, M., Liu, L., Ma, H.: T2sg: Traffic topology scene graph for topology reasoning in autonomous driving. In: CVPR (2025)
32. Ma, Z., Liang, S., Wen, Y., Lu, W., Wan, G.: Roadpainter: Points are ideal navigators for topology transformer. In: ECCV (2024)
33. Monninger, T., Zhang, Z., Mo, Z., Anwar, M.Z., Staab, S., Ding, S.: Mapdiffusion: Generative diffusion for vectorized online hd map construction and uncertainty estimation in autonomous driving. In: IROS (2025)
34. Ort, T., Walls, J.M., Parkison, S.A., Gilitschenski, I., Rus, D.: Maplite 2.0: Online hd map inference using a prior sd map. IEEE RAL (2022)
35. Peebles, W., Xie, S.: Scalable diffusion models with transformers. In: ICCV (2023)
36. Pei, M., Shan, J., Li, P., Shi, J., Huo, J., Gao, Y., Shen, S.: SEPT: standard-definition map enhanced scene perception and topology reasoning for autonomous driving. IEEE RAL (2025)
37. Peng, R., Cai, X., Xu, H., Lu, J., Wen, F., Zhang, W., Zhang, L.: Lanegraph2seq: Lane topology extraction with language model via vertex-edge encoding and connectivity enhancement. In: AAAI (2024)
38. Pham, K.S., Witte, C., Behley, J., Betz, J., Stachniss, C.: Coherent online road topology estimation and reasoning with standard-definition maps. In: IROS (2025)
39. Rong, F., Peng, W., Lan, M., Zhang, Q., Zhang, L.: Driving scene understanding with traffic scene-assisted topology graph transformer. In: ACM MM (2024)
40. Rowe, L., Girgis, R., Gosselin, A., Paull, L., Pal, C., Heide, F.: Scenario dreamer: Vectorized latent diffusion for generating driving simulation environments. In: CVPR (2025)
41. Sun, P., Jiang, Y., Chen, S., Zhang, S., Peng, B., Luo, P., Yuan, Z.: Autoregressive model beats diffusion: Llama for scalable image generation. arXiv preprint arXiv:2406.06525 (2024)
42. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł., Polosukhin, I.: Attention is all you need. In: NeurIPS (2017)
43. Wang, H., Li, T., Li, Y., Chen, L., Sima, C., Liu, Z., Wang, B., Jia, P., Wang, Y., Jiang, S., Wen, F., Xu, H., Luo, P., Yan, J., Zhang, W., Li, H.: Openlane-v2: A topology reasoning benchmark for unified 3d HD mapping. In: NeurIPS (2023)
44. Wang, L., Zhao, Y., Zhang, Z., Feng, J., Liu, S., Kang, B.: Image understanding makes for a good tokenizer for image generation. In: NeurIPS (2024)

45. Wang, P., Bai, S., Tan, S., Wang, S., Fan, Z., Bai, J., Chen, K., Liu, X., Wang, J., Ge, W., et al.: Qwen2-vl: Enhancing vision-language model's perception of the world at any resolution. arXiv preprint arXiv:2409.12191 (2024)
46. Wang, Z., Zhang, W., Zhang, W., Tan, X., Liu, H., Wang, Y., Li, G.: Lanediffusion: Improving centerline graph learning via prior injected BEV feature generation. In: ICCV (2025)
47. Wang, Z., Huang, Z., Fu, J., Wang, N., Liu, S.: Object as query: Lifting any 2d object detector to 3d detection. In: ICCV (2023)
48. Wilson, B., Qi, W., Agarwal, T., Lambert, J., Singh, J., Khandelwal, S., Pan, B., Kumar, R., Hartnett, A., Pontes, J.K., et al.: Argoverse 2: Next generation datasets for self-driving perception and forecasting. arXiv preprint arXiv:2301.00493 (2023)
49. Wu, D., Chang, J., Jia, F., Liu, Y., Wang, T., Shen, J.: Topomlp: A simple yet strong pipeline for driving topology reasoning. In: ICLR (2024)
50. Xie, M., Zeng, S., Chang, X., Liu, X., Pan, Z., Xu, M., Wei, X.: Seqgrowgraph: Learning lane topology as a chain of graph expansions. In: ICCV (2025)
51. Xu, Z., Liu, Y., Sun, Y., Liu, M., Wang, L.: Centerlinedet: Centerline graph detection for road lanes with vehicle-mounted sensors by transformer for hd map generation. In: ICRA (2023)
52. Yang, Y., Luo, Y., He, B., Li, E., Cao, Z., Zheng, C., Mei, S., Li, Z.: Topo2seq: Enhanced topology reasoning via topology sequence learning. In: AAAI (2025)
53. Ye, J., Paz, D., Zhang, H., Guo, Y., Huang, X., Christensen, H.I., Wang, Y., Ren, L.: Smart: Advancing scalable map priors for driving topology reasoning. In: ICRA (2025)
54. Zeng, S., Chang, X., Liu, X., Yuan, Y., Liang, S., Pan, Z., Xu, M., Wei, X.: Priordrive: Enhancing online hd mapping with unified vector priors. arXiv preprint arXiv:2409.05352 (2024)
55. Zhu, X., Zyrianov, V., Liu, Z., Wang, S.: Mapprior: Bird's-eye view map layout estimation with generative models. In: ICCV (2023)
56. Zürn, J., Vertens, J., Burgard, W.: Lane graph estimation for scene understanding in urban driving. IEEE RAL (2021)