Separated RoadTopoFormer

Mingjie Lu*, Yuanxian Huang*, Ji Liu, Jinzhang Peng, Lu Tian, Ashish Sirasao
Advanced Micro Devices, Inc., Beijing, China
(Mingjie.Lu, YuanXian.Huang, Ji.Liu, jinz.peng, lu.tian, ashish.sirasao)@amd.com

Abstract

Understanding the driving scenarios is crucial to realize autonomous driving. Previous works such as map learning and bev lane detection neglect the connection relationship between lane instances, and traffic elements detection tasks usually neglect the relationship with lane lines. To address these issues, the task is presented which includes 4 sub-tasks, the detection of traffic elements, the detection of lane centerlines, reasoning connection relationships among lanes, and reasoning assignment relationships between lanes and traffic elements. We present Separated RoadTopoFormer to tackle the issues, which is an end-to-end framework that detects lane centerline and traffic elements with reasoning relationships among them. We optimize each module separately to prevent interaction with each other and aggregate them together finally with few finetunes. For two detection heads, we adopted DETR-like architecture to detect objects, and for the relationship head, we concat two instance features from front detectors and feed them to the classifier to obtain relationship probability. Our final submission achieves 0.445 OLS which is competitive in both sub-task and combined scores.

1. Introduction

In recent years, the availability of public large-scale datasets and benchmarks has greatly facilitated autonomous driving research. Many datasets [2, 9] focus on sensing visible lane lines to keep vehicles on the right track only, or to obtain traffic information by detecting traffic signals only. However, the separation of tasks leads to a limited understanding of driving scenarios. For example, a driving vehicle will confuse when it sees a green light but the lane it follows is controlled by another red light. Based on this limitation, a key aspect of this task [8] is to understand the complex driving environment, which is a prerequisite for making reasonable decisions. On the one hand, this task wants to establish a strong association between traffic elements and lanes. On the other hand, understanding the separations between neighboring lanes is also necessary for guiding the vehicle driving on the desired trajectory. Both topology reasoning tasks are extremely challenging.

This task can be divided into two parts simply, which are scene structure perception and reasoning. The scene structure perception aims to find out what and where the traffic elements and lanes are and the reasoning aims to understand the relationship between them. The latter is highly dependent on the former, but the reverse is not certain. So we optimize each module separately to prevent interactions during training, and finally integrate them by finetuning. Experiments prove it works. We also have made other experimental improvements, please refer to Section 3.

2. Datasets

Road Genome, also known as OpenLane-V2 [8], is the first dataset focusing on topology reasoning in the autonomous driving area. It contains 2.1M instance-level annotations and 1.9M positive topology relationships. This challenge is based on subset_A, which contains 22477 training frames, 4806 val frames, and 4816 test frames. Each frame contains 6 surrounding images in the resolution 1550×2048 and a front-view image in the resolution 2048×1550. The final metric is OpenLane-V2 Score (OLS), which is the average of various metrics from different subtasks and is defined to describe the overall performance of the primary task: $OLS = \frac{1}{4}[DET_l + DET_t + f(TOP_{ll}) + f(TOP_{lt})]$, where f is a scaling function that balances the scale of different metrics.

3. Methods

3.1. Baseline

The official baseline [8] provides a simple and easy-to-follow framework that generates two feature maps from different views. One is in BEV (Bird's-eye view) and the other one is in PV (Perspective view). The former is used to predict lane centerlines (LCs) and the latter is for traffic elements (TEs) prediction. Two detection heads adopt similar DERT-like architectures. The following two relationship

*These authors contributed equally to this work.

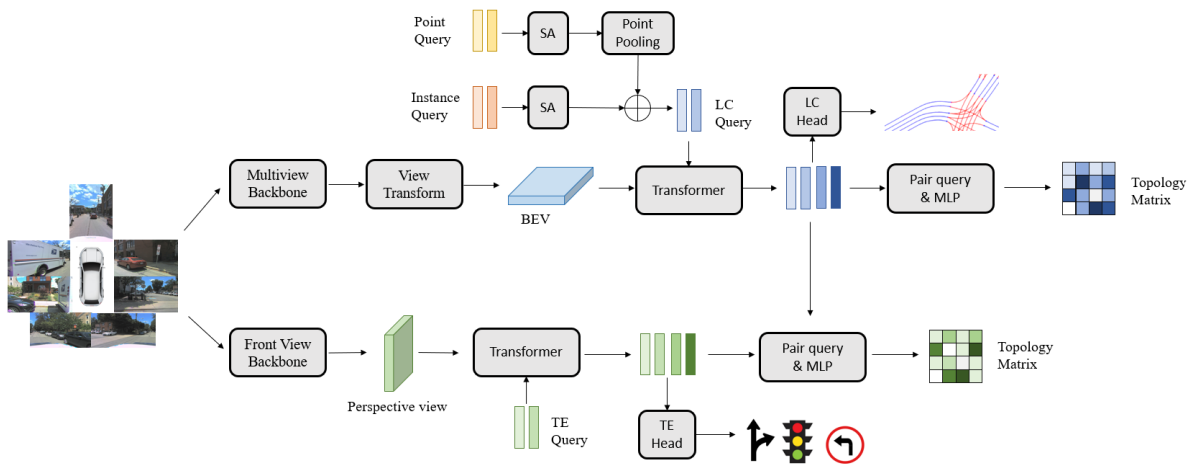

Figure 1. Overall framework.

Method	DET$_l$ (%)
Baseline	9.57
+ decoupled training	10.59
+ 11 points representation	13.58
+ swin-s & rescale	22.19
+ finetune with smaller lr	23.44
+ hierarchical query	23.80
+ intersection-sensitive	25.87
+ finetune whole model	26.95

Table 1. Ablation of 3D centerline detection on OpenLanev2 validation set.

Method	DET$_t$ (%)
Baseline	45.89
+ swin-s & decoupled training	58.41
+ DINO head	60.23
+ finetune whole model	61.42

Table 2. Ablation of traffic element detection on OpenLanev2 validation set.

prediction modules based on instance-level representations from front detected results to establish pairwise relationships which contain a $L \times L$ lanes relationship matrix and a $L \times T$ lanes-traffic elements relationship matrix, where L and T represent the numbers of LCs and TEs, respectively. Then two subsequent MLPs are used to predict the logits of two kinds of relationships, respectively.

Method	TOP$_{ll}$ (%)
Baseline	0.92
+ better LC detector	3.90
+ geometric clues	14.26
+ finetune whole model	15.37

Table 3. Ablation of topology prediction between lane centerlines on OpenLanev2 validation set.

3.2. Architecture

The design of our algorithm follows Road Genome [8]. However, unlike the Road Genome, our TE branch and LC branch do not share a common backbone as demonstrated in Figure 1. Instead, each branch has an independent backbone network to extract features. This modification allows for independent feature learning and data augmentation for two detection tasks.

Lane centerline detection. Given multi-view images, we first use a shared Swin-small [7] backbone to extract features from each view's image. Then, we apply BEVFormer [6] to transform the multi perspective view features into a unified BEV feature. Later, a Deformable DETR-like [11] transformer is utilized to extract query-wise information of the 3D lane centerlines based on the BEV feature. Finally, each output query is passed through an LC head to predict the confidence of a line and the coordination of 11 equally spaced 3D points in the centerline. The coordination of each 3D point is normalized according to the detection range.

Traffic element detection. We utilize a separated and independent Swin-small backbone to extract the perspective view feature from the front center image. DINO [10] head

Method	DET_l (%)	DET_t (%)	TOP_{ll} (%)	TOP_{lt} (%)	OLS (%)
Baseline [8]	9.57	45.89	0.92	11.46	24.72
Ours	26.95	61.42	15.37	21.81	43.57

Table 4. Submission results on OpenLanev2 validation set.

Method	DET_l (%)	DET_t (%)	TOP_{ll} (%)	TOP_{lt} (%)	OLS (%)
TopoNet [5]	19	58	2	16	33
Ours	22	72	13	23	45

Table 5. Submission results on OpenLanev2 test set.

is employed to detect 2D traffic elements.

Topology prediction. We follow the design of topology prediction in STSU [1]. Every two objects' query will be concatenated. The concatenated feature will pass through an MLP and a sigmoid layer, and output a relationship confidence. Only if the confidence is bigger than 0.5, the two objects will be considered as having a topology relationship. Instead of considering all queries like baseline, we only consider the query whose confidence is bigger than a prior threshold.

3.3. Bells and whistles

Hierarchical query. For 3D centerlines detection, the locations of points are significant for the final performance, we design two kinds of queries, i.e., point query and instance query, to make the query input transformer decoder have better representation ability. Point queries $Q_p \in \mathbb{R}^{N_p \times D}$ and instance queries $Q_I \in \mathbb{R}^{N \times D}$ are first passed through a self-attention model to model relationship between queries, where N_p represents the number of point queries which is set to 11 to be equal to the final output number of points, N represents the max number of centerlines, and D represents the dimension of the embeddings. To aggregate the feature of both two kinds of queries, a point pooling module is proposed to get a global feature across point queries. We utilize the sum operation to pool the point queries. Finally, LC query Q_{LC} is obtained by adding each instance query to the 3D global pooling point feature.

$$Q_{pooled} = PointPooling(Q_p) = \sum_{i=1}^{N_p} Q_{p,i}, Q_{p,i} \in \mathbb{R}^D \tag{1}$$

$$Q_{LC,i} = Q_{I,i} + Q_{pooled} \tag{2}$$

Intersection-sensitive classification head. The OpenLanev2 [8] dataset contains two kinds of centerline, i.e., normal lane centerline and connecting line in intersections, which are evidently different. In consist of normal lane centerlines, connecting lines in the intersection are more related to the overall structure of the road, and less related to the local texture features. Therefore, we distinguish these two categories in the classification head in the LC head. As shown in Tabel 1, this simple strategy improves 2.43% DET_l metric.

Swin backbone and input resolution. Because the input image size of the baseline [8] is the original resolution of the image which is 1550x2048, the batch size of each our GPU can be set to 1 when training the whole model. However, the backbone of the baseline is ResNet50 [4] and utilizes BatchNorm, which is inappropriate when the batch size is set to one. Therefore, we utilize Swin-small [7] as our backbone for both LC branch and TE branch which apply LayerNorm instead of BatchNorm. Besides, to speed up the training and save device memory, we resize the multi-view images to 775x1024. For the front view image, we keep its size as its original resolution (2048x1550), because its overhead is affordable. The backbones in both two branches are pre-trained in ImageNet1K [3].

11 points representation. Instead of representing the 3D line as 5 Brazier control points like STSU [1], we directly model the 3D line as 11 equally spaced keypoints in its skeleton. We found this simple representation is surprisingly better than the Brazier curve. Results are shown in Table 1.

DINO TE detector. We use DINO [10] detector head instead of original deformable-detr of baseline with 900 queries. As show in Table 2, DINO brings about 2% gain for traffic elements detection.

Geometric clues for relationship prediction between centerlines. The topological relationship between centerlines is not only related to semantic information but also associated with their geometric locations. If the endpoints of the centerlines of two lanes are very close, then there is a high probability that they are topologically related. Therefore, we introduce geometric clues for relationship predic-

tion between centerlines in two aspects. First, we concatenate the LC query with its start point and end point which are predicted by the LC regression head. Second, any two lane centerlines whose start and end points are less than 3 meters apart will be considered to have a topological relationship, even if their relationship confidence is less than 0.5. Results are shown in Table 3.

Decoupled training and integrated finetuning. Instead of training all modules of the whole network simultaneously, we decouple different modules and train only one of them each time. Specifically, we first independently train the LC module and TE module. Then, two relationship heads are trained with frozen backbones and detection heads. The decoupled training strategy helps us quickly verify an improvement idea for a single module. Meanwhile, this strategy enables each module to perform its own duties and avoids the impact between different tasks. After all modules are trained independently, we finetune the whole network with a smaller learning rate. During finetuning, only four heads are unfrozen, including the LC head, TE head, and two relationship heads. In the decoupled training set, we follow the training setting in Road Genome [8], including the optimizer, the learning rate update schedule, and so on. The learning rate will be adjusted proportionally with the batch size. In the finetuning stage, we set a smaller learning rate, which is a quarter of the decoupled training stage.

4. Final Results

For the final submission, we apply all the aforementioned strategies for performance improvement. The performances on OpenLanev2 validation and test set are demonstrated in Table 4 and 5, respectively.

References

[1] Yigit Baran Can, Alexander Liniger, Danda Pani Paudel, and Luc Van Gool. Structured bird's-eye-view traffic scene understanding from onboard images. *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 15641–15650, 2021. 3

[2] Li Chen, Chonghao Sima, Yang Li, Zehan Zheng, Jiajie Xu, Xiangwei Geng, Hongyang Li, Conghui He, Jianping Shi, Yu Qiao, and Junchi Yan. Persformer: 3d lane detection via perspective transformer and the openlane benchmark. In *ECCV*, 2022. 1

[3] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 248–255, 2009. 3

[4] Kaiming He, X. Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778, 2015. 3

[5] Tianyu Li, Li Chen, Xiangwei Geng, Huijie Wang, Yang Li, Zhenbo Liu, Shengyin Jiang, Yuting Wang, Hang Xu, Chunjing Xu, Feng Wen, Ping Luo, Junchi Yan, Wei Zhang, Xiaogang Wang, Yu Qiao, and Hongyang Li. Topology reasoning for driving scenes. *arXiv preprint arXiv:2304.05277*, 2023. 3

[6] 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. *arXiv preprint arXiv:2203.17270*, 2022. 2

[7] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2021. 2, 3

[8] Huijie Wang, Zhenbo Liu, Yang Li, Tianyu Li, Li Chen, Chonghao Sima, Yuting Wang, Shengyin Jiang, Feng Wen, Hang Xu, Ping Luo, Junchi Yan, Wei Zhang, Jun Yao, Yu Qiao, and Hongyang Li. Road genome: A topology reasoning benchmark for scene understanding in autonomous driving. *arXiv preprint arXiv:2304.10440*, 2023. 1, 2, 3, 4

[9] Fisher Yu, Haofeng Chen, Xin Wang, Wenqi Xian, Yingying Chen, Fangchen Liu, Vashisht Madhavan, and Trevor Darrell. Bdd100k: A diverse driving dataset for heterogeneous multitask learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 2636–2645, 2020. 1

[10] Hao Zhang, Feng Li, Shilong Liu, Lei Zhang, Hang Su, Jun Zhu, Lionel Ni, and Heung-Yeung Shum. DINO: DETR with improved denoising anchor boxes for end-to-end object detection. In *The Eleventh International Conference on Learning Representations*, 2023. 2, 3

[11] Xizhou Zhu, Weijie Su, Lewei Lu, Bin Li, Xiaogang Wang, and Jifeng Dai. Deformable detr: Deformable transformers for end-to-end object detection. *arXiv preprint arXiv:2010.04159*, 2020. 2