2024 IEEE International Conference on Robotics and Automation (ICRA 2024)
May 13-17, 2024. Yokohama, Japan

Augmenting Lane Perception and Topology Understanding with Standard Definition Navigation Maps

Katie Z Luo[†], Xinshuo Weng[‡], Yan Wang[‡], Shuang Wu[‡], Jie Li[‡],
Kilian Q Weinberger[†], Yue Wang[§‡], Marco Pavone[¶‡]

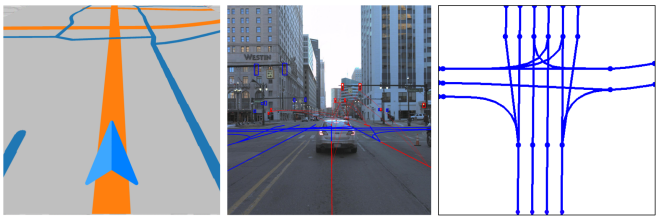
***Abstract*— Autonomous driving has traditionally relied heavily on costly and labor-intensive High Definition (HD) maps, hindering scalability. In contrast, Standard Definition (SD) maps are more affordable and have worldwide coverage, offering a scalable alternative. In this work, we systematically explore the effect of SD maps for real-time lane-topology understanding. We propose a novel framework to integrate SD maps into online map prediction and propose a Transformer-based encoder, SD Map Encoder Representations from transFormers, to leverage priors in SD maps for the lane-topology prediction task. This enhancement consistently and significantly boosts (by up to 60%) lane detection and topology prediction on current state-of-the-art online map prediction methods without bells and whistles and can be immediately incorporated into any Transformer-based lane-topology method. Code is available at https://github.com/NVlabs/SMERF.**

I. Introduction

In order for autonomous driving and driver assistance systems to operate reliably, they need to be aware of the environment and scene elements in tremendous detail. Among others, accurate lane geometry, relational reasoning of lane graphs, and associating lanes to traffic lights and signs are paramount for vehicles to drive correctly. These tasks, known together as lane-topology [1], remain as challenges to deploying autonomous vehicles into the real world.

One solution to this challenge, where autonomous driving systems have found success, lies in deploying strictly within geo-fenced areas, where regions are annotated in detail in the form of High Definition (HD) maps [2], [3], [4]. HD maps typically include centimeter level map elements [5] such as road boundaries, lane dividers, road markings, and traffic signs, as well as lane graphs and association of lanes to traffic signs. This precision mapping removes ambiguity from self-driving, making HD maps critical enablers for essentially all commercial robo-taxi services (e.g. Waymo, Cruise). In addition, HD maps also annotate areas like construction zones and pedestrian crossings to be high alert areas.

While HD maps provide a solution for reliable self-driving, such maps are prohibitively expensive to obtain as each area needs to be painstakingly annotated by humans and continuously updated to reflect any changes in road conditions or ongoing daily constructions. For these reasons, over-reliance on HD maps not only inhibits the scalability of self-driving, but also requires a large number of annotators at-the-ready to keep the maps constantly updated.



(a) SD Map (b) Topology Reasoning (c) Lane Centerlines

Fig. 1: **Lane-Topology Reasoning.** Leveraging standard definition (SD) map (a) with prior information of the road-level topology, our work aims to improve lane centerline detection (c), lane-topology reasoning between lane centerlines, and traffic elements (b). In the SD map, orange lines and teal lines correspond to roads and pedestrian ways, respectively.

In contrast, Standard Definition (SD) maps are cheaper to obtain (*e.g.,* crowdsourcing, aerial images) and are already available for much of the world's areas. Concretely, SD maps mark out road-level topology with metadata[1], as opposed to the full semantic and geometric lane-level details in HD maps. Such SD maps do not need to be updated as frequently as HD maps unless the road topology changes significantly (*e.g.* new roads are constructed, or old roads are removed).

While being much cheaper and having broader coverage, SD maps include crucial information for road topology which can complement onboard cameras for lane-topology reasoning. In particular, when merging or exit roads are not visible in the camera images due to occlusion, an SD map can provide priors for more accurate downstream planning. Additionally, an SD map may provide priors over the existence of intersections before the self-driving car approaches. Such prior knowledge is helpful for long-horizon behavior planning, *e.g.*, switching to a left lane early before making a left turn at the intersection.

In this work, we explore the use of SD maps to improve online lane-topology reasoning in the absense of HD maps. We propose a novel and compellingly simple way of encoding the SD map in a Transformer encoder architecture [6] to learn feature representations that can be consumed in downstream lane-topology tasks. We name our method *SMERF (SD Map*

† Computer Information Sciences Department, Cornell University {kzl6, kqw4}@cornell.edu
‡ NVIDIA {xweng, yanwan, shwu, jieli}@nvidia.com
§ Department of Computer Science, University of Southern California yue.w@usc.edu
¶ Department of Aeronautics and Astronautics, Stanford University pavone@stanford.edu

[1]Metadata can vary across different SD map providers, but it typically includes the type of the roads, the number of lanes of a road, the existence of a traffic sign, and the direction of the road.

Encoder Representations from transFormers).

The framework for augmenting with SD maps is immediately applicable to any Transformer-based lane-topology methods, which currently dominate the state-of-the-art performance metrics [7], [8], [9]. We demonstrate that adding SD maps as an additional source of information gives a boost in performance for lane-topology reasoning — across *all* available architectures. When used with the current best open-sourced lane-topology model [7], lane detection and lane-topology prediction achieve state-of-the-art performance *without any additional tuning*. This showcases the strong generalizability of SMERF map representations, and is a testament to the inherent information present in SD maps for topology understanding. Our contributions are summarized as follows:

- To our knowledge, we are the first work to systematically explore the utility of SD maps for lane-topology understanding.
- We propose SMERF, an SD map representation and Transformer based encoder model for lane-topology prediction.
- We empirically demonstrate that our proposed method of incorporating SD maps significantly boosts the performance of *all* lane-topology methods evaluated.

II. Related Work

Online HD Map Prediction has emerged to reduce the vast amount of human efforts in annotating and maintaining HD maps by predicting the HD maps on-the-fly while the car is in use. HDMapNet [2] pioneers learning-based models to predict map elements from onboard sensors, followed by a post-processing step to convert dense rasterized segmentations to vectorized map representations. To eliminate the need for hand-crafted post-processing, VectorMapNet [10] and InstaGraM [11] introduced end-to-end models for vectorized HD map learning. As the vectorized map learning typically involves key-point sampling along the polylines which can cause information loss, prior work [12], [13] proposed a pivot-based representation and differentiable rasterizer, tailored to corners and fine-grained geometry. Besides output representations, MapTR [14] and InsightMapper[15] proposed a hierarchical query design and models map elements as a point set with a group of equivalent permutations. Recent work such as LaneGAP [16] and TopoNet [17] also explored end-to-end approaches to learn the lane graph and model map element relationships. In contrast to prior work which only uses onboard sensors as inputs, our work is the first to investigate the effect of SD maps to vectorized map learning and can be seamlessly employed to improve prior work.

Learn to Fuse Prior for HD Map Learning. Orthogonal to the above work, recent efforts capitalize on the fusion of prior information to improve the robustness and performance of online HD mapping. NMP [18] learns a neural representation and builds a global map prior from past traversals to improve online map prediction, while [19] optimizes in the latent space to learn a global consistent prior of maps. [20] supplements the onboard camera images with satellite images to deal with the long-range perception of HD maps. NeMO [21] and StreamMapNet [22] improve performance using temporal information by fusing frames from history, while Bi-Mapper [23] designed a multi-view fusion module to leverage priors in the perspective view. Our work is closely related to these methods, however, we leverage a different prior – SD maps, that is much more compact in terms of storage and representation, and also widely accessible.

Transformers for 3D Perception. Transformers have demonstrated their dominant performance in 3D perception from visual inputs. DETR3D [24] leverages a Transformer architecture to connect 2D observations and sparse 3D predictions, enabling non-maximum suppression (NMS) free object detection. BEVFormer [25] extends DETR3D by transforming features from perspective views into a dense birds-eye view (BEV). PETR3D [26] further incorporates 3D positional information into feature extraction, producing 3D position-aware features. In addition to camera images, recent methods leverage multi-modal inputs to complement a single modality. FUTR3D [27] capitalizes on 3D-to-2D queries proposed by DETR3D to fuse features from multiple modalities. Our method is also a Transformer-based architecture that performs online mapping in an end-to-end fashion.

III. Approach

Problem setup. Following prior work [1], [7], we assume a multi-camera setup: the ego-vehicle is equipped with C synchronized, multi-view cameras and their corresponding camera intrinsic and extrinsic parameters. In addition, we have access to the SD maps and the ego-vehicle's 2D position and heading as a 3-DoF rigid transformation G_p from a global positioning system (GPS) that is used to align the SD maps with onboard sensor inputs. From these inputs, the task is to detect the *lane centerlines* of the road and the *traffic elements* of the scene such as the traffic lights and stop signs. Further, we infer the connectivity of the lane centerlines and how they relate to each traffic element. All pairwise relationships are represented as affinity matrices.

The pipeline of SMERF is shown in Fig. 2. The proposed SMERF (lower half) augments an existing lane-topology model (upper half) with priors from SD maps in order to better detect lane centerlines and relational reasoning. Specifically, we first retrieve the SD map, which is encoded into a feature representation using a Transformer encoder. Then, we apply cross-attention between the SD map feature representation with the features from onboard camera inputs to construct the BEV features for lane detection and relational reasoning. The pipeline is trained end-to-end with the lane-topology model without requiring any additional training signals.

A. SD Map Input

We obtain SD maps from OpenStreetMap (OSM) [28], a crowd-sourced platform offering SD maps and geographical details of worldwide locations. Concretely, SD maps from OSM contain road-level topology (*i.e.* road geometry and connectivity) and annotated type-of-road information for each

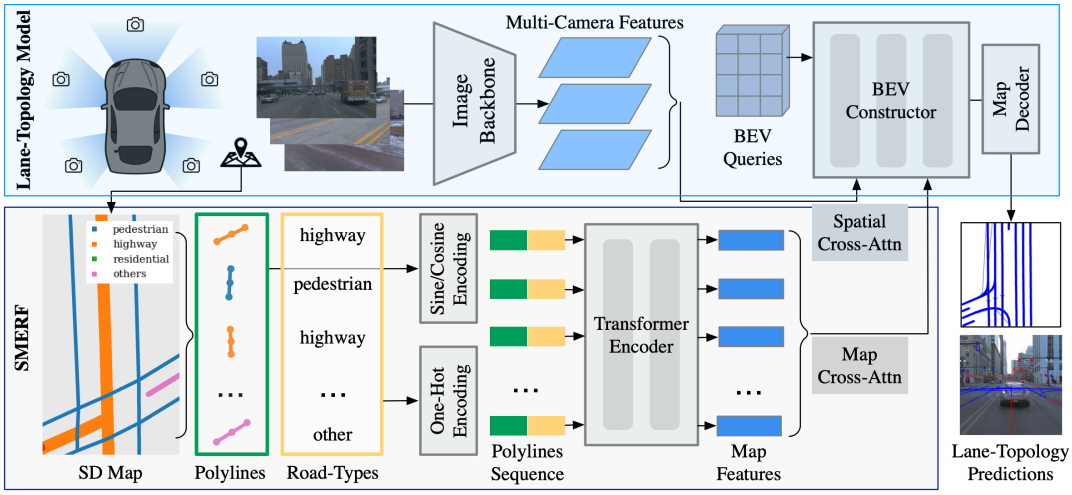


Fig. 2: **Overall approach of SMERF.** The SD map is queried at the ego vehicle's location as a prior for lane-topology reasoning. SD map is first converted into a polyline-sequence representation, then encoded by a Transformer encoder. Our method is amenable to any transformer-based lane-topology models via cross-attention with the SD map features.

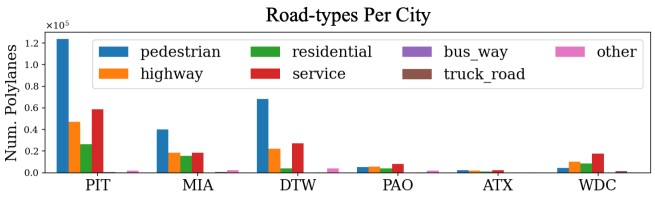


Fig. 3: **Distribution of road types.** We visualize the road types obtained from OpenStreetMap corresponding to the frames in OpenLane-V2 dataset validation split. The colors distinguish by the road-types located in each city.

road segment (*e.g.* highway, residential roads, and pedestrian crossings). For every frame, we extract a local SD map from OSM based on the ego vehicle's position from G_p. The resulting SD map encompasses M polylines, where each polyline corresponds to a road segment. Notably, we transform the point location of the polylines to the ego vehicle's coordinates using G_p. Moreover, each polyline is further annotated with specific road-type labels. An in-depth analysis of the availability of the road types, along with their distribution patterns, is presented in Fig. 3.

B. Encoding Representation from SD Maps

In order to encode the SD map in a form that can be consumed by a downstream lane-topology model, we introduce a polyline-sequence representation and a Transformer encoder to obtain our final map representation for the scene.

Polyline Sequence Representation. Given the SD map of the scene, we evenly sample along each of the M polylines for a fixed number of N points, denoted by $\{(x_i, y_i)\}_{i=1}^{N}$. We employ sinusoidal embeddings with varied frequencies to encode the polyline point locations. Sinusoidal embeddings enhance the sensitivity to positional variations. This sensitivity benefits the model, enabling it to effectively reason about the structure of polylines. Consider a vertical polyline with a small curvature, characterized by closely similar y-axis values for all points. Directly inputting these point coordinates into the model may result in an inadequate distinction of this curvature. However, with sinusoidal embeddings [6], the distinction becomes pronounced, thus improving the model's interpretability of such features. Given a coordinate position $p \in \{x_i, y_i\}$ and an embedding dimension $j \in \{1 \cdots d/2\}$, the sinusoidal embedding can be formulated as:

$$E(p, 2j) = \sin\left(\frac{p}{T^{2j/d}}\right),$$

$$E(p, 2j+1) = \cos\left(\frac{p}{T^{2j/d}}\right),$$

where d is the dimension of the embedding, and $T = 1000$ is the temperature scale. This enables the transformation of (x_i, y_i) coordinates into their corresponding sinusoidal embeddings of dimension d. In practice, we normalize each polyline's coordinates with respect to the BEV range into the range of $(0, 2\pi)$ prior to embedding them.

We use a one-hot vector representation for the road-type label with dimension K for the main types of lanes present in OSM. This not only ensures that input values are normalized between 0 and 1, but also addresses cases where a road segment may fall into multiple road types. Finally, we concatenate the polyline positional embeddings with the road type as one-hot vectors for the final polyline sequence representation with shape $M \times (N \cdot d + K)$.

Transformer Encoder of Map Features. Given the polyline sequence representation of the SD map, we wish to use a Transformer encoder [6] to learn a feature representation for the downstream lane-topology task. We embed the polyline sequence with a linear layer, typical of Transformer encoder architectures. This ensures that the discrete, one-hot representation of the road-types can be meaningfully transformed into continuous space. We then utilize L layers of multi-head self-attention to extract and encode the global geometric and semantic information from the SD map input.

TABLE I: **Performance on the OpenLane-V2 Dataset.** We report the performance of adding SMERF to state-of-the-art lane-topology models, and we observe that SMERF can significantly improve both the Baseline model and TopoNet.

	DET_l	TOP_ll	DET_t	TOP_lt	OLS
Baseline	17.0	2.3	48.5	16.2	30.2
Baseline+SMERF	26.8	3.9	48.9	19.2	34.8
△ Improvement	57.6%	73.1%	0.8%	18.6%	15.3%
TopoNet	28.2	4.1	44.5	20.6	34.5
TopoNet+SMERF	33.4	7.5	48.6	23.4	39.4
△ Improvement	18.7%	83.8%	9.3%	13.6%	14.2%

The resultant output has a shape of $M \times H$, where H denotes the feature dimension produced by the self-attention layer.

C. Lane-Topology Prediction with SMERF

The SD map representation from SMERF can now be used by any Transformer-based lane-topology model. With the release of the lane-topology task alongside the OpenLane-V2 dataset [1], the predominant paradigm for lane detection and relational reasoning models emerged consisting of an BEV Transformer encoder and a DeTR-based map decoder [7], [8], [9]. In this work, we propose to augment these state-of-the-art lane-topology models with features representations from the SD map. Current methods only leverage multi-view camera inputs, and have difficulty predicting in areas that are occluded or are far away. This is complimented by the additional SD map information, which allows the model to reason about these blind spots.

To make use of current state-of-the-art lane-topology models, SMERF fuses the SD map features with the intermediate BEV feature representations by leveraging multi-head cross-attention. This method is compatible with nearly all transformer-based lane-topology models by applying cross-attention between the BEV feature queries and the SD map features in each intermediate layer of the model's encoder (Fig. 2, "Map Cross-Attn"). In practice, we cross-attend the SD map features after each spatial cross-attention operation. Thus, the fused BEV features include not only the 3D information derived from the images, but also the road-level geometric information extracted from the SD map. Subsequently, the lane-topology model decoder takes the SD map-augmented features as inputs to predict the lane centerlines, traffic elements, and affinity matrices for the association of lane centerlines and traffic elements.

IV. Experiments

Dataset and Evaluation Metrics. We validate our approach on the OpenLane-V2 dataset [1], a large, real-world perception dataset for scene structure in autonomous driving. To the best of our knowledge, this is the only dataset providing ground truth to test lane and traffic detection, as well as topology relationships among lane centerlines and between lane centerlines and traffic elements. For this work, we report results on the primary subset (`subset_A`), which is labeled on top of the Argoverse dataset [29]. We adopt the metrics from the dataset's lane centerline evaluation and evaluate performance within 50m in-front and behind, and 25m to either side of the ego-vehicle. Reported metrics include DET_l, TOP_ll, DET_t, and TOP_lt [1], which correspond to the mean average precision (mAP) on directed lane centerlines, traffic elements, topology among lane centerlines, and topology between lane centerlines and traffic elements. The mAP is averaged over match thresholds of $\{1, 2, 3\}$m on Frechet distance for lane centerlines and a match threshold of 0.75 IoU for traffic elements when determining correspondence between detections and ground truths. We additionally report the consolidated OpenLane-V2 Score (OLS), which was released with the dataset's CVPR 2023 challenge.

Baselines. We experiment with two high-performing, open-source lane-topology models: BEVFormer-DeTR baseline and TopoNet [7]. The BEVFormer-DeTR baseline was released with the OpenLane-V2 dataset as "baseline large", and is denoted as "Baseline" in our experiments. It leverages the architecture from [25] to encode multi-camera features into birds-eye-view (BEV), and decode them into lane centerlines and traffic elements using a customized Deformable-DeTR [30] head for lane-topology reasoning. TopoNet is the current state-of-the-art open-sourced work, which also constructs a BEV representation of the scene, and additionally leverages a graph neural network [31] to explicitly reason about lane centerline and traffic element topology.

Implementation Details. For reproducibility, we use the official implementation of both the Baseline model [1] and the TopoNet model [7]. Both models use the default ResNet50 image backbone, and the output representation of the Baseline model is changed to the 11-point representation used in TopoNet for a fair comparison. Our SMERF Transformer encoder is implemented with the official Pytorch implementation. All road types from the queried SD map are grouped into the following categories: pedestrian, highway, residential, service, bus_way, and truck_road, with an additional catch-all category. We set $N = 11$ for the polyline sequence representation of SD maps, $d = 32$ for the dimension of positional embeddings, and $L = 6$ layers for the SD map Transformer encoder.

A. Lane Topology Prediction Performance

We show our performance on the OpenLane-V2 dataset in Tab. I, comparing results of lane-topology models with and without SMERF. Observe that across all metrics, we obtain a significant performance boost by using SD map information as compared to models without. For both Baseline and TopoNet, adding SMERF gives about 15% performance boost to the consolidated OLS metric. In particular, lane detection and lane-topology prediction, corresponding to metrics DET_l and TOP_ll respectively, receive the largest boost in performance. The Baseline model and TopoNet gain 9.8 mAP and 5.2 mAP, respectively, for lane detection and both models nearly double their lane centerline-topology prediction performances. This aligns with our intuition that SD maps contain information of the road topology which is helpful to compliment onboard camera inputs to further reason lane-level topology.

TABLE II: **Performance Breakdown.** We report the performance of baseline models as compared to SMERF method, broken down by lane centerlines that are close *vs*. far, and non-intersection *vs*. intersections. Notice how improvements are greater at far away lanes and at intersections. We leave out DET_t metrics for brevity, since we do not filter by traffic elements.

	Close (0 - 25 m)				Far (25 - 50 m)				Non-Intersection/Connector				Intersection/Connector			
	DET_l	TOP_ll	TOP_lt	OLS	DET_l	TOP_ll	TOP_lt	OLS	DET_l	TOP_ll	TOP_lt	OLS	DET_l	TOP_ll	TOP_lt	OLS
Baseline	20.4	4.8	16.3	32.8	16.3	2.3	14.1	29.4	20.3	3.0	16.2	31.6	14.7	2.0	11.9	27.9
Baseline+SMERF	22.4	6.6	18.4	35.0	26.3	3.9	17.5	34.2	29.2	5.3	19.3	36.3	21.2	3.7	14.9	32.0
△ Improvement	9.8%	38.1%	13.0%	6.8%	61.0%	72.6%	23.9%	16.6%	43.4%	74.2%	19.0%	14.7%	44.3%	87.8%	24.9%	14.6%
TopoNet	28.0	9.1	22.6	37.6	26.5	4.4	18.2	33.6	32.7	8.1	21.2	37.9	23.7	4.6	16.0	32.4
TopoNet+SMERF	32.4	12.6	24.3	41.5	33.0	7.7	21.7	39.0	37.0	11.9	23.7	42.2	28.5	8.1	19.5	37.4
△ Improvement	15.7%	38.2%	7.6%	10.4%	24.9%	76.8%	19.0%	16.0%	13.3%	46.6%	11.7%	11.3%	20.3%	74.4%	22.1%	15.5%

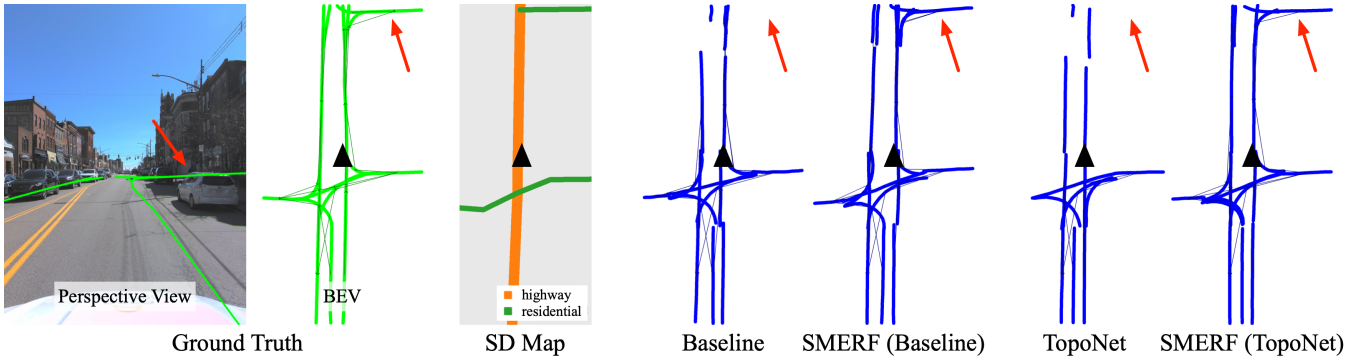


Fig. 4: **Qualitative Results.** We visualize the lane predictions from OpenLane-V2 dataset validation split along with the ground truth lane-topology and the corresponding SD map from the location. Observe that adding SD maps allows the model to infer lanes that are far away and even occluded by a building (marked with the red arrows). We denote the ego-vehicle's position with a black triangles for visualization purposes.

TABLE III: **Comparison of SD map models.** In short, adding SD maps boosts lane-topology performance regardless of the methods, and the proposed SMERF is most effective and can provide the highest performance gains.

Method	DET_l	TOP_ll	DET_t	TOP_lt	OLS
Baseline (– map)	17.0	2.3	48.5	16.2	30.2
Raster Map	20.6	2.0	49.4	15.5	30.9
VectorNet	19.1	2.3	**50.3**	16.2	31.2
SMERF (Ours)	**26.8**	**3.9**	48.9	**19.2**	**34.8**

Performance Breakdown. To better understand where the performance gains are from, we break performance down by evaluating at lane centerlines that are close-by *vs*. far-away and intersections *vs*. non-intersections in Tab. II. When comparing performance between close-by vs. far-away areas, we observe that the majority of improvements lies in the lane centerline predictions that are further away; by adding SD maps as a prior, far-away lanes detection and topology reasoning enjoy a significant improvement. Both models' performance without the SD map prior drops significantly when attempting to reason about lane topology that are far-away—3.4 points and 4 points respectively on the OLS metric. However, by incorporating SD maps via SMERF, the models have a much smaller performance degradation in comparison—0.8 points and 1.5 points. Surprisingly, when comparing performance at intersections *vs*. non-intersections, we observe that improvements are more significant for lane-topology reasoning (TOP_ll metric). This suggests that SD maps contain priors that are useful to reason about more complicated scenes such as intersections.

Qualitative Comparison. In Fig. 4 from left to right, we visualize the frontal camera view, ground truth lane-topology, the corresponding SD map of the area, and lane-topology predictions of the baseline models with and without SMERF. Observe that using the SD map allows the models to better reason and predict lane centerlines further away, especially in the case where the information is missing in the camera images due to occlusion or far range. This observation aligns with our quantitative results and provides an intuition for why SD maps help most in these challenging cases, which could be crucial to long-horizon behavior planning.

B. Analysis and Ablation Study

We analyze the information provided by SD maps and ablate components of SMERF. All experiments are based on the Baseline provided by the official OpenLane-V2 codebase.

Comparison of SD Map Models. In this work, we leverage information provided in SD maps to improve lane-topology prediction. An astute reader may wonder if SD map information can improve performance regardless of SD map representation encoders and how effective our proposed SMERF is. In Tab. III, we compare SMERF against two representative methods: 1) Raster Map [32] and 2) VectorNet [33]. Observe that using SD map boosts lane-topology performance, *regardless of the method used*. Raster Maps represent the SD map as a heatmap with a one-hot

TABLE IV: **Ablation on the road-type information.** Observe that the addition of semantic information (road types) from SD maps is helpful to improve the performance of SMERF.

Road-Types			DET_l	TOP_ll	DET_t	TOP_lt	OLS
Road	Ped-Way	Others					
–	–	–	17.0	2.3	48.5	16.2	30.2
✓			20.6	3.7	51.5	18.0	33.4
✓	✓		22.7	3.8	52.3	19.8	34.8
✓	✓	✓	26.8	3.9	48.9	19.2	34.8

TABLE V: **Results on geo-disjoint training and validation split.** We re-split the training and validation set to be geographically disjoint. Observe that adding SMERF still provides a consistent performance boost, despite the models being evaluated on a more challenging data split.

	DET_l	TOP_ll	DET_t	TOP_lt	OLS
Baseline	5.4	0.3	41.0	5.7	16.9
Baseline+SMERF	8.8	0.5	46.3	6.9	22.1
△ Improvement	64.6%	66.7%	13.0%	21.4%	31.1%
TopoNet	14.9	1.0	34.3	7.6	21.7
TopoNet+SMERF	17.0	1.4	35.4	8.6	23.4
△ Improvement	14.4%	31.2%	3.1%	14.0%	7.5%

representation to encode the road type; the map features are obtained using a ResNet50 backbone. While it is able to improve lane detection, such a representation struggles with relational reasoning and performs poorly on metrics of TOP_ll and TOP_lt. VectorNet represents the SD map as a graph of polylines and uses a Graph Neural Network to refine the map features. Thus, it is better at relational reasoning. Overall, SMERF shows the strongest performance since its architecture is designed explicitly for SD map encoding.

Analysis on Road Types. We analyze how the road type information contributes to the performance. We first group the types of roads into three high-level categories: (1) roads (*i.e.* highway and residential), (2) pedestrian-way (pedestrian walkways, crosswalks, *etc.*), and (3) other types (bus_way, truck_road, service, *etc.*). Then, we report performance in Tab. IV by incrementally adding the road type information. Observing that adding more priors of semantic information (*i.e.* road types) improves performance. Even with only labels for "roads" present in SD maps, the Baseline model is able to achieve a significant performance gain.

Evaluation on Geo-disjoint Data Split. The standard training and validation split in the OpenLane-V2 dataset consists of geographically overlapping areas, and models trained and tested under such a setting are known to overfit [34], [7]. We analyze the performance on a *geographically disjoint* training and validation split and report our results in Tab. V. Observe that improvements with the SD map are consistent and significant even in the geographically disjoint splits, despite the performance drops in both the Baseline and TopoNet on this more challenging data split. Future work on these geo-disjoint splits is suggested for reducing overfitting and improving generalization across geographical areas.

TABLE VI: **Ablation on Transformer encoder components.** We report performance after adding each component of the map transformer. Adding positional encoding is crucial for good topology reasoning.

	DET_l	TOP_ll	DET_t	TOP_lt	OLS
Baseline (– map)	17.0	2.3	48.5	16.2	30.2
+ map transformer	18.2	2.6	48.1	16.8	30.9
+ x,y-pos encoding	20.0	**3.9**	**49.6**	18.9	33.2
+ normalization	**26.8**	**3.9**	48.9	**19.2**	**34.8**

TABLE VII: **Ablation on Transformer Components.** We note that using fewer heads actually improves performance, as the per-head dimension gets larger.

# Heads	DET_l	TOP_ll	DET_t	TOP_lt	OLS
4	26.8	3.9	48.9	19.2	34.8
8	22.8	3.6	52.5	19.0	34.5
16	21.3	4.1	50.4	19.3	33.9

Effects on Different Model Components. We additionally ablate components of our model to justify our implementation details. We report ablation results when adding different Transformer architecture components in Tab. VI. Adding the transformer model naively gives only a small boost in performance. By encoding the x,y positions of the SD map polylines with sine-cosine encoding [6], we gain expressivity for the model to represent the locations of the map elements, thus boosting performance. Lastly, normalizing point coordinates into a range of $(0, 2\pi)$ prior to the positional encodings gives a final boost in lane detection performance.

We additionally ablation the SMERF architecture design and the effect of different numbers of heads on performance in Tab. VII. Observe that the dimension, which is by default quite small, performs best with fewer heads since the dimension-per-head is reduced as the number of heads is increased. Due to dataset size, model performance on lane detection is sensitive to the per-head dimension.

V. Discussion and Conclusion

In this work, we explore the benefits of leveraging readily available and cost-effective SD maps, and study how they can improve online map prediction and lane-topology reasoning. Our method, SMERF, demonstrates consistent performance improvements over various lane-topology models, indicating the versatility of this approach. Our work is the first to approach task of lane-topology prediction by leveraging SD map priors; further directions include refining the representation learning process for SD maps within the Transformer architecture to enable more significant performance gains.

Acknowledgment

We thank Boris Ivanovic, Boyi Li, and the entire Nvidia DeepMap team for their insightful discussions and valuable feedback. This research was supported in part by a grant from the NSF (IIS-2107161), and Katie Luo was supported by an Nvidia Graduate Fellowship.

REFERENCES

[1] H. Wang, T. Li, Y. Li, L. Chen, C. Sima, Z. Liu, Y. Wang, S. Jiang, P. Jia, B. Wang, F. Wen, H. Xu, P. Luo, J. Yan, W. Zhang, and H. Li, "Openlane-v2: A topology reasoning benchmark for scene understanding in autonomous driving," *arXiv preprint arXiv:2304.10440*, 2023.

[2] Q. Li, Y. Wang, Y. Wang, and H. Zhao, "HDMapNet: An Online HD Map Construction and Evaluation Framework," in *2022 International Conference on Robotics and Automation (ICRA)*. IEEE, 2022, pp. 4628–4634.

[3] A. Sadat, S. Casas, M. Ren, X. Wu, P. Dhawan, and R. Urtasun, "Perceive, predict, and plan: Safe motion planning through interpretable semantic representations," in *European Conference on Computer Vision*. Springer, 2020, pp. 414–430.

[4] S. Shi, L. Jiang, D. Dai, and B. Schiele, "Motion transformer with global intention localization and local movement refinement," 2023.

[5] R. Liu, J. Wang, and B. Zhang, "High definition map for automated driving: Overview and analysis," *The Journal of Navigation*, vol. 73, no. 2, p. 324–341, 2020.

[6] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," *Advances in neural information processing systems*, vol. 30, 2017.

[7] 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.

[8] D. Wu, F. Jia, J. Chang, Z. Li, J. Sun, C. Han, S. Li, Y. Liu, Z. Ge, and T. Wang, "The 1st-place solution for cvpr 2023 openlane topology in autonomous driving challenge," *arXiv preprint arXiv:2306.09590*, 2023.

[9] M. Lu, Y. Huang, J. Liu, J. Peng, L. Tian, and A. Sirasao, "Separated roadtopoformer," *arXiv preprint arXiv:2307.01557*, 2023.

[10] Y. Liu, Y. Wang, Y. Wang, and H. Zhao, "VectorMapNet: End-to-End Vectorized HD Map Learning." *arXiv preprint arXiv:2206.08920*, vol. 283, 2022.

[11] J. Shin, F. Rameau, H. Jeong, and D. Kum, "InstaGraM: Instance-Level Graph Modeling for Vectorized HD Map Learning," *arXiv preprint arXiv:2301.04470*, 2023.

[12] W. Ding, L. Qiao, X. Qiu, and C. Zhang, "PivotNet: Vectorized Pivot Learning for End-to-End HD Map Construction," *arXiv preprint arXiv:2308.16477*, 2023.

[13] G. Zhang, J. Lin, S. Wu, Y. Song, Z. Luo, Y. Xue, S. Lu, and Z. Wang, "Online Map Vectorization for Autonomous Driving: A Rasterization Perspective," *arXiv preprint arXiv:2306.10502*, 2023.

[14] 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," *arXiv preprint arXiv:2208.14437*, 2022.

[15] Z. Xu, K. K. Wong, and H. Zhao, "InsightMapper: A Closer Look at Inner-instance Information for Vectorized High-Definition Mapping," *arXiv preprint arXiv:2308.08543*, 2023.

[16] 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," *arXiv preprint arXiv:2303.08815*, 2023.

[17] T. Li, L. Chen, X. Geng, H. Wang, Y. Li, Z. Liu, S. Jiang, Y. Wang, H. Xu, C. Xu, *et al.*, "Topology Reasoning for Driving Scenes," *arXiv preprint arXiv:2304.05277*, 2023.

[18] X. Xiong, Y. Liu, T. Yuan, Y. Wang, Y. Wang, and H. Zhao, "Neural Map Prior for Autonomous Driving," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 17 535–17 544.

[19] Y. B. Can, A. Liniger, D. P. Paudel, and L. Van Gool, "Prior Based Online Lane Graph Extraction from Single Onboard Camera Image," *arXiv preprint arXiv:2307.13344*, 2023.

[20] W. Gao, J. Fu, H. Jing, and N. Zheng, "Complementing Onboard Sensors with Satellite Map: A New Perspective for HD Map Construction," *arXiv preprint arXiv:2308.15427*, 2023.

[21] X. Zhu, X. Cao, Z. Dong, C. Zhou, Q. Liu, W. Li, and Y. Wang, "Nemo: Neural map growing system for spatiotemporal fusion in bird's-eye-view and bdd-map benchmark," *arXiv preprint arXiv:2306.04540*, 2023.

[22] T. Yuan, Y. Liu, Y. Wang, Y. Wang, and H. Zhao, "Streammapnet: Streaming mapping network for vectorized online hd map construction," *arXiv preprint arXiv:2308.12570*, 2023.

[23] S. Li, K. Yang, H. Shi, J. Zhang, J. Lin, Z. Teng, and Z. Li, "Bi-mapper: Holistic bev semantic mapping for autonomous driving," *arXiv preprint arXiv:2305.04205*, 2023.

[24] Y. Wang, V. C. Guizilini, T. Zhang, Y. Wang, H. Zhao, and J. Solomon, "Detr3d: 3d object detection from multi-view images via 3d-to-2d queries," in *Conference on Robot Learning*. PMLR, 2022, pp. 180–191.

[25] 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 *European conference on computer vision*. Springer, 2022, pp. 1–18.

[26] Y. Liu, T. Wang, X. Zhang, and J. Sun, "Petr: Position embedding transformation for multi-view 3d object detection," in *European Conference on Computer Vision*. Springer, 2022, pp. 531–548.

[27] X. Chen, T. Zhang, Y. Wang, Y. Wang, and H. Zhao, "Futr3d: A unified sensor fusion framework for 3d detection," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 172–181.

[28] J. Bennett, *OpenStreetMap*. Packt Publishing Ltd, 2010.

[29] 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 *Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks (NeurIPS Datasets and Benchmarks 2021)*, 2021.

[30] X. Zhu, W. Su, L. Lu, B. Li, X. Wang, and J. Dai, "Deformable DETR: deformable transformers for end-to-end object detection," *CoRR*, vol. abs/2010.04159, 2020. [Online]. Available: https://arxiv.org/abs/2010.04159

[31] T. N. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," *arXiv preprint arXiv:1609.02907*, 2016.

[32] B. Yang, M. Liang, and R. Urtasun, "Hdnet: Exploiting hd maps for 3d object detection," in *Conference on Robot Learning*. PMLR, 2018, pp. 146–155.

[33] J. Gao, C. Sun, H. Zhao, Y. Shen, D. Anguelov, C. Li, and C. Schmid, "Vectornet: Encoding hd maps and agent dynamics from vectorized representation," *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun 2020. [Online]. Available: http://dx.doi.org/10.1109/CVPR42600.2020.01154

[34] A. Simonelli, S. R. Bulo, L. Porzi, P. Kontschieder, and E. Ricci, "Are we missing confidence in pseudo-lidar methods for monocular 3d object detection?" *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, Oct 2021. [Online]. Available: http://dx.doi.org/10.1109/ICCV48922.2021.00321