Supplementary Materials for LaneDiffusion: Improving Centerline Graph Learning via Prior Injected BEV Feature Generation

Zijie Wang[1,2]*, Weiming Zhang[3]*, Wei Zhang[3]*, Xiao Tan[3], Hongxing Liu[3], Yaowei Wang[4,5], Guanbin Li[1,2,5,6]†

[1]Sun Yat-sen University, [2]Shenzhen Loop Area Institute, [3]Baidu Inc.
[4]Harbin Institute of Technology, Shenzhen, [5]Pengcheng Laboratory
[6]Guangdong Key Laboratory of Big Data Analysis and Processing

wangzj75@mail2.sysu.edu.cn, liguanbin@mail.sysu.edu.cn

In this manuscript, we provide additional details that could not be included in the main paper due to space constraints, including: (A) the derivation of Eq. (9) in the main paper; (B) further implementation details and ablation analysis of the lane prior refinement encoder-decoder structure; and (C) detailed descriptions of both the segment-level and fine-grained point-level metrics.

A. Derivation of Eq. (9)

According to Bayes's theorem, we have

$$q(\mathbf{x}_{t-1}|\mathbf{x}_t,\mathbf{x}_0,\mathbf{x}_\mathrm{c}) \propto q(\mathbf{x}_t|\mathbf{x}_{t-1},\mathbf{x}_\mathrm{c})q(\mathbf{x}_{t-1}|\mathbf{x}_0,\mathbf{x}_\mathrm{c}), \quad (1)$$

where

$$q(\mathbf{x}_t|\mathbf{x}_{t-1},\mathbf{x}_\mathrm{c}) = \mathcal{N}(\mathbf{x}_t;\mathbf{x}_{t-1}+\gamma_t\mathbf{x}_\mathrm{res},\kappa^2\gamma_t\mathbf{I}), \quad (2)$$

$$q(\mathbf{x}_{t-1}|\mathbf{x}_0,\mathbf{x}_\mathrm{c}) = \mathcal{N}(\mathbf{x}_{t-1};\mathbf{x}_0+\eta_{t-1}\mathbf{x}_\mathrm{res},\kappa^2\eta_{t-1}\mathbf{I}). \quad (3)$$

Now, considering the quadratic form in the exponent of $q(\mathbf{x}_{t-1}|\mathbf{x}_t,\mathbf{x}_0,\mathbf{x}_\mathrm{c})$, we have

$$\begin{aligned}
&-\frac{(\mathbf{x}_t-\mathbf{x}_{t-1}-\gamma_t\mathbf{x}_\mathrm{res})(\mathbf{x}_t-\mathbf{x}_{t-1}-\gamma_t\mathbf{x}_\mathrm{res})^T}{2\kappa^2\gamma_t}\\
&-\frac{(\mathbf{x}_{t-1}-\mathbf{x}_0-\eta_{t-1}\mathbf{x}_\mathrm{res})(\mathbf{x}_{t-1}-\mathbf{x}_0-\eta_{t-1}\mathbf{x}_\mathrm{res})^T}{2\kappa^2\eta_{t-1}}\\
&=-\frac{1}{2}\left[\frac{1}{\kappa^2\gamma_t}+\frac{1}{\kappa^2\eta_{t-1}}\right]\mathbf{x}_{t-1}\mathbf{x}_{t-1}^T \qquad (4)\\
&+\left[\frac{\mathbf{x}_t-\gamma_t\mathbf{x}_\mathrm{res}}{\kappa^2\gamma_t}+\frac{\mathbf{x}_0+\eta_{t-1}\mathbf{x}_\mathrm{res}}{\kappa^2\eta_{t-1}}\right]\mathbf{x}_{t-1}^T+\mathrm{const}\\
&=-\frac{(\mathbf{x}_{t-1}-\boldsymbol{\mu})(\mathbf{x}_{t-1}-\boldsymbol{\mu})^T}{2\lambda^2}+\mathrm{const},
\end{aligned}$$

where

$$\boldsymbol{\mu} = \frac{\eta_{t-1}}{\eta_t}\mathbf{x}_t+\frac{\gamma_t}{\eta_t}\mathbf{x}_0, \quad (5)$$

$$\lambda^2 = \kappa^2\frac{\eta_{t-1}}{\eta_t}\gamma_t, \quad (6)$$

and ”const” denotes terms that are independent of $\mathbf{x}_{t-1}$. This quadratic form results in the Gaussian distribution in Eq. (9) of the main paper.

B. Implementation of Lane Prior Refinement and More Ablation Analysis

For the lane prior refinement mechanism in the Lane Prior Diffusion Module (LPDM), we employ an encoder-decoder architecture. Our ablation experiments in the main paper indicate that feature concatenation and feature addition yield similar performance. Therefore, here we use the concatenation-based approach as an example. In this paradigm, the generated BEV feature $\mathbf{g}$ and the original feature $\mathbf{x}_\mathrm{c}$ are concatenated and then processed sequentially by an encoder built from stacked convolutional blocks and a decoder composed of stacked inverse convolutional blocks. To further investigate the impact of different encoder-decoder configurations, we conduct additional ablation studies. Specifically, we explore two settings: (i) *Spatial&Channel*, in which both the spatial dimensions and channel dimensions are modified during the calculation, and (ii) *Channel Only*, where only the channel dimension is altered. The corresponding variations in the intermediate feature shapes are shown in Tab. 1. Note that extra residual blocks, which do not change the feature shape, are added at the end of the encoder and at the beginning of the decoder. A comparison of these two settings is reported in Tab. 2, revealing that the *Channel Only* setting achieves slightly better performance. Therefore, we adopt this configuration as our final setting.

*Equal contribution. Work done during an internship at Baidu.
†Corresponding author.

Shape Method Stage	Spatial&Channel	Channel Only
Input	$512 \times 200 \times 100$	$512 \times 200 \times 100$
Encoder	$256 \times 200 \times 100$	$256 \times 200 \times 100$
	$128 \times 100 \times 50$	$128 \times 200 \times 100$
	$128 \times 50 \times 25$	$128 \times 200 \times 100$
Decoder	$128 \times 50 \times 25$	$128 \times 200 \times 100$
	$128 \times 100 \times 50$	$128 \times 200 \times 100$
	$256 \times 200 \times 100$	$256 \times 200 \times 100$
	$512 \times 200 \times 100$	$512 \times 200 \times 100$
Output	$256 \times 200 \times 100$	$256 \times 200 \times 100$

Table 1. The variation in the shape of the intermediate features under the two settings.

Method	kernel size	stride	padding	TOPO F1 ↑	JTOPO F1 ↑	APLS ↑	SDA ↑
Spatial&Channel	4	2	1	46.3	38.5	36.9	10.1
Channel Only	5	1	2	**46.8**	**38.8**	**37.1**	**10.6**

Table 2. Ablation analysis for the structure of the lane prior refinement encoder-decoder.

C. Details of Metrics

In this section, we provide a detailed overview of all metrics referenced in the main paper.

C.1. Segment-level Metrics

Segment-level metrics evaluate the segment-level graph $G = (V, E)$, where the vertices V represent centerline segments and the edges $E \subseteq \{(x, y) \mid (x, y) \in V^2\}$ denote the connectivity among these segments.

IoU [4]. This metric measures the intersection-over-union between predicted and ground truth centerline segments. It is defined as:

$$\mathrm{IoU}(\boldsymbol{\mathcal{B}}_1, \boldsymbol{\mathcal{B}}_2) = \frac{\boldsymbol{\mathcal{B}}_1 \cap \boldsymbol{\mathcal{B}}_2}{\boldsymbol{\mathcal{B}}_1 \cup \boldsymbol{\mathcal{B}}_2}, \tag{7}$$

where $\boldsymbol{\mathcal{B}}_1, \boldsymbol{\mathcal{B}}_2 \in \mathbb{R}^{60\times30}$ are dense representations (i.e., curves rasterized on a grid) of the predicted and ground truth shapes, respectively.

mAP$_{cf}$ [6]. This metric evaluates the quality of centerline segment construction by using the Chamfer distance to determine if a predicted segment matches a ground truth segment. Predefined Chamfer distance thresholds $T = \{0.5, 1.0, 1.5\}$ are used to compute the mean average precision:

$$\mathrm{mAP}_{cf} = \frac{1}{|T|} \sum_{t \in T} AP_t. \tag{8}$$

DET$_l$ [8]. Similar to mAP$_{cf}$, this metric employs the discrete Fréchet distance to assess the matching quality between predicted and ground truth segments. The DET$_l$ score is averaged over the match thresholds $T = \{1.0, 2.0, 3.0\}$:

$$\mathrm{DET}_l = \frac{1}{|T|} \sum_{t \in T} AP_t. \tag{9}$$

TOP$_{ll}$ [8]. This metric measures the topological correctness of the centerline segments. Given a ground truth graph $G = (V, E)$ and a predicted graph $\hat{G} = (\hat{V}, \hat{E})$, matching between the ground truth vertices V and the predicted vertices $\hat{V}$ is established using the Fréchet distance as a similarity measure. We define the matching vertex set $\hat{V}'$, which satisfies $V = \hat{V}'$ and $\hat{V}' \subseteq \hat{V} \cup \{v_d\}$, where $\{v_d\}$ denotes a set of dummy vertices for unmatched elements. The TOP$_{ll}$ metric is computed as the mean average precision (mAP) over all vertices:

$$\mathrm{TOP}_{ll} = \frac{1}{|V|} \sum_{v \in V} \frac{\sum_{\hat{n}' \in N'(v)} P(\hat{n}')\, \mathbb{1}(\hat{n}' \in N(v))}{|N(v)|}, \tag{10}$$

where $N(v)$ is the ordered list of neighbors of vertex v ranked by confidence, $P(\hat{n}')$ is the precision of vertex $\hat{n}'$, and positive edges are those with confidence greater than 0.5.

C.2. Point-level Metrics

To better assess the quality of the predicted centerline graph (particularly its continuity) we adopt fine-grained point-level metrics following CGNet [2]. For this purpose, we construct a point-level graph $\ddot{G} = (\ddot{V}, \ddot{E})$, where $\ddot{V}$ comprises all points from the polylines and $\ddot{E}$ denotes the connectivity between these points.

GEO Metric [3]. This metric evaluates the positional accuracy of points. Given the ground truth graph $\ddot{G} = (\ddot{V}, \ddot{E})$ and the predicted graph $\hat{\ddot{G}} = (\hat{\ddot{V}}, \hat{\ddot{E}})$, we first interpolate (densify) both graphs so that the distance between any two connected vertices is 0.25 m. A vertex pair $(v \in \ddot{V}, \hat{v} \in \hat{\ddot{V}})$ is considered a valid match if the distance between them is less than 0.5 m. The maximal one-to-one matching is then determined, yielding the set of matched vertices $\ddot{V}_m$. The GEO precision and recall are computed as follows:

$$\text{Precision}_{\text{GEO}} = \frac{|\ddot{V}_m|}{|\hat{\ddot{V}}|}, \tag{11}$$

$$\text{Recall}_{\text{GEO}} = \frac{|\ddot{V}_m|}{|\ddot{V}|}. \tag{12}$$

Finally, the F1-score is calculated by

$$\text{F1}_{\text{GEO}} = \frac{2 \times \text{Precision}_{\text{GEO}} \times \text{Recall}_{\text{GEO}}}{\text{Precision}_{\text{GEO}} + \text{Recall}_{\text{GEO}}}. \tag{13}$$

TOPO Metric [3]. While the GEO metric treats each point independently, it does not consider connectivity. The TOPO metric incorporates topology by extending the GEO evaluation. For each matched vertex pair $(v, \hat{v}) \in \mathbb{S}$ determined by the GEO metric, we traverse the graph for a distance of less than 8.0 m to construct sub-graphs S_v and $\hat{S}_{\hat{v}}$ from $\ddot{G}$ and $\hat{\ddot{G}}$, respectively. The GEO metric is then computed between these sub-graphs. The TOPO precision and recall are given by:

$$\text{Precision}_{\text{TOPO}} = \frac{\sum_{(v,\hat{v}) \in \mathbb{S}} \text{Precision}_{\text{GEO}}(S_v, \hat{S}_{\hat{v}})}{|\hat{\ddot{V}}|}, \tag{14}$$

$$\text{Recall}_{\text{TOPO}} = \frac{\sum_{(v,\hat{v}) \in \mathbb{S}} \text{Recall}_{\text{GEO}}(S_v, \hat{S}_{\hat{v}})}{|\ddot{V}|}. \tag{15}$$

The F1-score is then calculated as

$$\text{F1}_{\text{TOPO}} = \frac{2 \times \text{Precision}_{\text{TOPO}} \times \text{Recall}_{\text{TOPO}}}{\text{Precision}_{\text{TOPO}} + \text{Recall}_{\text{TOPO}}}. \tag{16}$$

JTOPO Metric [5]. The TOPO metric evaluates the overall topological correctness, but junction points—where lanes merge or fork—are particularly critical as they better reflect continuity. Junction points are defined as vertices with an out-degree or in-degree greater than 1. The JTOPO metric is a variant of the TOPO metric that specifically selects junction points for sub-graph construction and evaluation.

APLS [7]. Based on Dijkstra's shortest path algorithm, this metric sums the differences in optimal path lengths between nodes in the ground truth graph $\ddot{G}$ and the predicted graph $\hat{\ddot{G}}$, and is formulated as:

$$\text{APLS} = 1 - \frac{1}{N} \sum \min \left\{ 1, \frac{|d(a,b) - d(a',b')|}{d(a',b')} \right\}, \tag{17}$$

where N is the number of paths, $d(a, b)$ is the length of the path from vertex a to vertex b, and a' is the node in the predicted graph closest to the ground truth node a.

SDA [1]. This metric evaluates the accuracy of predicted junction points within a circular region around the ground truth junctions. The Hungarian algorithm is used to determine the optimal assignment of junction points between the ground truth graph $\ddot{G}$ and the predicted graph $\hat{\ddot{G}}$. A pair of points is considered a true positive if the distance between them is less than 1.0 m. The F1 score is then computed based on the number of true positives.

References

[1] Martin Büchner, Jannik Zürn, Ion-George Todoran, Abhinav Valada, and Wolfram Burgard. Learning and aggregating lane graphs for urban automated driving. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13415–13424, 2023.

[2] Yunhui Han, Kun Yu, and Zhiwei Li. Continuity preserving online centerline graph learning. In *European Conference on Computer Vision*, pages 342–359. Springer, 2024.

[3] Songtao He and Hari Balakrishnan. Lane-level street map extraction from aerial imagery. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 2080–2089, 2022.

[4] Qi Li, Yue Wang, Yilun Wang, and Hang Zhao. Hdmapnet: An online hd map construction and evaluation framework. In *2022 International Conference on Robotics and Automation (ICRA)*, pages 4628–4634. IEEE, 2022.

[5] Bencheng Liao, Shaoyu Chen, Bo Jiang, Tianheng Cheng, Qian Zhang, Wenyu Liu, Chang Huang, and Xinggang Wang. Lane graph as path: Continuity-preserving path-wise modeling for online lane graph construction. In *European Conference on Computer Vision*, pages 334–351. Springer, 2024.

[6] Yicheng Liu, Tianyuan Yuan, Yue Wang, Yilun Wang, and Hang Zhao. Vectormapnet: End-to-end vectorized hd map learning. In *International Conference on Machine Learning*, pages 22352–22369. PMLR, 2023.

[7] Adam Van Etten, Dave Lindenbaum, and Todd M Bacastow. Spacenet: A remote sensing dataset and challenge series. *arXiv preprint arXiv:1807.01232*, 2018.

[8] Huijie Wang, Tianyu Li, Yang Li, Li Chen, Chonghao Sima, Zhenbo Liu, Bangjun Wang, Peijin Jia, Yuting Wang, Shengyin Jiang, et al. Openlane-v2: A topology reasoning benchmark for unified 3d hd mapping. *Advances in Neural Information Processing Systems*, 36, 2024.