UniHDMap: Unified Lane Elements Detection for Topology HD Map Construction

Genghua Kou[1]‡, Fan Jia[2], Dongming Wu[1], Yingfei Liu[2], Ying Li[1], Tiancai Wang[2]
[1] Beijing Institute of Technology, [2] MEGVII Technology
koughua@bit.edu.cn, {jiafan, wangtiancai}@megvii.com

Abstract

We proposed a solution for the CVPR 2024 Autonomous Grand Challenge of team CrazyFriday in the track on Mapless Driving. We developed a unified detection framework with an aligned representation named UniHDMap, which is used to detect lanes, pedestrian crossing, and road boundaries, while the popular YOLOv8 is used to detect traffic elements. Finally, two independent MLP-based heads are adopted for lane-lane and lane-traffic topology prediction. We achieved an OULS score of 0.6281.

1. Introduction

Given multi-view images and standard-definition (SD) maps covering the entire field of view, mapless driving requires understanding the 3D scene structure in autonomous driving and analyzing the relationship between perceived entities between traffic elements and centerlines. It consists of five sub-tasks, including lane detection, pedestrian crossing and road boundaries (area) detection, traffic element detection, lane-lane topology, and lane-traffic topology prediction.

In this study, we propose a unified detection framework to share information including lanes, pedestrian crossing, and road boundaries. Specifically, we modify Lanesegnet [4] for road segmentation to output all road detection elements in a unified representation. We also use YOLOv8 for 2D traffic detection. Following TopoMLP [8, 9], we adopt two independent MLP-based heads for lane-lane and lane-traffic topology prediction.

Our proposed framework ranks third on the leaderboard with an OLS score of 0.6281.

2. Method

In this section, we present our model in detail.

2.1. BEV Feature extraction

We use a standard backbone network, such as ResNet-50 backbone [3], to derive feature maps from the original image. Then, the PV to BEV encoder module of BEVFormer [5] is used for view conversion. Transformer-based detection methods use decoders to collect features from BEV features and update decoder queries through multiple layers. At the same time, we inject the vector encoding information of the standard-definition (SD) Map in this step to provide location guidance for map elements. Following SMERF [7], we first convert the SD map to the vectored Polyline sequence representation $\{(x_i, y_i)\}$. After encoding by the transformer encoder, the cross-attention is used to interact with the bev queries. Finally, enhanced BEV features are generated.

2.2. Lane and Area Detection

The lane and area share the same structure to investigate the associations between them. Based on the defined lane query, we aim to employ the transformer decoder to predict the 3D points. Following LaneSegNet [4], we first design a centerline regression branch to regress the vectorized point position of the centerline in 3D coordinates. The output is in the format of $C = N \times 3$. Due to the symmetry of the left and right lane boundaries, we introduce an offset branch to predict the offset $C_{off} = N \times 3$, Therefore, the left and right lane boundary coordinates can be calculated by adding them.

The area shares the same structure with the lane but with different categories. Additionally, in order to accelerate convergence and improve detection accuracy, we adopted the one-to-many [6] method and expanded the number of queries by 5 times as additional supervision.

2.3. Traffic Detection

Following TopoMLP [8], we utilize YOLOv8[1] as the 2D detector, which only takes the front image as input and pre-

‡The work is done during the internship at MEGVII Technology.

[1] The official codes we adopt in our competition solution are available at https://github.com/ultralytics/ultralytics.

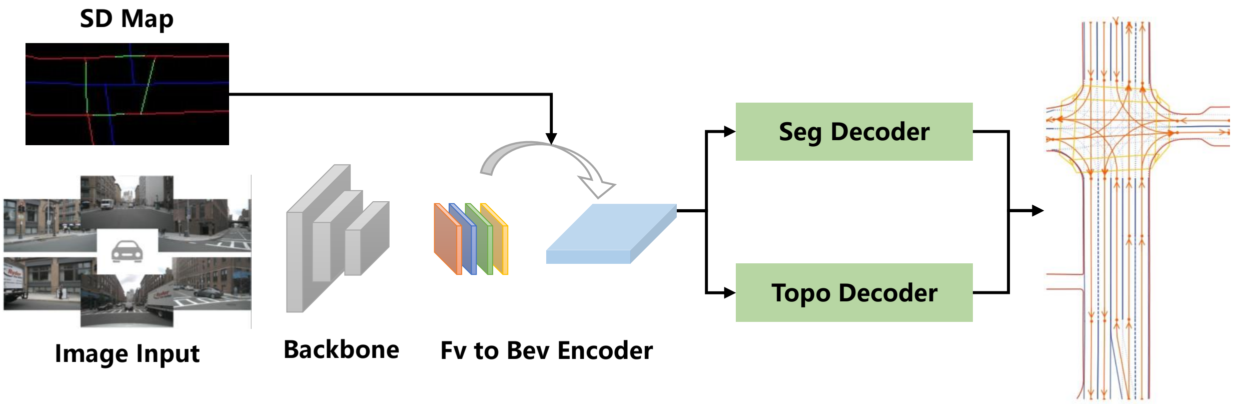

Figure 1. The overall pipeline of lane detection. This process involves inputting multiple images into the backbone, which encodes them into the FV feature. This feature is then transferred to the BEV feature and interacts with the SD map. Various heads decode the enhanced BEV feature to make predictions about lanes and topography.

Method	Backbone	#Epoch	DET_l	DET_a
Ours	ResNet50	24	0.3800	0.3385
Ours	ViT-L	24	0.4309	0.3876

Table 1. Ablation study of different backbones and training epochs over OpenLaneV2 test set, in terms of DET_l and DET_a.

dicts a set of 2D boxes [2].

2.4. Topology Detection

The lane-lane and lane-traffic topology head shared the same architecture. We collect the decoded features and the predicted lane or traffic coordinates from the last decoder layer. The lane coordinates are transformed into the same dimension as the decoded features using an MLP [8], and we sum up both two kinds of features. The summed features are concatenated as the topology size of $N \times M \times 2C$, where C represents the feature dimension. Another MLP further transforms the topology features into a binary topology representation. We apply Focal loss to supervise the topology learning.

3. Experiments

In this section, we first provide some details on implementation. Then we evaluate the additional parts in our method on the OpenLaneV2 test and validation set. The final results of the challenge will be presented as well.

3.1. Implementation Details

For lane detection, the size of all input images is resized to 2048×2048. Our model is implemented with different backbones, including ResNet50 [3] and ViT-L [1]. The entire network is optimized by AdamW optimizer with a learning rate of 2e-4. The number of lane queries is set to 500 balanced for lane and area. The model is trained with 24 epochs if not specialized. For traffic element detection, we statistically analyze the distribution of annotations in the vertical direction of the image and crop them into 896×1550 for training efficiency. We load the COCO-pretrained checkpoint and finetune for 20 epochs as our 2D detector baseline. And others all follow the default settings of YOLOv8-x [2]. Except for the traffic element, other parts are trained together to share the information associated. Notably, we add auxiliary supervision on both BEV and PV.

3.2. Lane and Area Performance with different backbone

We test different backbones on the OpenLane-V2 validation dataset for verifying the scaling up of our model. As shown in Table 1, it can be seen that a stronger backbone brings much better performance.

Methods	DET_l	DET_a
baseline	0.3514	0.3178
+ auxiliary SDMap	0.3820	0.3398
+ auxiliary BEV supe.	0.3956	0.3472
+ one-to-many matching	0.4117	0.3395
+ auxiliary PV supe.	0.4256	0.3580

Table 2. The lane detection performance DET_l and DET_a on OpenLaneV2 validation set with backbone ResNet-50.

3.3. Ablation study

We evaluate with different enhancement methods on the OpenLane-V2 validation dataset in Table 2, including SD map injection, auxiliary PV supervise (PV supe.), auxiliary BEV supervise (BEV supe.), and one-to-many matching [6]. We gradually added auxiliary dense supervision

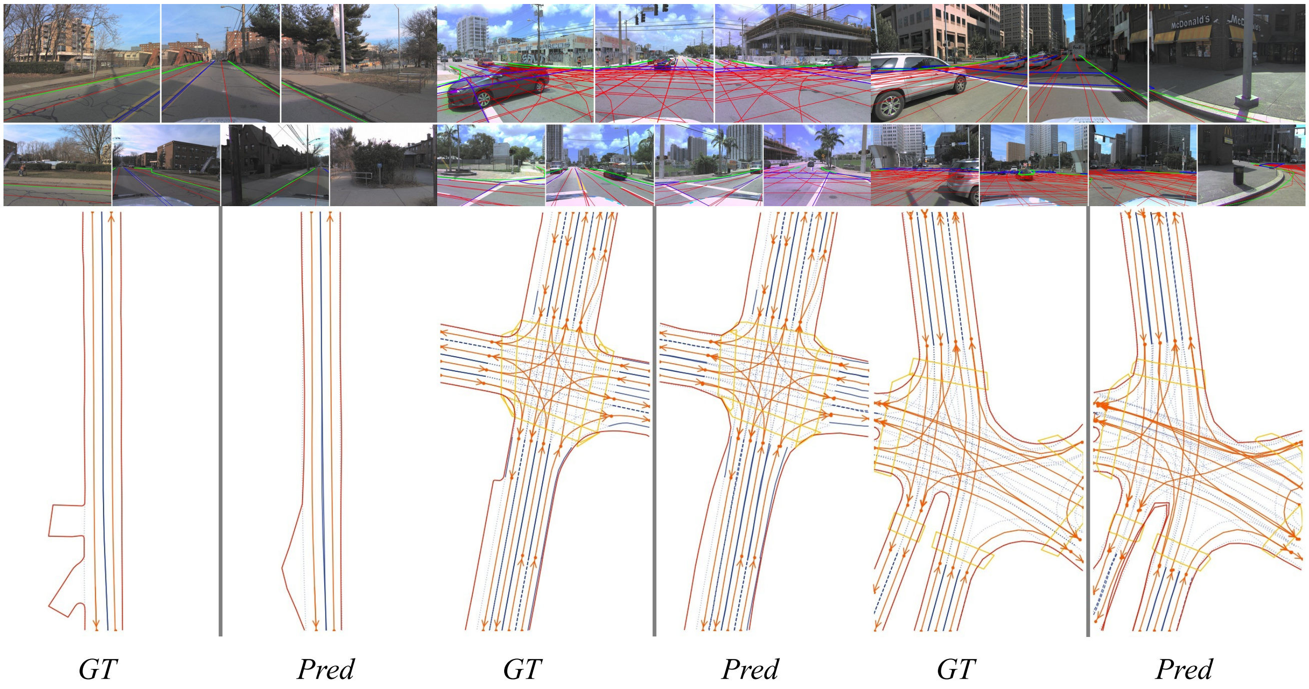

Figure 2. The illustration of prediction results from our model on the validation set.

DET_l	DET_a	DET_t	TOP_{ll}	TOP_{lt}	UniScore
0.4994	0.4638	0.7927	0.4392	0.5211	0.6281

Table 3. The final leaderboard of mapless challenge.

and auxiliary one-to-many matching. With all these components, our framework achieved a performance increase.

3.4. Final Result

The final result of UniHDmap in the CVPR 2024 Autonomous Grand Challenge on Mapless Driving is shown in Table 3 and Figure 2. Note that the model for final submission is jointly trained on the training and validation sets of OpenlaneV2 but is not ensembled.

References

[1] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020. 2

[2] Zheng Ge, Songtao Liu, Feng Wang, Zeming Li, and Jian Sun. Yolox: Exceeding yolo series in 2021. *arXiv preprint arXiv:2107.08430*, 2021. 2

[3] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 1, 2

[4] Tianyu Li, Peijin Jia, Bangjun Wang, Li Chen, Kun Jiang, Junchi Yan, and Hongyang Li. Lanesegnet: Map learning with lane segment perception for autonomous driving. *arXiv preprint arXiv:2312.16108*, 2023. 1

[5] Zhiqi Li, Wenhai Wang, Hongyang Li, Enze Xie, Chonghao Sima, Tong Lu, Yu Qiao, and Jifeng Dai. Bevformer: Learning bird's-eye-view representation from multi-camera images via spatiotemporal transformers. In *European conference on computer vision*, pages 1–18. Springer, 2022. 1

[6] Bencheng Liao, Shaoyu Chen, Yunchi Zhang, Bo Jiang, Qian Zhang, Wenyu Liu, Chang Huang, and Xinggang Wang. Maptrv2: An end-to-end framework for online vectorized hd map construction. *arXiv preprint arXiv:2308.05736*, 2023. 1, 2

[7] Katie Z Luo, Xinshuo Weng, Yan Wang, Shuang Wu, Jie Li, Kilian Q Weinberger, Yue Wang, and Marco Pavone. Augmenting lane perception and topology understanding with standard definition navigation maps. *arXiv preprint arXiv:2311.04079*, 2023. 1

[8] Dongming Wu, Jiahao Chang, Fan Jia, Yingfei Liu, Tiancai Wang, and Jianbing Shen. Topomlp: An simple yet strong pipeline for driving topology reasoning. *ICLR*, 2024. 1, 2

[9] Dongming Wu, Fan Jia, Jiahao Chang, Zhuoling Li, Jianjian Sun, Chunrui Han, Shuailin Li, Yingfei Liu, Zheng Ge, and Tiancai Wang. The 1st-place solution for cvpr 2023 openlane topology in autonomous driving challenge. *arXiv preprint arXiv:2306.09590*, 2023. 1