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

Advancements in 3D Lane Detection Using LiDAR Point Clouds: From Data Collection to Model Development

Runkai Zhao[1‡], Yuwen Heng[2], Heng Wang[1], Yuanda Gao[2], Shilei Liu[2], Changhao Yao[3‡], Jiawen Chen[2] and Weidong Cai[1]

***Abstract*— Advanced Driver-Assistance Systems (ADAS) have successfully integrated learning-based techniques into vehicle perception and decision-making. However, their application in 3D lane detection for effective driving environment perception is hindered by the lack of comprehensive LiDAR datasets. The sparse nature of LiDAR point cloud data prevents an efficient manual annotation process. To solve this problem, we present *LiSV-3DLane*, a large-scale 3D lane dataset that comprises 20k frames of surround-view LiDAR point clouds with enriched semantic annotation. Unlike existing datasets confined to a frontal perspective, *LiSV-3DLane* provides a full 360-degree spatial panorama around the ego vehicle, capturing complex lane patterns in both urban and highway environments. We leverage the geometric traits of lane lines and the intrinsic spatial attributes of LiDAR data to design a simple yet effective automatic annotation pipeline for generating finer lane labels. To propel future research, we propose a novel LiDAR-based 3D lane detection model, *LiLaDet*, incorporating the spatial geometry learning of the LiDAR point cloud into Bird's Eye View (BEV) based lane identification. Experimental results indicate that *LiLaDet* outperforms existing camera- and LiDAR-based approaches in the 3D lane detection task on the K-Lane dataset and our *LiSV-3DLane*. The project code will be available at `https://github.com/RunkaiZhao/LiLaDet`.**

I. INTRODUCTION

3D lane detection, which aims at providing accurate localization of lane lines in the real-world 3D coordinate system, offers spatial awareness for autonomous navigation and collision avoidance. Detecting lane lines in a surround view could provide a comprehensive understanding of traffic scenarios. LiDAR systems typically offer a 360-degree view of the environment, capturing lane information from all directions as shown in Fig. 1(a) and (b). However, the existing LiDAR-based 3D lane dataset [1] only focuses on the frontal view due to the high cost of LiDAR data collection and labor-intensive annotation process caused by the sparse nature of point clouds [2], [3]. To mitigate these challenges, this paper introduces a dedicated procedure for constructing a comprehensive LiDAR-based surround-view 3D Lane dataset from scratch.

Existing camera-based 3D lane detection methods are ill-posed as 2D images cannot be converted to 3D representations without depth information. [4]–[9] assume that the ground is flat and assign zero height to all lanes. This planar assumption cannot be generalized to real-world driving conditions such as non-linear terrains with slopes and bumps. On the other hand, LiDAR-based 3D lane detection methods [1], [10] project LiDAR points into a Bird's Eye View (BEV) grid image and identify line lanes by semantic segmentation, which can generally display lane shapes but fail to capture accurate 3D information due to the reduced spatial details of the voxelization process as illustrated in Fig. 2.

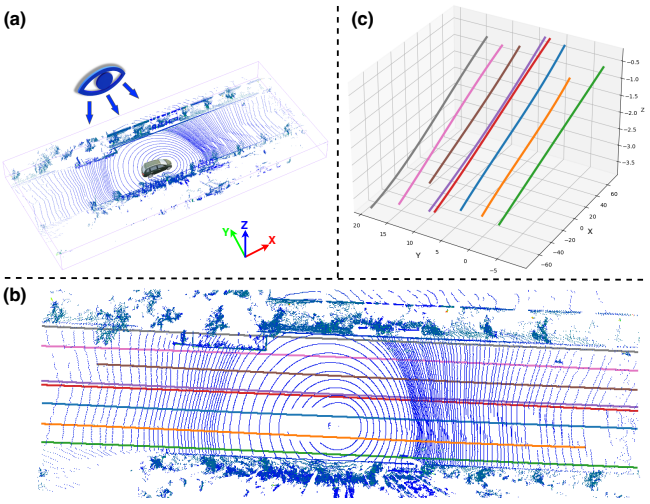

Fig. 1: Our work aims at extracting 3D lane lines from a *surround-view LiDAR point cloud* **(a)**. These detected lanes are visualized in BEV **(b)** and 3D coordinate system **(c)**.

In this paper, we present a ***Li***DAR-based ***S***urround-***V***iew ***3D Lane*** dataset, namely ***LiSV-3DLane***, collecting 20,025 frames of point clouds with 3D lanes manually annotated. Compared to existing datasets, ours contains omnidirectional LiDAR points. Since the manual annotation only captures sparse geometry information of lane lines, we also design an automatic annotation pipeline to generate finer lane labels specifically for dense prediction tasks. This pipeline can be used for a single-frame point cloud by harnessing lane spatial geometries and point cloud attributes (intensity and coplanarity). Lastly, we propose a novel ***Li***DAR-based 3D ***La***ne ***Det***ection framework, dubbed as ***LiLaDet***, which is designed for lane semantic feature and geometry learning. Given a point cloud as input, our model first identifies the lane segments from the projected BEV space to generate 3D lane point proposals *(BEV Pathway)*. Then, to complement the spatial detail loss caused by voxelization, we design a *Spatial Pathway* to refine the lane point proposals with geometric regression and confidence prediction, which accu-

1 School of Computer Science, University of Sydney, {rzha9419, hwan9147}@uni.sydney.edu.au, tom.cai@sydney.edu.au
2 Baidu ACG, {hengyuwen,liushilei, chenjiawen}@baidu.com, yuanda.gao94@gmail.com
3 School of Electronic Information and Electrical Engineering, Shanghai Jiao Tong University, {lio_n1r}@sjtu.edu.cn
‡ Work done during an internship at Baidu ACG

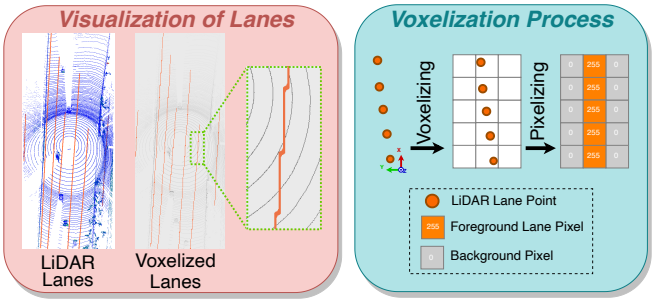

Fig. 2: Current LiDAR-based 3D lane detection methods project LiDAR points into a BEV grid map. The voxelization process leads to the loss of spatial details due to the low resolution of discrete grid cells.

rately restores the 3D positions.

Our main contributions include: **i**) we introduce a LiDAR-based surround-view lane dataset, *LiSV-3DLane* encompassing 20k frames that capture diverse and sophisticated urban and highway scenes; **ii**) we propose an automatic lane annotation pipeline to enrich the acquired manual annotation and generate finer lane annotation, leveraging the inherent lane streamline geometry and intrinsic attributes of spatial points; **iii**) we design a novel LiDAR-based framework, *LiLaDet*, to facilitate the identification of lane markings with point cloud data, integrating both BEV and 3D spatial perspectives to achieve accurate lane identification and localization.

II. RELATED WORKS

3D lane detection was initially developed with camera-based methods among which the pioneering work in the camera-based 3D lane detection domain is 3D-LaneNet [4]. It first encodes hierarchical image features and then projects these features onto a virtual top-view plane by using camera parameters. Gen-LaneNet [6] introduces an expandable two-stage framework, where it first segments lane pixels from the 2D frontal image and then applies convolutional layers to learn the geometric transformation required to restore the 3D lanes. Unlike previous methods, [7]–[9] add strong spatial and structural priors into lane image feature extraction. Anchor3DLane [9] employs frontal image features to directly regress lane anchors that are defined in 3D space. However, the success of monocular 3D lane detection heavily relies on the flat ground assumption to restore 3D information, which prevents them from estimating the depth precisely.

Compared to cameras, LiDAR sensors can offer surround-view 3D perception in varying lighting and weather conditions [11], [12]. However, extracting lane lines from LiDAR point data is challenging due to the sparse nature of the point cloud. To handle this problem, recent works [13], [14] explore the fusion of multiple frames captured at different timestamps and viewpoints, projecting points onto the BEV plane to detect lane marks with segmentation-based methods. However, in practical applications, multi-sensor synchronization is challenging to achieve with appropriate sequential data augmentation due to the high-expense collaborative calibration. [10] proposes a two-stage LiDAR lane detection network, incorporating a row-wise BEV lane feature learning and a local lane correlation refinement. As the first LiDAR-based 3D lane dataset, K-Lane [1] only provides planar lane annotations on a downscaled BEV space and the spatial loss of voxelization is inevitable as shown in Fig. 2. In conclusion, the realm of LiDAR-based 3D lane detection remains underexplored for holistic driving scene understanding in both dataset and model development. To boost the development, we bring in the first large-scale LiDAR-based surround-view 3D lane dataset (*LiSV-3DLane*). An automatic lane annotation pipeline is designed to generate finer lane annotation with richer semantic details. To better learn the 3D lane semantic and geometric features, we develop a novel LiDAR-based 3D lane detection framework *LiLaDet* which is robust to various driving scenarios.

III. LISV-3DLANE DATASET

A. Dataset Introduction

Raw LiDAR Data. *LiSV-3DLane* is a comprehensive LiDAR point cloud dataset that focuses particularly on surround-view 3D lane data. It comprises 20,025 frames from 1,003 unique driving sequences. The dataset contains different daytime periods including morning, afternoon, dusk, and night, and various lighting conditions including sunny, cloudy, and rainy. Besides normal road conditions, it also captures challenging driving scenarios for urban and highway areas such as crowded traffic zones and under-construction roads. These diversified conditions lead to varying degrees of lane occlusions which could enhance the robustness and generalizability in the training of learning-based lane detection models.

Sensor Suite. *LiSV-3DLane* is collected using a Velodyne VLS-128 LiDAR sensor with 128 channels and 0.1∼0.4 degree horizontal angular resolution, and seven cameras with 3840×2160 image resolutions. These sensors are finely calibrated and synchronized to ensure high data quality.

Lane Manual Annotation. The ground-truth lanes are annotated by qualified specialists following the acknowledged lane annotation standard [1], [7], [15]. A lane line in 3D space is manually annotated as a set of points to demarcate drivable zones and is represented by $\{[x_i, y_i, z_i]\}_{i=1}^{N_p}$ where N_p is the number of lane points. Although sparse point-wise annotation is amenable to manual labeling, it lacks density and continuity. Capturing only discrete locations along a lane, as shown in Fig. 3(a), the manually acquired annotation yields low-level geometry information, which constrains contextual understanding of a full driving environment (e.g., how a curve lane functions in relation to the overall traffic flow). To address this limitation, we propose an automatic annotation pipeline that creates detailed lane shapes and considers their inherent point cloud characteristics (intensity and coplanarity). This automatic pipeline can be applied to other LiDAR-based 3D lane datasets to produce dense lane points and provide deeper insights into lane features.

B. Automatic Lane Annotation Pipeline

The current techniques used to annotate lanes in LiDAR point clouds rely on merging multiple sequential frames

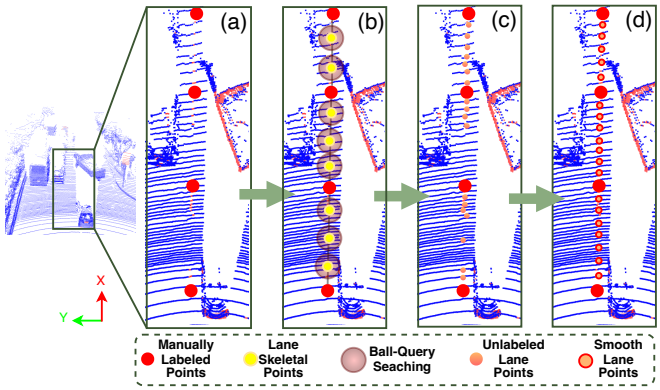

Fig. 3: **Automatic Lane Annotation Pipeline for generating finer lane annotation. (a)** Raw manual lane annotation; **(b)** Lane skeleton representations and lane skeletal points equidistantly sampled along links; **(c)** Unlabeled lane points selected by ball-query searching; **(d)** Smooth lane points sampled from the interpolated cubic curve function.

through Simultaneous Localization And Mapping (SLAM) [13], [14]. Yet, this method demands a high-expense time synchronization approach. Based on our observation, lane points can be identified with their spatial geometries and point cloud attributes. Specifically, we use the RANdom SAmple Consensus (RANSAC) validation step, lane skeletal abstraction, ball-query searching, and cubic curve interpolation to produce finer lane annotation as shown in Fig. 3.

RANSAC Neighboring Ground Plane Fitting. To assess the accuracy and reliability of manually annotated lane points, we introduce a validation step based on the RANSAC algorithm [16]. It can identify the nearest terrain surface plane corresponding to each manually labeled lane point. After calculating each point's perpendicular distance to the plane, a pre-defined distance threshold of 0.01 meters is adopted to determine if this lane point can be accepted. To bolster ground truth quality, erroneous lane points are projected onto the identified plane for height recalibration. The RANSAC algorithm is implemented by using the open-source machine-learning library `scikit-learn`.

Lane Skeletonization. Inspired by recent works [17], [18] modeling driving scene components as connected polylines, we characterize lane geometric shape by linking lane points in the positive x direction and sampling skeletal points equidistantly along each link as shown in Fig. 3(b).

Ball-Query Lane Points Searching. Assuming the lane skeletal point as a reference centroid, we employ ball query searching to identify local unlabeled lane points as shown in Fig. 3(b) and (c). Attributed to the reflectivity of lane paint material, LiDAR lane points typically exhibit higher intensity than ground points. However, other points on curbs or shrubs also have relatively high intensity and can be misclassified as lane points. To mitigate this confusion, we incorporate coplanarity as a criterion to filter incorrect lane points.

Cubic Curve Interpolation. Lastly, the smooth lane points are sampled from an interpolated cubic curve function as shown in Fig. 3(d), which offers a complete lane shape with richer geometric details.

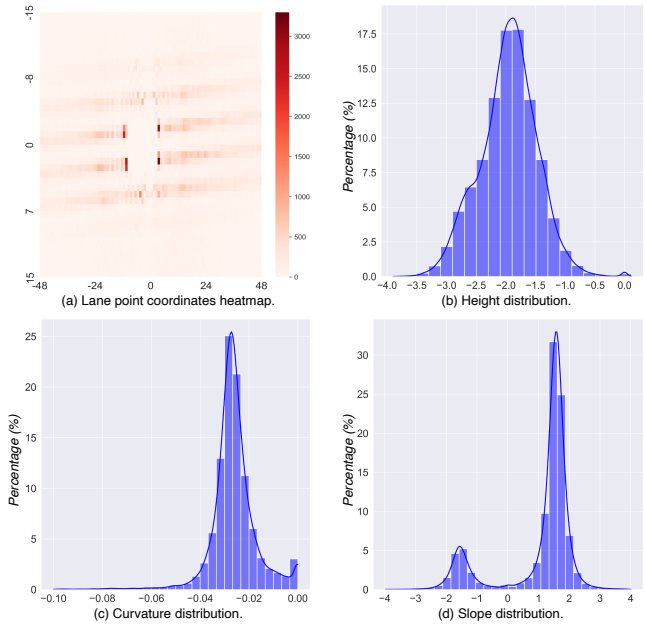

Fig. 4: **Dataset Statistics Analysis.** We analyze the coordinates, height, curvature, and slope of lanes to illustrate the diversity of lane geometry in *LiSV-3DLane*.

C. Dataset Analysis

In our *LiSV-3DLane* dataset, lane points are annotated within the LiDAR coordinate system, so we focus on the analysis of their 3D positional attributes. Fig. 4(a) visualizes 2D coordinates of all lane points using a heatmap. The horizontal axis is the x coordinate, the vertical axis is the y coordinate, and the frequency of lane point occurrences is indicated by color shade. From the visualization, it is apparent that the labeled lanes surround the ego vehicle, facilitating surround-view contextual learning for model development. Different from the existing LiDAR-based lane dataset [1], we provide lane labels with height values, enabling the model to predict realistic 3D lane coordinates. Since the LiDAR sensor is placed at the top of the ego vehicle, the height values of lanes are all negative and their distribution is illustrated in Fig. 4(b). A set of 3D lane points is often interpolated as a cubic curve function to model their geometric shape [6]. Taking x-y plane as an example, the curve function is expressed as $y = ax^3 + bx^2 + cx + d$ where the primary controllable parameter a significantly influences lane curvature. Its value distribution is illustrated in Fig. 4(c). It is worth noting that lanes exhibit diverse curvatures and they do not fit a uniform geometric shape. This variability hinders the lane geometry understanding. The features of lanes, including their coordinates throughout the whole scene and their varied shapes, can explain that the anchor-based method using 3D bounding boxes as in [19] or line-like structures as in [9] is not suitable for lane detection. Extensive pre-defined anchors of various shapes are required to cover each possible position, which is impractical in model convergence and operation time. Camera-based 3D lane detection adopts a flat ground assumption to collerate 2D frontal image to BEV space. As Fig. 4(d) indicates, lanes in real-world scenarios

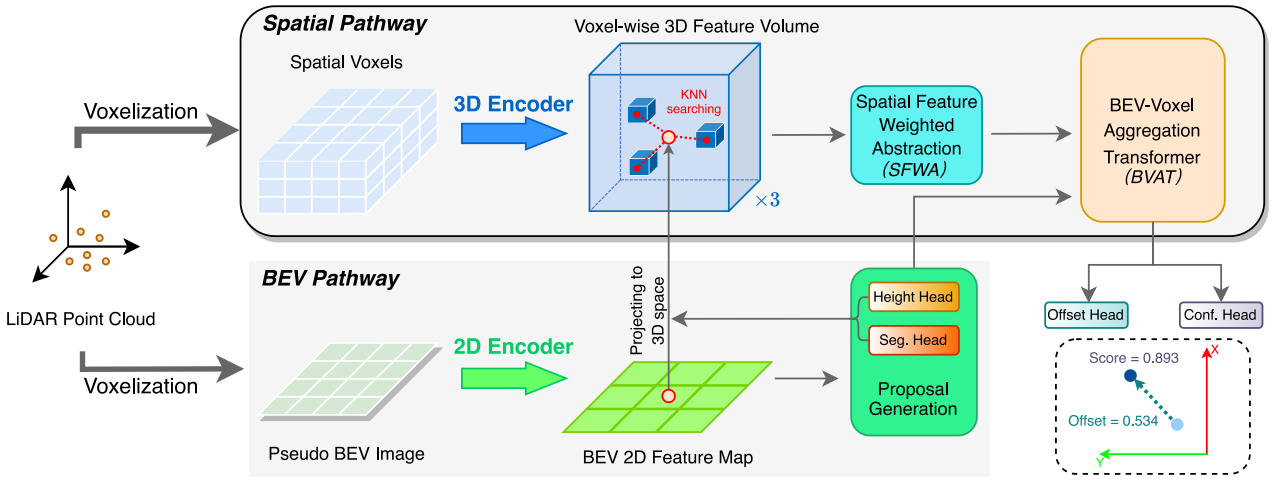

Fig. 5: **Overview of our proposed *LiLaDet* framework.** Given a LiDAR point cloud as input, our model first identifies the lane segments from the projected BEV space to generate 3D lane point proposals at the *BEV Pathway*. Then, we design a *Spatial Pathway* to refine the lane proposal points through geometric regression and confidence prediction.

rarely conform to a perfectly horizontal alignment, thereby underscoring the limitations of camera-based models for capturing reliable spatial geometry.

IV. LILADET FRAMEWORK

In this study, we introduce *LiLaDet*, a novel LiDAR-based 3D lane detection framework. As schematically illustrated in Fig. 5, the framework consists of two pathways: the BEV and the spatial pathways, which operate on two voxelized representations of the LiDAR point cloud input: pseudo BEV grid image and spatial voxels, respectively. In the *BEV Pathway*, we train a 2D encoder to extract BEV feature maps, leveraging the global receptive field of the attention mechanism to capture spatial dependencies across the scene. These feature maps subsequently yield a segmentation map and height prediction for lane proposal generation. In the *Spatial Pathway*, a stack of 3D sparse convolutions with multiple scales encodes the input into hierarchical spatial features. Then, at the three deepest scales, features are separately searched using the k Nearest Neighbour (k-NN) algorithm to generate local spatial features around lane proposal points [20]. We introduce a Spatial Feature Weighted Abstraction *(SFWA)* module for weighted aggregation of multi-scale local spatial features. Lastly, the BEV and spatial features are merged in a BEV-Voxel Aggregation Transformer *(BVAT)* module which employs a cross-attention mechanism to decode features from different modalities when making final lane predictions. The details of our proposed framework are explained below.

A. 2D BEV Pathway

In BEV segmentation methods [1], [2], [21], [22], traffic entities of arbitrary shapes are identified using a set of foreground pixels in the BEV space. Such a pixel-wise representation provides adaptability to the inherent lane geometry variances, which prevents lane detection from the human-crafted spatial and structural priors or anchors. Inspired by this, we formulate lane detection as a semantic segmentation problem. Concretely, we first generate a pseudo BEV grid image by pillarizing raw LiDAR points [23]. Analogous to 2D image processing, we employ a Vision Transformer (ViT) [24], [25] with a multi-scale receptive field to learn discriminative BEV feature maps $F_{bev} \in \mathbb{R}^{L_{bev} \times W_{bev} \times C_{bev}}$ where L_{bev} and W_{bev} denote the size of the BEV feature map and C_{bev} denotes the feature channel dimension.

Lane Point Proposal Generation. We forward BEV feature maps F_{bev} into a 2D convolutional segmentation head to partition foreground lane pixels from the grid scene. To obtain realistic 3D coordinates of lane lines, we additionally append a shared Multi-Layer Perceptron (MLP) head to predict the height of each positive lane pixel with residual learning. Given xy coordinates in LiDAR space calculated from the segmentation map and z coordinate of the height map, we can project the lane pixels of the 2D BEV plane back to 3D space, generating 3D lane proposal points.

B. 3D Spatial Pathway

Semantic segmentation can effectively capture lane instances from LiDAR points, but the generated lane proposal points in the *BEV Pathway* are not accurately localized in 3D space. Since the original BEV grid image is encoded to obtain the low-resolution BEV feature maps, such coarse BEV maps cannot provide sufficient spatial details to restore accurate lane localization in the input scene [20], [26]. To mitigate this problem, we design the 3D *Spatial Pathway* to complement the *BEV Pathway* by directly using LiDAR point cloud. This explicit 3D data representation provides vital spatial geometric cues for further lane proposal refinement. We employ 3D voxel CNN with sparse convolutions [20], [27] to efficiently encode a LiDAR point cloud into hierarchical spatial feature volumes. The raw LiDAR point cloud is first discretized into a set of 3D volumes. The 3D encoder stacks $3 \times 3 \times 3$ sparse convolutional layers to gradually process voxelized point cloud into 3D spatial feature volumes $F_{sp}^{l_n}$ at different scale levels l_n where n is the scale level order (i.e., n = 1, 2, 3, 4).

Spatial Feature Weighted Abstraction *(SFWA)*. As the receptive field enlarges as the 3D voxel CNN network goes deeper, voxel-wise spatial features at different scale levels exert different influences on spatial contextual understanding. In open wild scenarios, lane detection is sensitive to fine-scale voxel-wise spatial features, such as the adjacent road surface features. Conversely, in complex traffic situations, spatial features at the coarse scale can better capture long-range spatial relationships with large objects such as vehicles or other traffic infrastructures.

To capture the local voxel-wise spatial features surrounding each lane proposal point, a set abstraction module like [20], [28] is employed at each scale level. We utilize the *k*-NN algorithm to retrieve *k*-nearest voxel-wise features and concatenate them as $\{\mathcal{F}^{l_n} \in \mathbb{R}^{k \times C_v} : [f_1^{l_n}; \cdots; f_k^{l_n}]\}$ where C_v is the channel dimension of the voxel-wise spatial feature. Considering transform invariance in point cloud processing [28], we use the max-pooling operation to generate a local abstract feature vector, denoted as $\{S^{l_n} \in \mathbb{R}^{1 \times C_v} : \mathcal{P}(\mathcal{F}^{l_n})\}$ where $\mathcal{P}$ is the max-pooling operation. In our case, we only consider the spatial features at the l_2, l_3, and l_4 scale levels. This abstract module is performed at each scale level, then all abstract features are concatenated as $\{S \in \mathbb{R}^{3 \times C_v} : [S^{l_2}; S^{l_3}; S^{l_4}]\}$. Afterwards, three attention weights $W \in \mathbb{R}^3$ are predicted, emphasizing the scale level contributing the most in lane spatial positional learning as follows: $W = \sigma(\mathcal{M}(S))$ where $\mathcal{M}$ stands for a shared MLP layer and σ denotes the softmax function, then these weights are separately multiplicated with the corresponding voxel-wise spatial feature $\mathcal{F}^{l_n}$. The weighted features from all scale levels are lastly concatenated and processed by another shared MLP to generate final spatial features for a lane proposal point: $\{F_{sp} \in \mathcal{R}^{1 \times C_{sp}} : \mathcal{M}(\mathcal{R}([\hat{\mathcal{F}^{l_2}}; \hat{\mathcal{F}^{l_3}}; \hat{\mathcal{F}^{l_4}}]))\}$ where $\mathcal{R}$ denotes the reshape operation and C_{sp} is the spatial feature channel dimension.

BEV-Voxel Aggregation Transformer *(BVAT)*. The Cross-Attention (CA) mechanism is employed to fuse and calculate the correlation between multiple feature resources in vision tasks [24], [29]–[31], which can enrich a query $\mathbf{Q}$ vector with complementary information provided by a pair of key $\mathbf{K}$ and value $\mathbf{V}$ vectors. Specifically, this process can be defined as the following functions:

$$\bar{\mathbf{Q}} = \mathbf{Q}\mathbf{W_q}, \bar{\mathbf{K}} = \mathbf{K}\mathbf{W_k}, \bar{\mathbf{V}} = \mathbf{V}\mathbf{W_v} \tag{1}$$

$$\mathrm{CA}(\bar{\mathbf{Q}}, \bar{\mathbf{K}}, \bar{\mathbf{V}}) = softmax(\bar{\mathbf{Q}}\bar{\mathbf{K}}^T/\sqrt{D_h})\bar{\mathbf{V}} \tag{2}$$

where $\mathbf{W}_{\{\mathbf{q},\mathbf{k},\mathbf{v}\}}$ is linear transformations and D_h denotes the hidden feature embedding dimension. In our case, CA is applied across BEV grid image and spatial voxels. Suppose N lane proposal points are generated from the *BEV Pathway*, we have 2D BEV feature and 3D spatial feature with the size being $N \times C_{bev}$ and $N \times C_{sp}$ respectively. *BVAT* module aims at fusing these two features from different data modalities. Concretely, we generate the module inputs as:

$$\mathbf{Q}_{bev}, \mathbf{K}_{bev}, \mathbf{V}_{bev} = \mathrm{LN}(F_{bev}), \mathrm{LN}(F_{bev}), \mathrm{LN}(F_{bev}) \tag{3}$$

$$\mathbf{Q}_{sp}, \mathbf{K}_{sp}, \mathbf{V}_{sp} = \mathrm{LN}(F_{sp}), \mathrm{LN}(F_{sp}), \mathrm{LN}(F_{sp}) \tag{4}$$

where LN denotes Layer Normalization. The CA across two modalities is manipulated as:

$$\begin{aligned}\mathbf{Z} =& \mathrm{FFN}(\mathrm{CA}(\mathbf{Q}_{bev}, \mathbf{K}_{sp}, \mathbf{V}_{sp})) \\ &+ \mathrm{FFN}(\mathrm{CA}(\mathbf{Q}_{sp}, \mathbf{K}_{bev}, \mathbf{V}_{bev}))\end{aligned} \tag{5}$$

where FFN denotes Feed Forward Network and $\mathbf{Z}$ denotes output feature. Then, the MLP-based offset head and confidence head are attached to predict the *xy* coordinate offset and the confidence score of each lane point, respectively.

C. Learning Objectives

We use Binary Cross Entropy (BCE) loss to supervise the training of the segmentation head to partition pixel-wise lanes [1], [26]. For the geometric regression training, we use smooth L1-Norm loss [32] to supervise height prediction in the *BEV Pathway* and *xy* coordinate offset prediction in the *Spatial Pathway* [20], [23]. Notably, lane BEV segmentation output may yield false positive pixels (i.e. turning road markings) that have similar intensity as lane markings. To counteract this problem, a confidence head is introduced to geometrically select high-fidelity lane points based on predicted scores. We compute the Euclidean Distance between a lane proposal point and its nearest ground-truth point. If the distance is within a distance threshold τ, this point is assigned as true, otherwise, it is false. The confidence prediction head is trained using BCE loss.

V. EXPERIMENTS AND RESULTS

A. Datasets and Metrics

We split our *LiSV-3DLane* dataset into 12,000/3,982/4,043 frames for training/validation/testing sets. We also conduct experiments on K-Lane [1] to evaluate lane detection performance. We split a total of 15,382 frames into 7,687/3,848/3,847 frames for training/validation/testing sets, but the dataset only provides lane labels on the BEV plane without realistic height values. For effective 3D lane detection evaluation, given a labeled lane point in K-Lane, we assume the minimum height value of its neighboring points as the true height value. We employ standard 3D lane detection measures [6] using the bipartite matching method to match the predicted and ground-truth lanes for calculating precision, recall, and F1-score metrics. To calculate spatial similarity, we also use unilateral Chamfer Distance (CD), a common distance metric in point cloud processing tasks.

B. Implementation Details

The BEV and spatial pathways are trained separately using the AdamW optimizer with a learning rate of $2e^{-4}$. We use a small *xy* resolution of 0.04 m to form a pseudo BEV input image at 2400 × 1000 resolution [1], [19], and 0.32 m to form a downscaled lane ground-truth image. The *BEV Pathway* is trained with a batch size of 4 for 24 epochs. When testing, we use a density-based spatial clustering method to classify lane instances. In the *Spatial Pathway*, we voxelize the point cloud with a voxel size of [0.1, 0.1, 0.2], a maximum number of points per voxel of 32, and a maximum number of 12000. Given the lane point proposals,

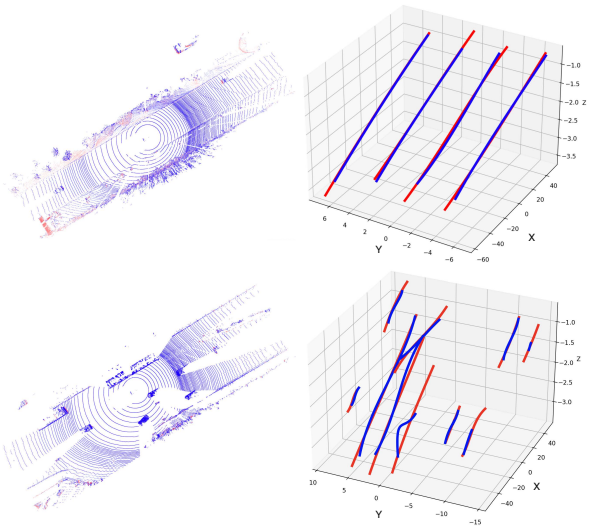

Fig. 6: Qualitative Evaluation of *LiLaDet* on *LiSV-3DLane*. **Left**: LiDAR point cloud inputs; **Right**: The predicted and ground-truth lanes are shown in blue and red, respectively.

we search for k=12 neighboring voxel features to capture local spatial information. The distance threshold τ is 0.5 meters. The *Spatial Pathway* is trained with a batch size of 20 for 6 epochs. All experiments are conducted on NVIDIA GeForce RTX 3090s and PyTorch 1.11.0.

C. Results and Analysis

To investigate the applicability of our proposed framework, we evaluate lane detection models under frontal and omnidirectional views and report their performances in Tables I and II. However, camera-based 3D lane detection methods have a limited field of view and strict planar assumptions of bridging 2D and 3D space, which suffer in realistic application. Turning to LiDAR-based methods, they are extrinsic-free, and directly process real-world data, but the existing LiDAR-based methods only identify lanes in BEV space without height predictions. To make a comprehensive performance analysis in spatial measures, we re-implement LLDN-GFC by adding a height prediction head as indicated by †. The comparisons show our proposed method outperforms other existing LiDAR-based methods under frontal and omnidirectional views. We also examine our model performance on the K-Lane dataset in Table III. Compared with other LiDAR-based lane detection methods, with the fusion of BEV and 3D spatial information, our model can effectively detect and localize the lanes in the omnidirectional scene.

In the qualitative evaluation in Fig. 6, our model has robust detection performance for urban and highway scenarios. However, in complex driving scenarios containing more traffic components, the insufficient number of point clouds allocated to lane lines fails to fully reveal the necessary spatial geometric features, leading to suboptimal solutions.

D. Ablation Studies

We validate the effectiveness of individual components of our proposed model and conduct the ablation experiments on the *LiSV-3DLane* test set. The results are shown in Table IV where **BP** stands for the *BEV Pathway* and **SP** stands for *Spatial Pathway*. In Case (1), two pathway features are aggregated by element-wise addition and then forwarded to the prediction heads. The lane detection performance of our model is gradually improved by adding **SFWA** and **BVAT**. The introduction of exploring spatial features in the LiDAR point cloud brings 1.83% and 26.5% improvements in detection performance and spatial similarity, respectively, which proves the benefits of utilizing LiDAR point information for 3D lane detection.

TABLE I: Experimental results on our *LiSV-3DLane* test set for Frontal View.

Method	*Modality*	*Precision*(%) ↑	*Recall*(%) ↑	*F1*(%) ↑	CD_{3D}(m) ↓	CD_{BEV}(m) ↓
3D-LaneNet [4]	Image	51.41	22.82	31.61	0.940	0.940
Gen-LaneNet [6]	Image	54.15	19.36	28.53	0.260	0.251
Anchor3DLane [9]	Image	57.23	33.40	42.18	0.312	0.301
LLDN-GFC [1]	LiDAR	65.79	79.92	72.17	-	0.232
†LLDN-GFC [1]	LiDAR	**65.81**	79.94	72.19	0.235	0.232
RLLDN-LC [10]	LiDAR	62.20	78.64	69.46	-	0.203
LiLaDet (ours)	LiDAR	65.78	**85.16**	**74.23**	**0.158**	**0.158**

TABLE II: Experimental results on our *LiSV-3DLane* test set for Surround View.

Method	*Modality*	*Precision*(%) ↑	*Recall*(%) ↑	*F1*(%) ↑	CD_{3D}(m) ↓	CD_{BEV}(m) ↓
LLDN-GFC [1]	LiDAR	61.25	78.18	68.69	-	0.198
†LLDN-GFC [1]	LiDAR	61.28	78.25	68.73	0.195	0.192
RLLDN-LC [10]	LiDAR	58.44	73.39	65.07	-	0.179
LiLaDet (ours)	LiDAR	**63.68**	**83.76**	**72.35**	**0.150**	**0.147**

TABLE III: Experimental results on K-Lane test set.

Method	*Modality*	*Precision*(%) ↑	*Recall*(%) ↑	*F1*(%) ↑	CD_{3D}(m) ↓	CD_{BEV}(m) ↓
3D-LaneNet [4]	Image	78.28	27.81	41.04	0.674	0.637
Gen-LaneNet [6]	Image	**82.37**	29.55	43.49	0.302	0.294
LLDN-GFC [1]	LiDAR	70.37	85.59	77.24	-	0.230
†LLDN-GFC [1]	LiDAR	70.37	85.62	77.25	0.223	0.221
RLLDN-LC [10]	LiDAR	71.73	85.84	78.15	-	0.198
LiLaDet (ours)	LiDAR	72.82	**87.32**	**79.41**	**0.173**	**0.172**

TABLE IV: Ablation study of *LiLaDet* on our *LiSV-3DLane* dataset.

Case	**BP** + **SP**	**SFWA**	**BVAT**	*F1*(%) ↑	$CD_{3D}(m)$ ↓
(1)	✓			70.52	0.204
(2)	✓	✓		71.76	0.187
(3)	✓	✓	✓	72.35	0.150

VI. CONCLUSION

In this paper, we present a LiDAR-based surround-view 3D Lane detection dataset, *LiSV-3DLane*. To handle the sparsity of manual lane annotation, we introduce an automated lane annotation pipeline to improve the labeling quality for dense prediction tasks. Subsequently, we propose a novel LiDAR-based 3D lane detection model, *LiLaDet*, utilizing the spatial structural information of the LiDAR points for extracting lane markings in the point cloud scan. Extensive experiments and ablation studies prove the effectiveness of our model. For future work, we will exploit the multi-modality fusion technique of incorporating more semantic cues into the proposed LiDAR-based framework, to reduce the computational requirements of processing 3D point clouds.

REFERENCES

[1] D.-H. Paek, S.-H. Kong, and K. T. Wijaya, "K-lane: Lidar lane dataset and benchmark for urban roads and highways," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPRW)*, 2022, pp. 4450–4459.

[2] D. Bai, T. Cao, J. Guo, and B. Liu, "How to build a curb dataset with lidar data for autonomous driving," in *2022 International Conference on Robotics and Automation (ICRA)*. IEEE, 2022, pp. 2576–2582.

[3] J. Kini, A. Mian, and M. Shah, "3dmodt: Attention-guided affinities for joint detection & tracking in 3d point clouds," in *2023 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2023, pp. 841–848.

[4] N. Garnett, R. Cohen, T. Pe'er, R. Lahav, and D. Levi, "3d-lanenet: End-to-end 3d multiple lane detection," in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2019, pp. 2921–2930.

[5] N. Efrat, M. Bluvstein, S. Oron, D. Levi, N. Garnett, and B. E. Shlomo, "3d-lanenet+: Anchor free lane detection using a semi-local representation," *arXiv preprint arXiv:2011.01535*, 2020.

[6] Y. Guo, G. Chen, P. Zhao, W. Zhang, J. Miao, J. Wang, and T. E. Choe, "Gen-lanenet: A generalized and scalable approach for 3d lane detection," in *European Conference on Computer Vision (ECCV)*, 2020, pp. 666–681.

[7] L. Chen, C. Sima, Y. Li, Z. Zheng, J. Xu, X. Geng, H. Li, C. He, J. Shi, Y. Qiao *et al.*, "Persformer: 3d lane detection via perspective transformer and the openlane benchmark," in *European Conference on Computer Vision (ECCV)*. Springer, 2022, pp. 550–567.

[8] R. Wang, J. Qin, K. Li, Y. Li, D. Cao, and J. Xu, "Bev-lanedet: An efficient 3d lane detection based on virtual camera via key-points," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023, pp. 1002–1011.

[9] S. Huang, Z. Shen, Z. Huang, Z.-h. Ding, J. Dai, J. Han, N. Wang, and S. Liu, "Anchor3dlane: Learning to regress 3d anchors for monocular 3d lane detection," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023, pp. 17 451–17 460.

[10] D.-H. Paek, K. T. Wijaya, and S.-H. Kong, "Row-wise lidar lane detection network with lane correlation refinement," in *2022 IEEE 25th International Conference on Intelligent Transportation Systems (ITSC)*. IEEE, 2022, pp. 4328–4334.

[11] M. Hahner, C. Sakaridis, D. Dai, and L. Van Gool, "Fog simulation on real lidar point clouds for 3d object detection in adverse weather," in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2021, pp. 15 283–15 292.

[12] M. Hahner, C. Sakaridis, M. Bijelic, F. Heide, F. Yu, D. Dai, and L. Van Gool, "Lidar snowfall simulation for robust 3d object detection," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022, pp. 16 364–16 374.

[13] R. Liu, Z. Guan, Z. Yuan, A. Liu, T. Zhou, T. Kun, E. Li, C. Zheng, and S. Mei, "Learning to detect 3d lanes by shape matching and embedding," in *2023 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, 2023, pp. 4280–4288.

[14] Z. Guan, R. Liu, Z. Yuan, A. Liu, K. Tang, T. Zhou, E. Li, C. Zheng, and S. Mei, "Flexible 3d lane detection by hierarchical shape matching," in *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, vol. 37, no. 1, 2023, pp. 694–701.

[15] F. Yan, M. Nie, X. Cai, J. Han, H. Xu, Z. Yang, C. Ye, Y. Fu, M. B. Mi, and L. Zhang, "Once-3dlanes: Building monocular 3d lane detection," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022, pp. 17 143–17 152.

[16] M. A. Fischler and R. C. Bolles, "Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography," *Communications of the ACM*, vol. 24, no. 6, pp. 381–395, 1981.

[17] 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," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020, pp. 11 525–11 533.

[18] 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.

[19] A. H. Lang, S. Vora, H. Caesar, L. Zhou, J. Yang, and O. Beijbom, "Pointpillars: Fast encoders for object detection from point clouds," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (CVPR)*, 2019, pp. 12 697–12 705.

[20] S. Shi, C. Guo, L. Jiang, Z. Wang, J. Shi, X. Wang, and H. Li, "Pv-rcnn: Point-voxel feature set abstraction for 3d object detection," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (CVPR)*, 2020, pp. 10 529–10 538.

[21] 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 *2023 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2023, pp. 2774–2781.

[22] L. Peng, Z. Chen, Z. Fu, P. Liang, and E. Cheng, "Bevsegformer: Bird's eye view semantic segmentation from arbitrary camera rigs," in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, 2023, pp. 5935–5943.

[23] B. Yang, W. Luo, and R. Urtasun, "Pixor: Real-time 3d object detection from point clouds," in *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition (CVPR)*, 2018, pp. 7652–7660.

[24] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly *et al.*, "An image is worth 16x16 words: Transformers for image recognition at scale," *arXiv preprint arXiv:2010.11929*, 2020.

[25] H. Fan, B. Xiong, K. Mangalam, Y. Li, Z. Yan, J. Malik, and C. Feichtenhofer, "Multiscale vision transformers," in *Proceedings of the IEEE/CVF international conference on computer vision (CVPR)*, 2021, pp. 6824–6835.

[26] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2015: 18th International Conference, Munich, Germany, October 5-9, 2015, Proceedings, Part III 18*. Springer, 2015, pp. 234–241.

[27] Y. Yan, Y. Mao, and B. Li, "Second: Sparsely embedded convolutional detection," *Sensors*, vol. 18, no. 10, p. 3337, 2018.

[28] C. R. Qi, H. Su, K. Mo, and L. J. Guibas, "Pointnet: Deep learning on point sets for 3d classification and segmentation," in *Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR)*, 2017, pp. 652–660.

[29] Y. Liu, J. Yan, F. Jia, S. Li, A. Gao, T. Wang, X. Zhang, and J. Sun, "Petrv2: A unified framework for 3d perception from multi-camera images," *arXiv preprint arXiv:2206.01256*, 2022.

[30] Z. Li, W. Wang, H. Li, E. Xie, C. Sima, T. Lu, Y. Qiao, and J. Dai, "Bevformer: Learning birds-eye-view representation from multi-camera images via spatiotemporal transformers," in *European conference on computer vision (ECCV)*. Springer, 2022, pp. 1–18.

[31] Y. Li, A. W. Yu, T. Meng, B. Caine, J. Ngiam, D. Peng, J. Shen, Y. Lu, D. Zhou, Q. V. Le *et al.*, "Deepfusion: Lidar-camera deep fusion for multi-modal 3d object detection," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022, pp. 17 182–17 191.

[32] R. Girshick, "Fast r-cnn," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015, pp. 1440–1448.