

# Coherent Online Road Topology Estimation and Reasoning with Standard-Definition Maps

Khanh Son Pham\*

Christian Witte\*

Jens Behley

Johannes Betz

Cyrill Stachniss

**Abstract**—Most autonomous cars rely on the availability of high-definition (HD) maps. Current research aims to address this constraint by directly predicting HD map elements from onboard sensors and reasoning about the relationships between the predicted map and traffic elements. Despite recent advancements, the coherent online construction of HD maps remains a challenging endeavor, as it necessitates modeling the high complexity of road topologies in a unified and consistent manner. To address this challenge, we propose a coherent approach to predict lane segments and their corresponding topology, as well as road boundaries, all by leveraging prior map information represented by commonly available standard-definition (SD) maps. We propose a network architecture, which leverages hybrid lane segment encodings comprising prior information and denoising techniques to enhance training stability and performance. Furthermore, we facilitate past frames for temporal consistency. Our experimental evaluation demonstrates that our approach outperforms previous methods by a large margin, highlighting the benefits of our modeling scheme.

## I. INTRODUCTION

As a prevailing robotic application, the operation of autonomous vehicles in urban environments is contingent upon the availability of high-definition (HD) maps of the surrounding environment. These HD maps should contain precise geometric information about the drivable lanes, including their correspondences with one another, and their relationship to other important traffic elements, such as traffic lights and signs. The generation of such HD maps is a time-consuming and resource-intensive semi-automated process that involves manual steps. At the same time, these maps are subject to changes over time [15]. For safe and robust autonomous driving, the detection of traffic elements such as traffic lights and traffic signs and their association with lanes or lane segments is crucial. For example, traffic light states or prohibitive traffic signs contain important information for predicting the future trajectories of other traffic participants.

Recent approaches [34], [56] have demonstrated encouraging results in the detection of HD map elements. However, they do not achieve the objective of computing consistent and accurate HD maps online while driving. The latest approaches reformulate map element detection into a lane segment formulation [22] to leverage the known geometric properties of lane layouts and further predict the topological

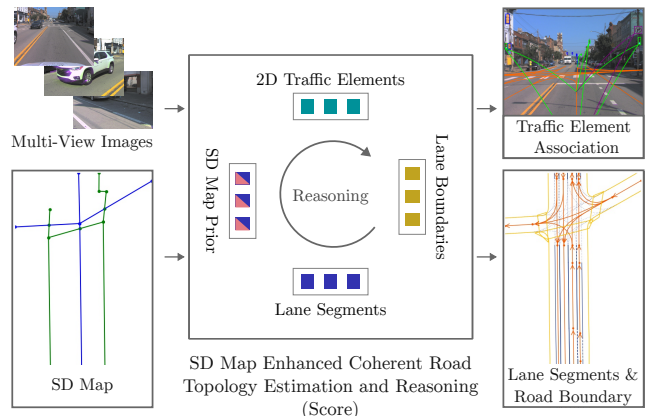


Fig. 1: Given multi-view images and a standard-definition (SD) map, our method predicts lane segments and lane boundaries, as well as traffic elements. It reasons about the topology among lane segments and estimates the traffic element-to-lane association.

relationship of lane segments or centerlines [21]. Other work [37] seeks to integrate prior knowledge about road geometry into their architectural design by incorporating standard-definition (SD) maps into the bird's eye view (BEV) transformation.

None of the prior approaches have adequately addressed the challenge of jointly detecting lane segments, their topological relationships, while exploiting available a-priori information of SD maps. For this reason, the CVPR 2024 Grand Autonomous Challenge, with its *Mapless Driving* track, proposed a problem formulation for solving these tasks simultaneously.

To this end, we approach to solve the union of tasks coherently, that is, to consider them as a whole with clear interfaces. Thus, we systematically design a network architecture to predict lane segments and their topological relationships by leveraging latent embeddings (queries), as depicted in Fig. 1. It encompasses the topology among lane segments, as well as the topology between lane segments and traffic elements (e.g., traffic lights or traffic signs). Given an SD map, we ingest prior information as enhanced hybrid queries with a positional sampling module. This provides the network hypotheses of potential lane segments, which can be iteratively validated and refined by interacting with the BEV features. We introduce temporal fusion and lane denoising for more consistent and robust predictions, while we optimize the overall performance by implementing further improvements to the architecture. The experimental results demonstrate the efficacy of the proposed method,

\* Equal contribution. S. Pham is with CARIAD SE and the Technical University Munich, Germany. C. Witte is with CARIAD SE and with the Center for Robotics, University of Bonn, Germany. J. Betz is with the Technical University Munich, Germany and the Munich Institute of Robotics and Machine Intelligence (MIRMI), Germany. J. Behley, C. Stachniss are with the Center for Robotics, University of Bonn, Germany. C. Stachniss is also with the Lamarr Institute for Machine Learning and Artificial Intelligence, Germany.

which reaches state-of-the-art performance on the OpenLane-V2 [44] dataset.

Additional qualitative results can be found at: <https://www.ipb.uni-bonn.de/html/projects/score>.

## II. RELATED WORK

To mitigate the reliance on expensive high-definition maps, recent works have focused on deriving vectorized representation of HD map elements as polylines. This representation is commonly predicted in bird’s eye view (BEV) space from onboard sensors.

**Bird’s Eye View Transformation.** Using a categorical depth distribution, Lift-Splat-Shoot (LSS) [40] and their derivative works [13], [23], [36] project features onto a 3D volume and then collapse the volume onto a 2D plane with efficient pooling. Transformer-based approaches [5], [24], [35], [47], [59] cross-attend BEV queries to images features and thus extract a BEV representation.

**HD Map Generation.** Leveraging semantic segmentation, HDMapNet [20] utilizes a post-processing step to output a polyline representation of map elements. In a two-stage approach, VectorMapNet [34] first identifies map elements and then outputs polylines using a detection transformer [3]. MapTR [26] proposes to directly predict polylines in a single-stage approach by employing hierarchical queries, while the same authors introduce one-to-many assignment, decoupled attention, and auxiliary losses in their follow-up work [27]. Further improvements to MapTR include predicting pivotal points [8], modeling the output as Bézier curves [41], and incorporating geometric properties into the learning process [55], [58]. Another line of work puts emphasis on the query generation, e.g., by leveraging instance [33] or semantic [6] segmentation masks or by introducing hybrid queries [52], [60].

**Topology Estimation.** Understanding road topology is crucial for scene comprehension in autonomous driving, which includes predicting lane centerlines and their topology, i.e., constructing a lane graph with successor-predecessor relationship. Can et al. [2] perform the graph estimation for single monocular images by associating and merging piecewise centerline estimates based on a predictions of a transformer-based model. Building on this, TopoRoad [51] introduces minimal cycle queries to maintain the correct order of intersections. LaneGAP [25] uses a heuristic algorithm to reconstruct the graph from a set of lanes. TopoNet [21] transforms 2D detections for traffic elements into a unified feature space to model the relationship with a scene graph neural network, while in TopoMLP [48], the authors emphasize the importance of a strong 2D detection performance. Other works combine trajectory knowledge and visual SLAM to estimate the lane information [39] or extend the topology estimation idea by leveraging geometric properties [9] or queries initialized by 2D priors [19]. Li et al. [22] propose to predict lane segments, each consisting of semantic meaningful lane entities defined by a lane centerline, lane boundaries, as well as successor relationship and class information.

**SD Map-aided Map Generation.** To improve upon the lane-lane relationship estimation, SMERF [37] integrates prior information in form of standard-definition maps into the network architecture by encoding the SD map elements with a transformer encoder and by ingesting the feature representation into the BEV transformation process. Other approaches propose to fuse the SD map information in a rasterized fashion [38] or to ingest the knowledge at different stages of the network [53], [57].

**Temporal Fusion.** To achieve temporal consistency, previous works [10], [12], [24] have shown various methods on fusing past knowledge with the current observations. One line of work cross-attends to image features of previous timesteps by back-projecting reference points and aggregating features for semantic segmentation or 3D object detection [24], [31], while other methods [12] stack the current BEV features with motion-compensated BEV features from the previous timestep. Further work recurrently fuses BEV features [10] or propagates high confidence queries to the next timestep [32]. StreamMapNet [56] introduces the idea of streaming temporal fusion to HD map element detection by recurrently fusing BEV features and propagating past queries. Wang et al. [45] also apply denoising to enhance lane queries for temporal fusion.

**Mapless Driving.** The CVPR 2024 Grand Autonomous Challenge introduced the joint tasks of lane segment detection, road boundary estimation, and traffic element association with SD map information. Participating works [49], [54] address this joint formulation by integrating the prior SD map information similar to [37], and use large backbones and ensembles to improve performance.

## III. SD MAP ENHANCED COHERENT ROAD TOPOLOGY ESTIMATION AND REASONING

We aim to predict HD map elements and their corresponding relationships based on onboard multi-view images. To this end, we propose SD map enhanced coherent road topology estimation and reasoning, called **Score**, which jointly predicts lane segments, road boundaries, and traffic elements such as traffic lights or traffic signs. Further, our method derives an association between the traffic elements and the lane segments, as well as the topology among the lane segments.

Given multi-view images, the image backbone outputs multi-scale image feature maps  $\mathcal{F}_{PV} = \{\mathbf{F}_{PV}^i\}$  for the  $i$ -th camera, while top-view features  $\mathbf{F}_{BEV}$  are obtained by a BEV transformation [24]. To leverage the geometric constraints of road lanes, we utilize the lane segment formulation [44]. A lane segment (ls) is defined as a set of polylines, representing the centerline and lane boundaries, along with a classification label indicating the type of lane segment and lane boundary types (e.g., solid, dashed, or invisible). Further, we consider pedestrian crossing as lane segments.

As a separate task, we infer road boundaries (rb) as polylines with  $n_{rb}$  points, while the traffic elements (te) are predicted in 2D with bounding box annotation. We denote the topology among lane segments as lane graph ( $\mathbf{V}_{ls}, \mathbf{A}_{lsls}$ )

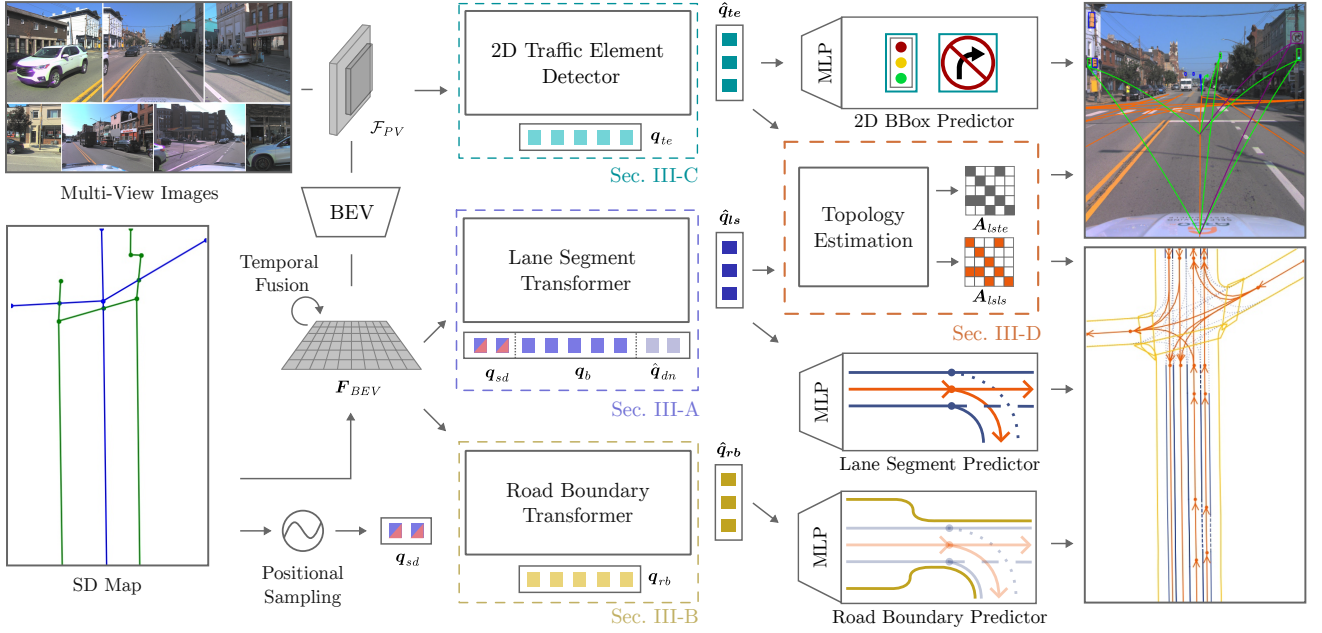


Fig. 2: Overall architecture of our approach. First, multi-view image features are transformed into a BEV representation. Utilizing SD map prior information, we enhance learnable queries by a positional sampling and add lane denoising queries. In the *Lane Segment Transformer*, lane segment queries (blue) are refined with BEV features, while two further transformer decoder output refined road boundary queries (yellow) and 2D traffic element queries (cyan). In the *Topology Estimation* module, these queries interact to derive the lane-to-lane and traffic element-to-lane association. Simple MLPs are employed to decode the queries  $\hat{q}$  into the corresponding output format.

with  $V_{ls}$  as the set of lane segments and the edge set  $A_{lsls}$  as an adjacency matrix [21], [22]. Similarly, we denote the correspondence between lane segment and traffic elements as a bipartite graph  $(V_{ls} \cup V_{te}, A_{lsste})$ . The SD map is provided as a set of polylines with class labels (e.g., road, sidewalk).

An overview of our proposed architecture Score is illustrated in Fig. 2. The network is composed of multiple components, which utilize queries as interfaces. The lane segment transformer (Sec. III-A) and road boundary transformer (Sec. III-B) use the BEV features to output instance-level map element embeddings. The 2D traffic element detector outputs traffic sign and traffic light detections based on 2D image features (Sec. III-C). In the topology estimation (Sec. III-D), the network derives adjacencies for the lane-lane and lane-traffic element topology.

#### A. Lane Segment Transformer

To infer lane segments, our model utilizes a similar architecture to LaneSegNet [22], incorporating a transformer-based decoder. In the transformer decoder, the hybrid queries  $q_{ls}$  cross-attend to the enriched BEV features  $F_{BEV}$ , which are then refined iteratively using self-attention [42] and deformable attention [50]. To address the limitations of deformable attention in capturing long-range dependencies, particularly for elongated lane shapes, we integrate lane attention, a lane segment-aware deformable attention [22]. This module distributes multiple reference points along the estimated lane boundaries of the previous transformer decoder layer. We enhance the lane decoding by leveraging hybrid queries and condition the decoding on the prior

knowledge in form of reference points. Then, the refined queries  $\hat{q}_{ls}$  are calculated as:

$$\hat{q}_{ls} = \text{LaneDecoder}(F_{BEV}, q_{ls} | p_{ls}) \in \mathbb{R}^{N_{ls} \times C}, \quad (1)$$

with  $q_{ls} = \text{concat}(q_{sd}, q_b, \hat{q}_{dn})$  comprise of SD map enhanced queries  $q_{sd}$ , denoising queries  $\hat{q}_{dn}$ , both of which will be explained in the following sections, and the base queries  $q_b$ .  $p_{ls} = \text{concat}(\hat{p}_{sd}, p_b, \hat{p}_{dn})$  denote their respective reference points and  $C$  is the channel dimension.

1) *SD Map Enhanced Queries*: Originally [22], for each query  $q_b$ , a randomly initialized reference point is generated, which has no prior driving scene information. To improve the initial placement, we propose incorporating geometric data from the SD map, thus, leveraging prior information as potential lane segment hypotheses.

For that, we define the set of  $n_{sd}$  queries  $q_{sd} \in \mathbb{R}^{n_{sd} \times C}$  and generate reference points for these queries by sampling one point from the middle of each edge and continue by sampling more points proportionally based on edge length until we reach  $n_{sd}$  points. This ensures evenly-spaced sampling and prevents duplicative information propagation. For the  $i$ -th SD map query  $q_{sd,i}$ , the initial reference point placement  $\hat{p}_{sd,i}$  is computed as follows:

$$\hat{p}_{sd,i} = \text{MLP}(q_{pos,i}) + p_{sd,i}, \quad (2)$$

where  $q_{pos,i}$  denotes the positional encoding of the  $i$ -th query and  $p_{sd,i}$  refers to the sampled reference points, which are normalized to  $[0, 1]$ . This formulation allows for both, a learnable and an a-priori term, providing the network further flexibility. Lastly, a self-attention mask between SD map

enhanced queries  $q_{sd}$  and other queries prevents the leakage of deficient information, as an SD map can have annotation errors such as missing or wrong map elements.

2) *Lane Denoising*: To improve convergence of detection transformer (DETR)-style networks, Li et al. [18] propose the concept of query denoising. The key idea is to include ground truth queries augmented with noise in the training process, which effectively addresses the inherent instabilities associated with bipartite matching [18]. Inspired by this approach, we introduce lane denoising. For  $i \in \{1, \dots, N_{GT}\}$ , random noise  $\Delta p_{dn,i} = (\Delta x, \Delta y, \Delta z) \stackrel{\text{i.i.d.}}{\sim} \mathcal{U}(-1, 1)$  is added to the  $N_{GT}$  ground truth annotations, comprising three polylines with  $n_{pts}$  points each, representing the centerline and the left and right boundaries. Along with the their reference points  $\hat{p}_{dn} = \lambda_{dn} (p_{dn} + \Delta p_{dn})$ , the final denoising queries are obtained as:

$$\hat{q}_{dn} = q_{dn} + \text{MLP}(\hat{p}_{dn}), \quad (3)$$

with  $\lambda_{dn} \in (0, 1)$  being a hyperparameter for rescaling. Similar to DN-DETR [18], we introduce lane denoising groups, each consisting of a noisy version of all ground truth lane segments. The number of denoising groups is computed as  $\lfloor \frac{n_{dn}}{N_{GT}} \rfloor$ . For each group, we derive a different set of initial reference points from the noisy ground truth to prevent propagating similar information. Finally, to avoid ground truth information leakage [18], masked attention is used to prevent the denoising groups from interacting with other queries.

### B. Road Boundary Transformer

In contrast to the lane segment prediction, the detection of road boundaries necessitates the inference of irregularly shaped polylines that differentiate between drivable and non-drivable area. Thus, we reformulate our lane segment head to accommodate this task to model the road boundary instances as queries and finally output polylines. As opposed to prior work, e.g., MapTR [26], we do not employ hierarchical queries, but leverage one query per instance reducing the computational requirements. Inspired by multi-point attention [56], we predict  $n_{rb}$  points for the road boundary for each intermediate transformer decoder layer, which are used in the consecutive decoder layer to sample  $n_{ref}$  equidistant reference points.

### C. 2D Traffic Element Detection

To associate traffic elements with lane segments, we first detect traffic lights and traffic signs as 2D bounding boxes using the image features of the front camera.

Given our query-based approach, we utilize DN-DETR [18], a denoising Deformable DETR for 2D bounding box detection. Traffic element queries  $q_{te}$  interact with the image features  $F_{PV}^0$  of the front camera and the traffic decoder outputs the most confident predictions as queries. Similar to TopoMLP [48], we use proposals generated by YOLOv9 [43] and incorporate them as additional queries. For simplicity, we denote all traffic element queries as

$q_{te} \in \mathbb{R}^{N_{te} \times C}$ . Thus, the refined queries  $\hat{q}_{te}$  for traffic elements are computed as

$$\hat{q}_{te} = \text{TrafficDecoder}(F_{PV}^0, q_{te}). \quad (4)$$

Further, we employ YOLOv9 [43] and propose to encode the bounding box parameters (e.g., center, height, etc.) into the embedding space with a simple MLP to obtain  $\hat{q}_{te}$ .

### D. Topology Estimation

The topology estimation is designed to facilitate reasoning about the pairwise relationships between lane instances within the same embedding space [2], [21], as well as to associate traffic elements with lane segments. Given the lane segments queries  $\hat{q}_{ls}$  and traffic elements queries  $\hat{q}_{te}$ , our topology head encodes the queries and repeats the lane segment queries  $N_{te}$  times and the traffic element queries  $N_{ls}$  times. The resulting matrices are stacked such that we obtain  $Q_{ls,te} \in \mathbb{R}^{N_{ls} \times N_{te} \times 2C}$ . Similar to [22], [48], we compute the similarity score  $A_{ls,te}^S \in [0, 1]^{N_{ls} \times N_{te}}$  between the instance pair  $(i, j)$  as an adjacency matrix:

$$A_{ls,te}^S(i, j) = \sigma(\text{MLP}(Q_{ls,te}(i, j))), \quad (5)$$

with  $\sigma(\cdot)$  being the sigmoid function. The same operation can be applied straight-forward to estimate the topology  $A_{ls,ls}^S$  among all  $q_{ls}$  instances, i.e., lane segments.

Inspired by Topologic [9], we design a post-processing function that maps the absolute distance between the start and end points of all possible lane instance pairs to a distance-based topology matrix  $A_{ls,ls}^D \in [0, 1]^{N_{ls} \times N_{ls}}$ :

$$A_{ls,ls}^D(i, j) = \frac{2}{1 + \exp\left(\frac{d_{i,j}}{\alpha}\right)}, \quad (6)$$

with  $d_{i,j}$  being the distance between the start/end point of lane segment  $i$  and  $j$  and  $\alpha \in \mathbb{R}$  denoting a hyperparameter for the distance matching. The final relationship scores  $A_{ls,ls}$  are calculated as follows:

$$A_{ls,ls}(i, j) = \min(A_{ls,ls}^S(i, j) + \beta A_{ls,ls}^D(i, j), 1), \quad (7)$$

where  $\beta$  denotes a weight factor to account for the distance-based similarity. We determined empirically that  $\alpha = 2.5$  and  $\beta = 0.8$  lead to the best performance.

### E. Holistic Improvements

We introduce further improvements to the query-based lane segment transformer and road boundary transformer with the aim of enhancing the detection performance.

1) *SMERF*: We employ the idea of SMERF [37] and cross-attend the encoded SD map with the BEV features.

2) *One-to-Many (o2m)*: To further accelerate convergence and improve detection accuracy, we integrate one-to-many matching [14] for lane segment and road boundary detection.

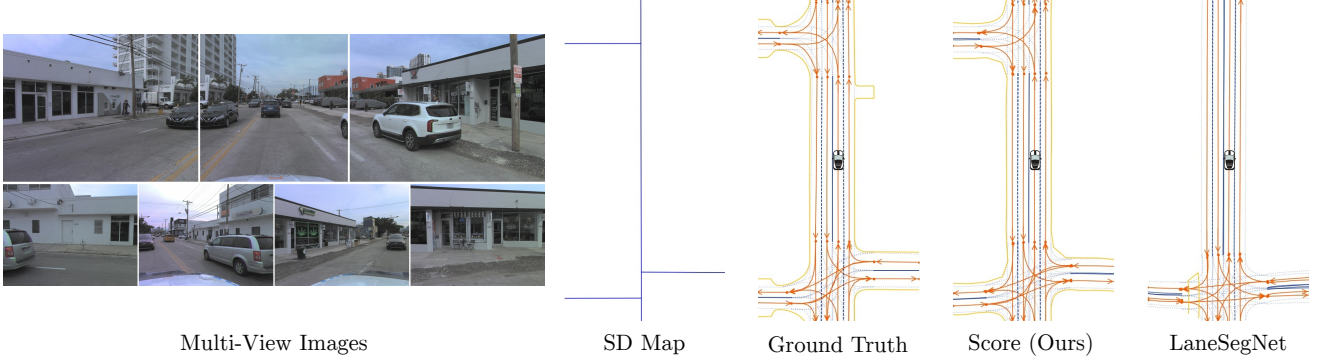


Fig. 3: Qualitative results. Our method demonstrates the capacity to incorporate a SD map prior and past information in an effective manner, facilitating the prediction of partly-occluded roads and providing more consistent predictions than those produced by LaneSegNet [22].

3) *Dataset Resampling*: An analysis of the distribution of categories within the dataset reveals a significant imbalance that can have implications for the model’s performance. Specifically, statistical analysis shows that nearly 96% of the training samples involve the vehicle moving straight, while only 4% contain turning maneuvers. To address this issue, we increase the number of samples in scenarios where the wheel angle is higher than  $7^\circ$ .

4) *Temporal Fusion*: To leverage also past observations, we employ temporal fusion analogous to StreamMapNet [56]. In a streaming fashion, the BEV features  $F_{BEV}$  are recurrently fused with the previous BEV features utilizing a ConvGRU [7] to integrate temporal information. Subsequently, the fused BEV features undergo ego-motion compensation and are being propagated to the next timestep. We initially conduct a warm-up phase [4], [56], i.e., an initial training that does not involve past frames, and then subsequently employ temporal fusion.

#### F. Loss Functions

Our model employs the Hungarian algorithm [17] to match predictions to ground truth during training. The final loss function for the lane head is defined as:

$$\mathcal{L}_{ls} = \lambda_{vec,ls}\mathcal{L}_{vec} + \lambda_{seg,ls}\mathcal{L}_{seg} + \lambda_{cls,ls}\mathcal{L}_{cls} + \lambda_{type}\mathcal{L}_{type} + \lambda_{top}\mathcal{L}_{top} + \lambda_{o2m}\mathcal{L}_{o2m} + \lambda_{dn}\mathcal{L}_{dn}, \quad (8)$$

where

$$\mathcal{L}_{seg} = \lambda_{ce}\mathcal{L}_{ce} + \lambda_{dice}\mathcal{L}_{dice}. \quad (9)$$

For the road boundary head, the loss function is:

$$\mathcal{L}_{rb} = \lambda_{vec,rb}\mathcal{L}_{vec} + \lambda_{seg,rb}\mathcal{L}_{seg} + \lambda_{cls,rb}\mathcal{L}_{cls} + \lambda_{o2m}\mathcal{L}_{o2m}. \quad (10)$$

The vectorized geometric loss  $\mathcal{L}_{vec}$  supervises the predicted geometry by calculating the summed Manhattan distance of the centerlines and left and right boundaries between matched lane segment pairs. A combined cross entropy  $\mathcal{L}_{ce}$  and dice loss  $\mathcal{L}_{dice}$  is used to supervise predicted masks loss  $\mathcal{L}_{seg}$ . A cross entropy loss  $\mathcal{L}_{cls}$  is employed for lane-type and boundary predictions. The topological loss  $\mathcal{L}_{top}$  leverages focal loss [30] to supervise the relationships among lane segments based on the topology information [21]. The terms  $\mathcal{L}_{o2m}$  and  $\mathcal{L}_{dn}$  serve as auxiliary losses for one-to-many matching [27] and denoising [18], respectively.

## IV. EXPERIMENTAL EVALUATION

### A. Datasets and Metrics

The experiments are conducted using the OpenLane-V2 dataset [44], a comprehensive dataset designed for large-scale perception and reasoning in autonomous driving scenarios. OpenLane-V2 consists of two subsets (*subset A* and *subset B*) derived from Argoverse 2 [46] and nuScenes [1], respectively. Each subset includes 1,000 annotated scenes at a frequency of 2 Hz. As *subset B* does not contain lane segment annotations, our work focuses on *subset A*. The training set comprises roughly 27,000 frames, while the validation set includes about 4,800 frames. All lane segments within the spatial range of  $[-50\text{ m}, 50\text{ m}]$  along the x-axis and  $[-25\text{ m}, 25\text{ m}]$  along the y-axis are annotated in 3D space. We use the OpenLane-V2 UniScore (OLUS) as main metric:

$$\text{OLUS} = \frac{1}{5}[\text{DET}_{ls} + \text{DET}_a + \text{DET}_t + f(\text{TOP}_{tl}) + f(\text{TOP}_{tt})], \quad (11)$$

where  $\text{DET}_{ls}$ ,  $\text{DET}_a$ , and  $\text{DET}_t$  denote the mean average precision (mAP) for lane segment, area (road boundary and pedestrian crossing), and traffic element detection, respectively, while TOP refers to mAP over all vertices matched between ground truth graph and predicted graph. For fair comparison with other methods, we explicitly report the mAP for pedestrian crossings  $\text{DET}_{ped}$ . The mAP for area detection is then  $\text{DET}_a = \frac{1}{2}[\text{DET}_{ped} + \text{DET}_b]$  with  $\text{DET}_b$  being the mAP for boundary detection. The OpenLane-V2 Score (OLS) is used as metric for centerline prediction and topology estimation and is reported for methods employing a centerline representation.

OpenLane-V2 [44] provides SD map annotations. However, Luo et al. [37] show that leveraging OpenStreetMap (OSM) data yields better performance due to improved quality of the SD map annotations. Thus, we also leverage OSM SD maps and highlight the benefit in our ablation study.

Prior research [28], [56] has demonstrated that datasets utilized for HD map element detection are susceptible to data leakage, as training and validation locations exhibit substantial overlap. To assess the efficacy and resilience of our proposed approach, we conduct a comparative analysis on a geographically disjoint dataset split [56].



TABLE I: Comparison on the OpenLane-V2 validation split (*subset A*). Results for existing methods are sourced from TopoNet, with the best performance in each category highlighted in bold. \*: baseline results taken from project page [22], †: employs smaller backbone, ‡: results of the test split, CL/LS: centerline and lane segment detection, respectively.

Method	Task	SD map	Temp. Fusion	DET <sub>l/ls</sub>	DET <sub>ped</sub>	DET <sub>b</sub>	DET <sub>a</sub>	DET <sub>te</sub>	TOP <sub>ll</sub>	TOP <sub>lt</sub>	OLS/OLUS
MapTRv2 [27]	CL	✗	✗	17.7	-	-	-	43.5	5.9	15.2	31.0
TopoNet [21]	CL	✗	✗	28.6	-	-	-	48.6	10.9	23.8	39.8
TopoMLP [48]	CL	✗	✗	29.5	-	-	-	49.5	21.7	26.9	44.1
SMERF [37]	CL	✓	✗	33.4	-	-	-	48.6	15.4	23.8	42.9
Topologic [9]	CL	✓	✗	34.4	-	-	-	48.3	28.9	28.7	47.5
LaneSegNet [22]	LS	✗	✗	32.3	32.9	-	-	-	25.4	-	-
LaneSegNet* [22]	LS	✗	✗	27.8	-	-	23.8	36.9	24.1	21.3	36.7
TopoSD [53]†	LS	✓	✗	37.0	-	-	21.6	40.4	33.6	24.0	41.2
Score (Ours)	LS	✓	✓	<b>44.0</b>	<b>47.3</b>	<b>39.6</b>	<b>43.4</b>	<b>61.4</b>	<b>40.0</b>	<b>39.1</b>	<b>54.9</b>
LaneSegNet‡ [22]	LS	✗	✗	27.3	29.2	-	-	-	22.6	-	-
Score (Ours)‡	LS	✓	✓	<b>39.1</b>	<b>40.2</b>	<b>39.8</b>	<b>40.0</b>	<b>76.2</b>	<b>34.5</b>	<b>42.2</b>	<b>55.8</b>

### B. Experimental Setup and Parameters

As BEV transformation, we employ BEVFormer [24] with a ResNet-50 [11] backbone and a feature pyramid network (FPN) [29]. Similar to previous research [21], [22], we reduce the original image size by 50% and leverage AdamW [16] with a cosine annealing learning rate schedule for training. If not otherwise mentioned, our architecture is trained for 30 epochs with single-frame mode, followed by an additional 30 epochs dedicated to temporal fusion. For 2D detection, we train YOLOv9 [43] for 200 epochs. The training process uses 8 NVIDIA Tesla V100 GPUs, with a total batch size of 8.

Following [22], we select  $n_{pts} = 10$  for the each polyline of the lane segment and  $n_{rb} = 20$  for the road boundary. All points are predicted in 3D. The model uses 200 base queries, supplemented by an additional  $n_{dn} = 60$  queries for denoising and  $n_{sd} = 50$  SD enhanced queries.

To solve the bipartite matching for DETR-style architectures, we use the Hungarian algorithm and follow LaneSegNet [22] with their proposed matching costs for the lane segments. The same costs are applied for the road boundary matching. We employ the same weighting parameter  $\lambda$  for both matching and loss. The hyperparameters for the lane head are configured as follows:  $\lambda_{vec,ls} = 0.025$ ,  $\lambda_{seg,ls} = 3.0$ ,  $\lambda_{ce} = 1.0$ ,  $\lambda_{dice} = 1.0$ ,  $\lambda_{cls,ls} = 1.5$ ,  $\lambda_{type} = 0.01$ ,  $\lambda_{top} = 5.0$ ,  $\lambda_{o2m} = 1.0$ , and  $\lambda_{dn} = 1.0$ . The boundary head has different values for the following parameters:  $\lambda_{vec,rb} = 0.0125$ ,  $\lambda_{seg,rb} = 1.5$ , and  $\lambda_{cls,rb} = 0.5$ .

### C. Overall Results

We benchmark our proposed method, Score, against various state-of-the-art algorithms on the OpenLane-V2 validation *subset A*, as shown in Tab. I.

Our approach demonstrates substantial improvements over all prior methods, attaining the best results for each task. Notably, Score shows an increase of 36% on the validation split and 43% on the test split in lane segment detection DET<sub>ls</sub> compared to LaneSegNet, which highlights the effectiveness of leveraging SD maps. Our method capitalizes on the structured prior knowledge provided by the SD

TABLE II: Ablation study of individual components. Starting from a single-frame baseline, each modification is incrementally added, demonstrating its contribution to the overall performance.

Method	DET <sub>ls</sub>	DET <sub>ped</sub>	TOP <sub>ll</sub>
Baseline	32.3	32.9	25.4
+ Boundary & 2D Det. Head	32.1	34.4	28.6
+ SMERF [37]	35.8	41.7	31.3
+ SD Map Enhanced Queries	37.3	41.2	31.8
+ OSM SD Map	38.2	42.0	32.4
+ One-to-Many Matching	39.8	43.5	34.8
+ Lane Denoising	39.8	44.1	34.6
+ Topology Post-Processing	39.8	44.1	36.2
+ Dataset Resampling	40.0	45.1	36.2
+ Temporal Fusion	<b>44.0</b>	<b>47.3</b>	<b>40.0</b>

TABLE III: Disjoint split results. \*: Epochs for temporal training.

Method	Epochs	DET <sub>ls</sub>	DET <sub>ped</sub>	TOP <sub>ll</sub>
LaneSegNet	12	17.2	21.1	16.2
LaneSegNet	24	19.3	21.6	16.9
Score (Ours)	15*	<b>23.8</b>	<b>32.6</b>	<b>22.7</b>
Score (Ours)	30*	21.3	32.7	22.2

map, allowing for superior predictions, even in challenging scenarios where parts of the lane are occluded by vehicles, as shown in Fig. 3. Further, Tab. III highlights that our method also generalizes better compared to LaneSegNet due to the performance increase for the geographically disjoint dataset. As LGMap [49] and MapVision [54] employ large backbones, test-time augmentation and/or ensembles, results are not directly comparable. For a comparison and further results, we would like to refer the reader to our project page.

### D. Ablation Studies

We evaluate the effectiveness of each component of Score in Tab. II. As these components primarily impact the performance of the lane segmentation head, we present results focused solely on that aspect. The incorporation of the SD map into the BEV features [37] exhibits a substantial improvement, while our map-enhanced queries further demonstrate a significant gain in performance. The efficacy of this effect is enhanced by the utilisation of SD map data

TABLE IV: 2D traffic element detection results. Traffic element heads were trained on 2D detection task only. †: using YOLO proposals

Method	DET <sub>t</sub>
DN-DETR [18]	53.0
DN-DETR <sup>†</sup>	57.0
YOLOv9 [43]	<b>61.4</b>

TABLE VI: Results for different temporal fusion settings.

Epochs (single frame)	Epochs (temporal fusion)	DET <sub>ls</sub>	DET <sub>ped</sub>	DET <sub>b</sub>	DET <sub>ll</sub>
30	0	40	45.1	<b>40.5</b>	36.2
0	30	31.1	33.6	31.8	28.4
5	30	40.6	43.5	38.9	36.4
30	30	<b>44.0</b>	<b>47.3</b>	39.7	<b>40.0</b>

sourced from OpenStreetMap (OSM) instead of the original annotations of OpenLane-V2. Further, temporal fusion yields a substantial performance improvement.

In Tab. IV, we present ablation studies comparing the performance of variants of DN-DETR [18] and YOLOv9 [43]. Following [48], we also evaluated DN-DETR using proposals from YOLOv9. For our final architecture, we chose YOLOv9 due to its superior performance as 2D detection head.

To demonstrate the efficacy of our road boundary reformulation using lane attention, we present the results in Tab. V. Training on the road boundary estimation task only, our method with its reformulation outperforms a MapTRv2 [27] head for the same architectural design by a large margin. Furthermore, with additional improvements such as one-to-many and temporal fusion, we achieved a 37% increase in performance. Lastly, we analyze the impact of different warm-up durations compared to our fusion strategy on the model’s performance, as shown in Tab. VI.

Note that the SD map can have ambiguities, e.g., bridge overpass scenarios, but our approach shows robust behavior and uses the image information to infer the correct lane structure (c.f. qualitative results on our project page).

## V. CONCLUSIONS

We propose a coherent approach for jointly predicting lane segments, road boundaries, and 2D traffic elements using onboard sensors. The network is capable of reasoning about the lane topology and associating traffic elements with lanes. It utilizes SD maps to ingest prior knowledge, which can be particularly beneficial for occluded areas as shown in the visual examples. The experimental results demonstrate the efficacy of the network, as it outperforms all recent models on lane segment and road boundary detection, as well as on topology reasoning. Our approach reduces the dependence on HD maps, thus, paving the way toward scalable and mapless autonomous driving.

TABLE V: Boundary detection results. Detection heads were trained on the road boundary task only. †: with techniques (one-to-many, temporal fusion, etc.)

Method	DET <sub>b</sub>
MapTRv2 Head [27]	31.0
Lane Attention	38.4
Lane Attention <sup>†</sup>	<b>42.5</b>

## REFERENCES

- [1] H. Caesar, V. Bankiti, A.H. Lang, S. Vora, V.E. Liong, Q. Xu, A. Krishnan, Y. Pan, G. Baldan, and O. Beijbom. nuScenes: A Multimodal Dataset for Autonomous Driving. In *Proc. of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2020.
- [2] Y.B. Can, A. Liniger, D.P. Paudel, and L. Van Gool. Structured bird’s-eye-view traffic scene understanding from onboard images. In *Proc. of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2021.
- [3] N. Carion, F. Massa, G. Synnaeve, N. Usunier, A. Kirillov, and S. Zagoruyko. End-to-end object detection with transformers. In *Proc. of the Europ. Conf. on Computer Vision (ECCV)*, 2020.
- [4] J. Chen, Y. Wu, J. Tan, H. Ma, and Y. Furukawa. Maptracker: Tracking with strided memory fusion for consistent vector hd mapping. In *Proc. of the Europ. Conf. on Computer Vision (ECCV)*, 2024.
- [5] S. Chen, T. Cheng, X. Wang, W. Meng, Q. Zhang, and W. Liu. Efficient and robust 2d-to-bev representation learning via geometry-guided kernel transformer. *arXiv preprint*, arXiv:2206.04584, 2022.
- [6] S. Choi, J. Kim, H. Shin, and J.W. Choi. Mask2Map: Vectorized HD Map Construction Using Bird’s Eye View Segmentation Masks. In *Proc. of the Europ. Conf. on Computer Vision (ECCV)*, 2024.
- [7] J. Chung, C. Gulcehre, K. Cho, and Y. Bengio. Empirical Evaluation of Gated Recurrent Neural Networks on Sequence Modeling. In *Proc. of the Neural Information Processing Systems Workshops*, 2014.
- [8] W. Ding, L. Qiao, X. Qiu, and C. Zhang. Pivotnet: Vectorized pivot learning for end-to-end hd map construction. In *Proc. of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2023.
- [9] Y. Fu, W. Liao, X. Liu, Y. Ma, F. Dai, Y. Zhang, et al. TopoLogic: An Interpretable Pipeline for Lane Topology Reasoning on Driving Scenes. In *Proc. of the Conf. on Neural Information Processing Systems (NeurIPS)*, 2024.
- [10] C. Han, J. Yang, J. Sun, Z. Ge, R. Dong, H. Zhou, W. Mao, Y. Peng, and X. Zhang. Exploring recurrent long-term temporal fusion for multi-view 3d perception. *IEEE Robotics and Automation Letters (RA-L)*, 9(7):6544–6551, 2024.
- [11] K. He, X. Zhang, S. Ren, and J. Sun. Deep Residual Learning for Image Recognition. In *Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2016.
- [12] J. Huang and G. Huang. Bevdet4d: Exploit temporal cues in multi-camera 3d object detection. *arXiv preprint*, arXiv:2203.17054, 2022.
- [13] J. Huang, G. Huang, Z. Zhu, Y. Ye, and D. Du. Bevdet: High-performance multi-camera 3d object detection in bird-eye-view. *arXiv preprint*, arXiv:2112.11790, 2021.
- [14] D. Jia, Y. Yuan, H. He, X. Wu, H. Yu, W. Lin, L. Sun, C. Zhang, and H. Hu. Detsr with hybrid matching. In *Proc. of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2023.
- [15] J. Jiao. Machine learning assisted high-definition map creation. In *Proc. of the IEEE Annual Computer Software and Applications Conf. (COMPSAC)*, 2018.
- [16] D. Kingma and J. Ba. Adam: A Method for Stochastic Optimization. In *Proc. of the Intl. Conf. on Learning Representations (ICLR)*, 2015.
- [17] H. Kuhn. The hungarian method for the assignment problem. *Naval research logistics quarterly*, 2(1-2):83–97, 1955.
- [18] F. Li, H. Zhang, S. Liu, J. Guo, L.M. Ni, and L. Zhang. Dn-detr: Accelerate detr training by introducing query denoising. In *Proc. of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2022.
- [19] H. Li, Z. Huang, Z. Wang, W. Rong, N. Wang, and S. Liu. Enhancing 3D Lane Detection and Topology Reasoning with 2D Lane Priors. *arXiv preprint*, arXiv:2406.03105, 2024.
- [20] Q. Li, Y. Wang, Y. Wang, and H. Zhao. Hdmapnet: An online hd map construction and evaluation framework. In *Proc. of the IEEE Intl. Conf. on Robotics & Automation (ICRA)*, 2022.
- [21] T. Li, L. Chen, H. Wang, Y. Li, J. Yang, X. Geng, S. Jiang, Y. Wang, H. Xu, C. Xu, J. Yan, P. Luo, and H. Li. Graph-based Topology Reasoning for Driving Scenes. *arXiv preprint*, arXiv:2304.05277, 2023.
- [22] T. Li, P. Jia, B. Wang, L. Chen, K. Jiang, J. Yan, and H. Li. Lanesegnet: Map learning with lane segment perception for autonomous driving. In *Proc. of the Intl. Conf. on Learning Representations (ICLR)*, 2024.
- [23] Y. Li, Z. Ge, G. Yu, J. Yang, Z. Wang, Y. Shi, J. Sun, and Z. Li. Bevedepth: Acquisition of reliable depth for multi-view 3d object detection. In *Proc. of the Conf. on Advancements of Artificial Intelligence (AAAI)*, 2023.

- [24] Z. Li, W. Wang, H. Li, E. Xie, C. Sima, T. Lu, Y. Qiao, and J. Dai. Bevformer: Learning bird's-eye-view representation from multi-camera images via spatiotemporal transformers. In *Proc. of the Europ. Conf. on Computer Vision (ECCV)*, 2022.
- [25] B. Liao, S. Chen, B. Jiang, T. Cheng, Q. Zhang, W. Liu, C. Huang, and X. Wang. Lane graph as path: Continuity-preserving path-wise modeling for online lane graph construction. In *Proc. of the Europ. Conf. on Computer Vision (ECCV)*, 2024.
- [26] B. Liao, S. Chen, X. Wang, T. Cheng, Q. Zhang, W. Liu, and C. Huang. MapTR: Structured Modeling and Learning for Online Vectorized HD Map Construction. In *Proc. of the Intl. Conf. on Learning Representations (ICLR)*, 2023.
- [27] B. Liao, S. Chen, Y. Zhang, B. Jiang, Q. Zhang, W. Liu, C. Huang, and X. Wang. MapTRv2: An end-to-end framework for online vectorized hd map construction. *Intl. Journal of Computer Vision (IJCV)*, 133(3):1352–1374, 2024.
- [28] A. Lilja, J. Fu, E. Stenborg, and L. Hammarstrand. Localization is all you evaluate: Data leakage in online mapping datasets and how to fix it. In *Proc. of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2024.
- [29] T.Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie. Feature pyramid networks for object detection. In *Proc. of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2017.
- [30] T.Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár. Focal Loss for Dense Object Detection. In *Proc. of the IEEE Intl. Conf. on Computer Vision (ICCV)*, 2017.
- [31] X. Lin, T. Lin, Z. Pei, L. Huang, and Z. Su. Sparse4d: Multi-view 3d object detection with sparse spatial-temporal fusion. *arXiv preprint*, arXiv:2211.10581, 2022.
- [32] X. Lin, Z. Pei, T. Lin, L. Huang, and Z. Su. Sparse4d v3: Advancing end-to-end 3d detection and tracking. *arXiv preprint*, arXiv:2311.11722, 2023.
- [33] X. Liu, S. Wang, W. Li, R. Yang, J. Chen, and J. Zhu. Mgmmap: Mask-guided learning for online vectorized hd map construction. In *Proc. of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2024.
- [34] Y. Liu, T. Yuan, Y. Wang, Y. Wang, and H. Zhao. Vectormapnet: End-to-end vectorized hd map learning. In *Proc. of the Intl. Conf. on Machine Learning (ICML)*, 2023.
- [35] Y. Liu, T. Wang, X. Zhang, and J. Sun. Petr: Position embedding transformation for multi-view 3d object detection. In *Proc. of the Europ. Conf. on Computer Vision (ECCV)*, 2022.
- [36] Z. Liu, H. Tang, A. Amini, X. Yang, H. Mao, D.L. Rus, and S. Han. Bevfusion: Multi-task multi-sensor fusion with unified bird's-eye view representation. In *Proc. of the IEEE Intl. Conf. on Robotics & Automation (ICRA)*, 2023.
- [37] K.Z. Luo, X. Weng, Y. Wang, S. Wu, J. Li, K.Q. Weinberger, Y. Wang, and M. Pavone. Augmenting lane perception and topology understanding with standard definition navigation maps. In *Proc. of the IEEE Intl. Conf. on Robotics & Automation (ICRA)*, 2024.
- [38] Z. Ma, S. Liang, Y. Wen, W. Lu, and G. Wan. RoadPainter: Points Are Ideal Navigators for Topology transformER. In *Proc. of the Europ. Conf. on Computer Vision (ECCV)*, 2024.
- [39] B. Opra, B.L. Dem, J.M. Walls, D. Lukarski, and C. Stachniss. Leveraging GNSS and Onboard Visual Data from Consumer Vehicles for Robust Road Network Estimation. In *Proc. of the IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, 2024.
- [40] J. Philion and S. Fidler. Lift, splat, shoot: Encoding images from arbitrary camera rigs by implicitly unprojecting to 3d. In *Proc. of the Europ. Conf. on Computer Vision (ECCV)*, 2020.
- [41] L. Qiao, W. Ding, X. Qiu, and C. Zhang. End-to-end vectorized hd-map construction with piecewise bezier curve. In *Proc. of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2023.
- [42] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A.N. Gomez, L. Kaiser, and I. Polosukhin. Attention Is All You Need. In *Proc. of the Conf. Neural Information Processing Systems (NIPS)*, 2017.
- [43] C.Y. Wang and H.Y.M. Liao. YOLOv9: Learning What You Want to Learn Using Programmable Gradient Information. In *Proc. of the Europ. Conf. on Computer Vision (ECCV)*, 2024.
- [44] H. Wang, T. Li, Y. Li, L. Chen, C. Sima, Z. Liu, B. Wang, P. Jia, Y. Wang, S. Jiang, F. Wen, H. Xu, P. Luo, J. Yan, W. Zhang, and H. Li. OpenLane-V2: A Topology Reasoning Benchmark for Unified 3D HD Mapping. In *Proc. of the Conf. on Neural Information Processing Systems (NeurIPS)*, 2023.
- [45] S. Wang, F. Jia, Y. Liu, Y. Zhao, Z. Chen, T. Wang, C. Zhang, X. Zhang, and F. Zhao. Stream query denoising for vectorized hd map construction. In *Proc. of the Europ. Conf. on Computer Vision (ECCV)*, 2024.
- [46] B. Wilson, W. Qi, T. Agarwal, J. Lambert, J. Singh, S. Khandelwal, B. Pan, R. Kumar, A. Hartnett, J.K. Pontes, D. Ramanan, P. Carr, and J. Hays. Argoverse 2: Next Generation Datasets for Self-driving Perception and Forecasting. In *Proc. of the Conf. on Neural Information Processing Systems (NeurIPS)*, 2021.
- [47] C. Witte, J. Behley, C. Stachniss, and M. Raaijmakers. Epipolar Attention Field Transformers for Bird's Eye View Semantic Segmentation. In *Proc. of the IEEE Winter Conf. on Applications of Computer Vision (WACV)*, 2025.
- [48] D. Wu, J. Chang, F. Jia, Y. Liu, T. Wang, and J. Shen. TopoMLP: An Simple yet Strong Pipeline for Driving Topology Reasoning. In *Proc. of the Intl. Conf. on Learning Representations (ICLR)*, 2024.
- [49] K. Wu, S. Nian, C. Shen, C. Yang, and Z. Li. LGmap: Local-to-Global Mapping Network for Online Long-Range Vectorized HD Map Construction. *arXiv preprint*, arXiv:2406.13988, 2024.
- [50] Z. Xia, X. Pan, S. Song, L.E. Li, and G. Huang. Vision transformer with deformable attention. In *Proc. of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2022.
- [51] Z. Xu, Y. Liu, Y. Sun, M. Liu, and L. Wang. Centerlinedet: Centerline graph detection for road lanes with vehicle-mounted sensors by transformer for hd map generation. In *Proc. of the IEEE Intl. Conf. on Robotics & Automation (ICRA)*, 2023.
- [52] Z. Xu, K.Y.K. Wong, and H. Zhao. InsMapper: Exploring Inner-instance Information for Vectorized HD Mapping. In *Proc. of the Europ. Conf. on Computer Vision (ECCV)*, 2024.
- [53] S. Yang, M. Jiang, Z. Fan, X. Xie, X. Tan, Y. Li, E. Ding, L. Wang, and J. Wang. TopoSD: Topology-Enhanced Lane Segment Perception with SDMap Prior. *arXiv preprint*, arXiv:2411.14751, 2024.
- [54] Z. Yang, M. Liu, J. Xie, Y. Zhang, C. Shen, W. Shao, J. Jiao, T. Xing, R. Hu, and P. Xu. MapVision: CVPR 2024 Autonomous Grand Challenge Mapless Driving Tech Report. *arXiv preprint*, arXiv:2406.10125, 2024.
- [55] J. Yu, Z. Zhang, S. Xia, and J. Sang. Scalablemap: Scalable map learning for online long-range vectorized hd map construction. *arXiv preprint*, arXiv:2310.13378, 2023.
- [56] T. Yuan, Y. Liu, Y. Wang, Y. Wang, and H. Zhao. Streammapnet: Streaming mapping network for vectorized online hd map construction. In *Proc. of the IEEE Winter Conf. on Applications of Computer Vision (WACV)*, 2024.
- [57] H. Zhang, D. Paz, y. Guo, A. Das, X. Huang, K. Haug, H. Christensen, and L. Ren. Enhancing Online Road Network Perception and Reasoning with Standard Definition Maps. In *Proc. of the IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, 2024.
- [58] Z. Zhang, Y. Zhang, X. Ding, F. Jin, and X. Yue. Online vectorized hd map construction using geometry. In *Proc. of the Europ. Conf. on Computer Vision (ECCV)*, 2024.
- [59] B. Zhou and P. Krähenbühl. Cross-view transformers for real-time map-view semantic segmentation. In *Proc. of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2022.
- [60] Y. Zhou, H. Zhang, J. Yu, Y. Yang, S. Jung, S.I. Park, and B. Yoo. HiMap: Hybrid Representation Learning for End-to-end Vectorized HD Map Construction. In *Proc. of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2024.