

Pandar128 dataset for lane line detection^{*}

Filip Beránek¹[0009–0002–3563–3097], Václav Diviš¹[0000–0001–9935–7824], and Ivan Gruber¹[0000–0003–2333–433X]

Department of Cybernetics and New Technologies for the Information Society,
Technická 8, 301 00 Plzeň, Czech Republic bionothi@ntis.zcu.cz

Abstract. We present Pandar128, the largest public dataset for lane line detection using a 128-beam LiDAR. It contains over 52,000 camera frames and 34,000 LiDAR scans, captured in diverse real-world conditions in Germany. The dataset includes full sensor calibration (intrinsics, extrinsics) and synchronized odometry, supporting tasks such as projection, fusion, and temporal modeling.

To complement the dataset, we also introduce SimpleLidarLane, a light-weight baseline method for lane line reconstruction that combines BEV segmentation, clustering, and polyline fitting. Despite its simplicity, our method achieves strong performance under challenging various conditions (e.g., rain, sparse returns), showing that modular pipelines paired with high-quality data and principled evaluation can compete with more complex approaches.

Furthermore, to address the lack of standardized evaluation, we propose a novel polyline-based metric — Interpolation-Aware Matching F1 (IAM-F1) — that employs interpolation-aware lateral matching in BEV space. All data and code are publicly released to support reproducibility in LiDAR-based lane detection.

Keywords: Lane line detection · Lane detection · LiDAR · Point Cloud

1 Introduction

In pursuit of safer and more comfortable roadways, automotive manufacturers are developing the Advanced Driver Assistance Systems (ADAS) [29]. These systems, ranging from Adaptive Cruise Control (ACC) to Lane-Keeping Assist (LKA), represent a critical step towards fully autonomous vehicles [10].

LKA is one of the critical systems for autonomous vehicles which is required by law in the European Union (EU) to be present in all new cars sold in EU [5] starting from July 2024. The task of LKA is to detect accurate position and curvature of the ego lane and adjacent lanes, which is provided as an input for path planning of the car. Given its relevance to functional safety, the system must operate reliably across all conditions, including day and night, heavy rain, fog, and dense traffic.

^{*} The work has been supported by the grant of the University of West Bohemia, project No. SGS-2025-011. Computational resources were provided by the e-INFRA CZ project (ID:90254), supported by the Ministry of Education, Youth and Sports of the Czech Republic. https://github.com/filipberanek/lane_lines_dataset

The early methods for lane detection relied on classical image processing techniques applied to camera data, such as edge detection followed by line fitting [6,8,19]. Later on, approaches based on deep neural networks were developed, which improved the accuracy and robustness [14,18,23]. Unfortunately, the number of algorithms is significantly exceeding the amount of evaluating datasets. Up until now, there were only 3 datasets for benchmarking of lane detection. Camera-based datasets CULane [18] and TuSimple [1] and Light Detection and Ranging (LiDAR) based dataset K-lane [16].

Despite these advances, camera-based approaches still face several limitations that hinder their robustness in real-world applications. First, their performance is highly sensitive to lighting conditions—low light, glare, strong shadows, or nighttime scenarios can significantly reduce detection accuracy [16,18]. Second, projecting detected lanes from the camera image plane into the vehicle’s 3D coordinate system can introduce geometric distortions, particularly on curved or sloped roads [3]. These complications can be addressed by using LiDAR sensors, which inherently capture three-dimensional spatial information and exhibit robustness to challenging lighting conditions, including low light and glare [17,16,7].

To support the community in developing and testing LKA applications, we publish a new LiDAR-based dataset and hence contribute to the overall LKA development process as follows:

- We present Pandar128, the first lane detection dataset collected with a 128-beam LiDAR.
- Pandar128 is the largest dataset for lane line detection, containing 52200 images and 34829 LiDAR frames.
- For the first time, we include vehicle odometry together with full camera and LiDAR calibration parameters (intrinsics, extrinsics, and distortion), enabling accurate projection and sensor fusion.
- Furthermore, we present a baseline method for LiDAR detection combining deep learning and clustering.
- We introduce a novel evaluation metric for polyline-based lane detection that computes average point-to-line distance with tolerance-aware matching.

2 Related Work

2.1 Datasets

A comprehensive overview of ADAS-related datasets is provided in [21]. However, only a small subset of those datasets include LiDAR point clouds, and even fewer provide annotations for lane markings—an essential component for autonomous driving research. To the best of our knowledge, only three publicly available datasets offer labeled lane markings in LiDAR point clouds: Kaist-Lane (K-Lane) [16], LiSV-3DLane [27], and Waymo Open Dataset (Waymo) [22], the latter updated with semantic annotations in March 2024.

- **Waymo** [22] provides 3D semantic-segmentation labels for each LiDAR frame. The dataset includes a "Lane Marker" class among 23 semantic labels, covering painted road stripes such as solid and dashed dividing lines. Waymo collected 230k LiDAR frames labeled for semantic classes, under different weather conditions.

- **K-Lane**, introduced by Bai et al. [16], is a LiDAR-based end-to-end framework that predicts lane boundaries directly in the bird’s-eye view (BEV) space. Alongside the method, the authors also released a dataset containing over 15,000 annotated frames with up to six labeled lanes per scene. The data covers a wide range of road and traffic conditions, including varying occlusion levels, both daytime and nighttime driving, curved road segments, and complex topologies such as merging and diverging lanes.
- **LiSV-3DLane**, developed by Zhao et al. [26], is a large-scale 3D lane dataset comprising 20,000 surround-view LiDAR frames with detailed semantic annotations. Unlike prior datasets restricted to a frontal field of view, LiSV-3DLane offers full 360-degree coverage around the ego vehicle, allowing the capture of complex lane geometries in both urban and highway environments.

2.2 Lane Detection

In recent years, LiDAR-based lane detection has emerged as a robust alternative to camera-based methods, particularly under adverse weather and lighting conditions [17,30,25]. Several approaches have been proposed to extract lane markings directly from 3D point clouds using deep learning or hybrid techniques. These methods differ significantly in terms of input representation (raw point cloud vs. bird’s-eye view projection), network architecture, and output format (segmentation maps, polylines, or continuous surfaces).

Bai et al. [16] proposed an end-to-end LiDAR-based system called K-Lane, which directly predicts lane boundaries in bird’s-eye view using a deep neural network. Their method reports an F1-score of 82.1% on the K-Lane dataset, combining lane localization accuracy with classification quality and serves as an important benchmark in the field. The authors also introduce an instance-aware detection strategy, allowing their model to detect multiple lanes in complex urban scenarios.

Zhao et al. [26] introduced LiLaDet, a dual-path network that jointly performs BEV segmentation and 3D spatial regression. This architecture enables accurate localization of lane lines in world coordinates and outperforms previous state-of-the-art methods. Their approach was extensively evaluated on the LiSV-3DLane dataset, which provides a challenging and diverse benchmark for full-surround LiDAR-based lane detection.

Other recent works, such as [26,4], further explore the use of ray-based matching and multi-task learning to improve detection robustness and label efficiency. A broader review of LiDAR-based road structure extraction techniques is provided in [15], which positions lane detection as a critical application within the autonomous driving stack.

3 Data

While the majority of existing lane detection datasets rely on camera images, our work introduces the first large-scale dataset based on high-resolution LiDAR. To the best of our knowledge, it is not only the largest publicly available dataset of its kind but also the first to include comprehensive vehicle odometry, sensor intrinsics, and extrinsics parameters. These additional calibration parameters and odometry data enable a wide range of advanced research directions beyond single-frame perception, such as multi-sensor fusion, temporal accumulation of point clouds, or map-level lane modeling. In this section, we describe the data collection setup, annotation process, and provide a detailed overview of the dataset’s structure and statistics.

3.1 Sensor Setup

Our sensor configuration was inspired by the Woodscape dataset [24], which served as a reference platform for multi-modal data acquisition.

A high-resolution camera (2896×1876 pixels) is mounted behind the windshield, operating at 30 frames per second (FPS) or 15 FPS depending on environment. In rainy conditions, lower frame rate (15 FPS) was used to ensure higher image quality and reduce motion blur. The camera provides a 120° horizontal and 73° vertical field of view. All intrinsic and extrinsic calibration parameters, including distortion coefficients, are included in the dataset.

The LiDAR sensor is a Pandar128 LiDAR unit, mounted on the roof. This LiDAR provides full 360° coverage with a vertical FOV of 40°, operating at 20 FPS. The sensor has a ranging capability of up to 200 meters at 10% reflectivity. Compared to other widely used datasets such as KITTI-360 [12] and K-Lane [16], which rely on 64-beam LiDARs, the use of 128 beams allows for denser and more detailed scene reconstruction.

The final component of the sensor suite is a high-precision GPS/IMU system providing 20 Hz odometry. Instead of absolute coordinates, we store relative transformation matrices between consecutive frames, enabling motion compensation, temporal fusion, and map-level reconstruction. Although raw GPS data is omitted, the odometry ensures sufficient spatial consistency for multi-frame alignment and time-aware modeling.

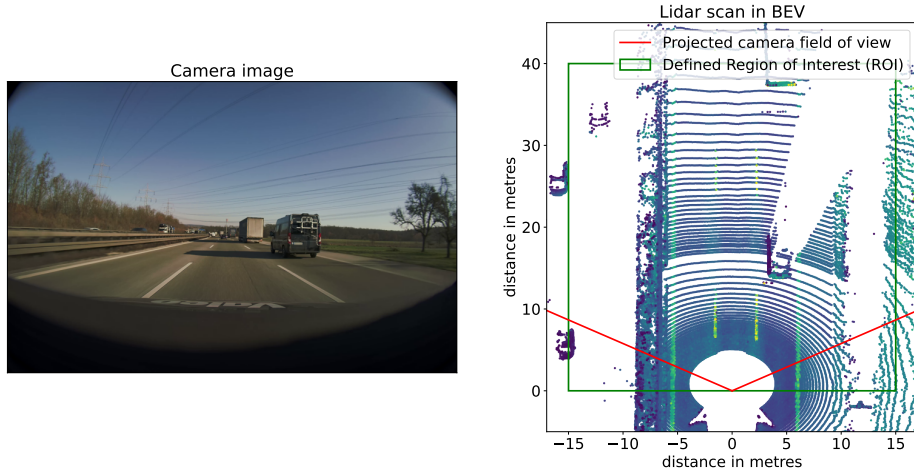


Fig. 1. Image from the front-facing camera (left) and the corresponding LiDAR scan (right).

Fig. 1 illustrates a synchronized snapshot from our dataset, combining the front-facing camera view with the corresponding LiDAR scan in a BEV. The camera’s horizontal field of view is overlaid on the BEV point cloud, highlighting the spatial alignment between modalities. For all downstream tasks, we define a consistent region of interest (ROI) matching K-Lane [16]: from 0 to 40 meters ahead of the vehicle and ± 15 meters laterally.

3.2 Dataset collection

Our dataset comprises of 29 driving traces (sequences), each representing a 60-second segment of continuous driving. Each trace includes approximately 1,800 video frames and cca 1,200 LiDAR scans with synchronized odometry data. In total, the dataset provides 46,802 image frames and 34,831 LiDAR and odometry frames.

A comparison with other publicly available datasets is provided in Table 1.

Dataset	Frames	Beams	Situations	Daytime Weather	Perspective [°]
[22] Waymo	230k	64	urban, highway	day, night rain, snow, fog	360
[26] LiSV-3DLane	20k	64	urban, highway	day, night	360
[16] K-Lane	15k	64	urban, highway	day, night rain, fog	180
Pandar128 (ours)	35k	128	urban, highway	day, rain	360

Table 1. Comparison of publicly available lane line detection datasets.

All data was collected in Germany, with a strong focus on highway driving under sunny and low-traffic conditions. While these scenarios dominate the dataset, it also includes a smaller number of traces featuring rain, construction zones, and mid-traffic conditions, enabling initial exploration of robustness in more challenging settings.

Key characteristics of the dataset based on trace level can be seen in Table 2.

Road Type			Weather Type		
City 3 (10.3%)	Expressway 2 (6.9%)	Highway 24 (82.8%)	Sunny 22 (75.9%)	Cloudy 2 (6.9%)	Rainy 5 (17.2%)
Traffic Level			Roadwork		
Mid-traffic 7 (24.1%)	Low-traffic 22 (75.9%)		No const. 26 (89.7%)	Const. zone 3 (10.3%)	

Table 2. Distribution of scenario attributes across the dataset, broken down by road type, weather conditions, traffic density, and presence of construction zones.

3.3 Annotations

We provide two complementary annotation types for lane lines:

- Semantic segmentation of the point cloud
- Lightweight polyline annotations

While semantic segmentation offers detailed point-level labels, polyline annotations defined as sparse sequences of (x, y, z) coordinates significantly reduce annotation time and cost. Furthermore, polylines are more computationally efficient for certain downstream tasks, such as lane modeling and tracking.

The annotation guidelines were designed to support not only lane detection, but also higher-level behaviors such as vehicle cut-ins, cut-outs, and trajectory planning. Semantic segmentation annotation rules are:

- Annotate all visible lane lines in the ego-lane and adjacent lanes in the ego direction.
- Do not annotate opposite-direction lanes when separated by a physical barrier.
- Include visible parts of exit and acceleration lanes.
- Do not annotate occluded or non-visible lane lines.
- Annotate white lane line with label 1, yellow lane line with label 2 and background as 0.

An example of semantic segmentation annotation can be seen in Fig. 2.

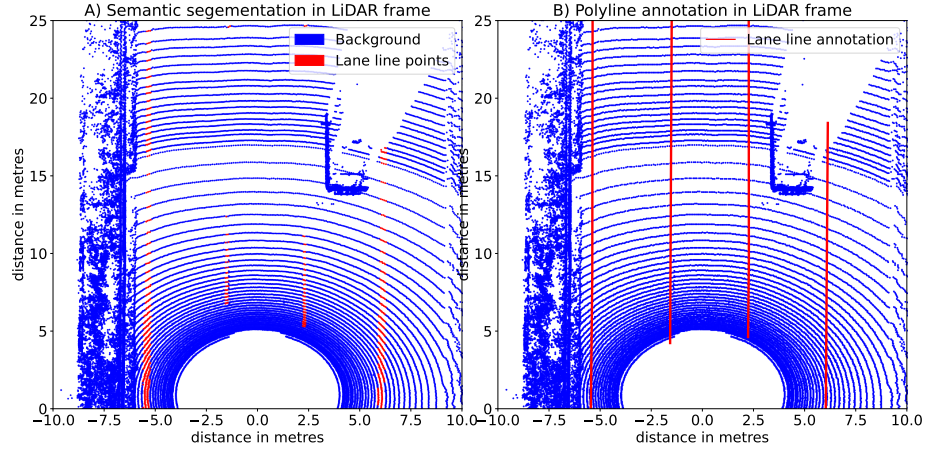


Fig. 2. A) Example of point cloud semantic segmentation annotation and B) Example of polyline annotation.

Polyline annotations are represented as ordered lists of (x, y, z) coordinates in 3D space, allowing fast and lightweight labeling while preserving key structural information. These polylines can optionally be rasterized into a meshgrid format following the approach in K-Lane [16] by mapping nearby mesh cells to the polyline trajectory. An example of our a polyline annotation using (x, y, z) coordinates is shown in Fig. 2. The same annotation rules apply as in the case of semantic segmentation, with the additional constraint that polylines must lie strictly within the physical extent of the lane line, ensuring geometric consistency.

We assessed lane curvature by computing the Pearson correlation between x and y coordinates of each polyline. Values near 1 indicate straight segments, while lower values reflect higher curvature or complex geometries. Aggregated per frame distribution is shown in Fig. 3.

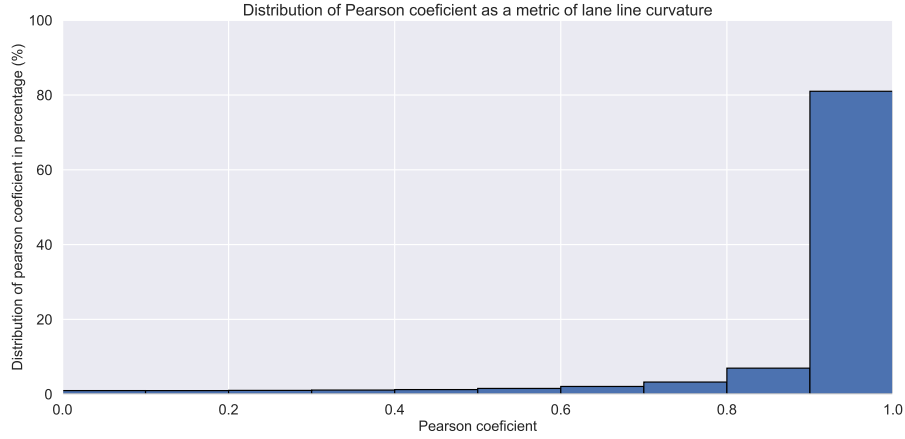


Fig. 3. Distribution of curvature (measured via Pearson coefficient) of annotated lane lines across frames.

Scene complexity was evaluated by counting lane lines per frame and aggregating these counts across all traces. The resulting distribution (Fig. 4) shows the variation in lane configurations.

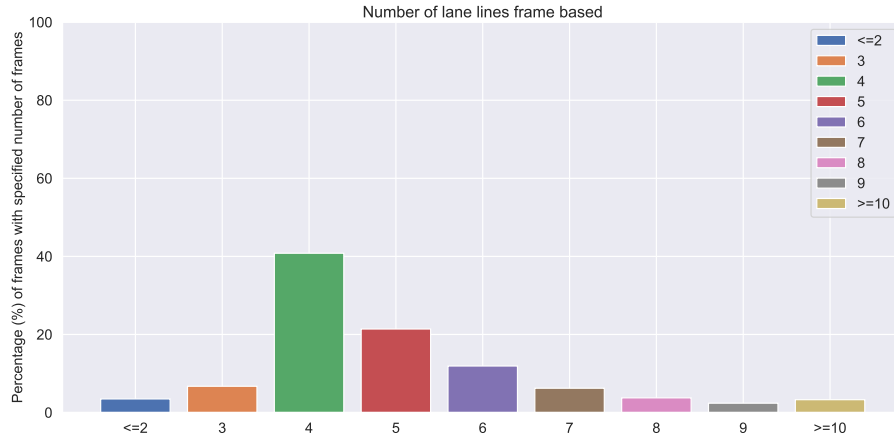


Fig. 4. Distribution of the number of annotated lane lines per frame across the dataset.

3.4 Data Split Strategy

We divided our dataset, which consists of 29 driving traces, into training, validation, and test subsets with careful attention to diversity in road type, weather, traffic density, and construction zones.

To ensure representative evaluation, both the validation and test sets contain 6 traces, each explicitly selected to include at least one challenging condition (e.g., rain, mid-traffic, construction zones, or urban driving).

The remaining 17 traces form the training set and include at least one instance of each rare scenario, supporting generalization to diverse environments.

This stratified split enables consistent benchmarking and promotes robust model evaluation across both common and edge-case driving conditions.

4 Methods

4.1 SimpleLidarLane: Baseline Extraction Method

To provide a lightweight and interpretable baseline for lane line extraction, we propose SimpleLidarLane, a modular pipeline that combines semantic segmentation with classical post-processing techniques. Unlike end-to-end approaches such as K-Lane [16], our method decomposes the task into clearly defined submodules:

1. Project the LiDAR point cloud onto a 2D meshgrid representing a bird’s-eye view (BEV) with a field of view covering 0–40 meters in front of the vehicle and ± 15 meters to the sides, which is similar to the setup used in [16]. The grid resolution is set to 5 cm per cell in real-world units.
2. Perform semantic segmentation using a U-Net [20] architecture with ResNet-18 [11] as backbone.
3. Apply anisotropic scaling by elongating the forward axis and compressing the lateral axis to enhance lane separation.
4. Clustering using DBSCAN [9].
5. Lane line fitting using RANSAC [2].

An overview of the individual steps is illustrated in Fig. 5.

This decomposition enhances interpretability: if the output fails, one can directly analyze whether the issue stems from the perception stage (e.g., misclassified pixels), the clustering (e.g., overlapping lane instances), or the fitting procedure (e.g., inaccurate curves). Such isolation supports targeted debugging and facilitates systematic performance improvements.

Compared to dense segmentation-based pipelines, our polyline representation reduces memory footprint, annotation effort, and aligns better with vector-based planning modules. While advanced methods like LiLaDet [26,16] achieve strong performance using deep 3D regression, and others explore multi-sensor fusion from camera and LiDAR domains [3], our design prioritizes robustness, clarity, and ease of integration.

Ultimately, SimpleLidarLane is intended to serve as a strong and interpretable baseline against which more complex architectures can be compared. Its simplicity, modularity, and efficiency also make it suitable as a practical tool for rapid deployment and prototyping in real-world lane detection applications.

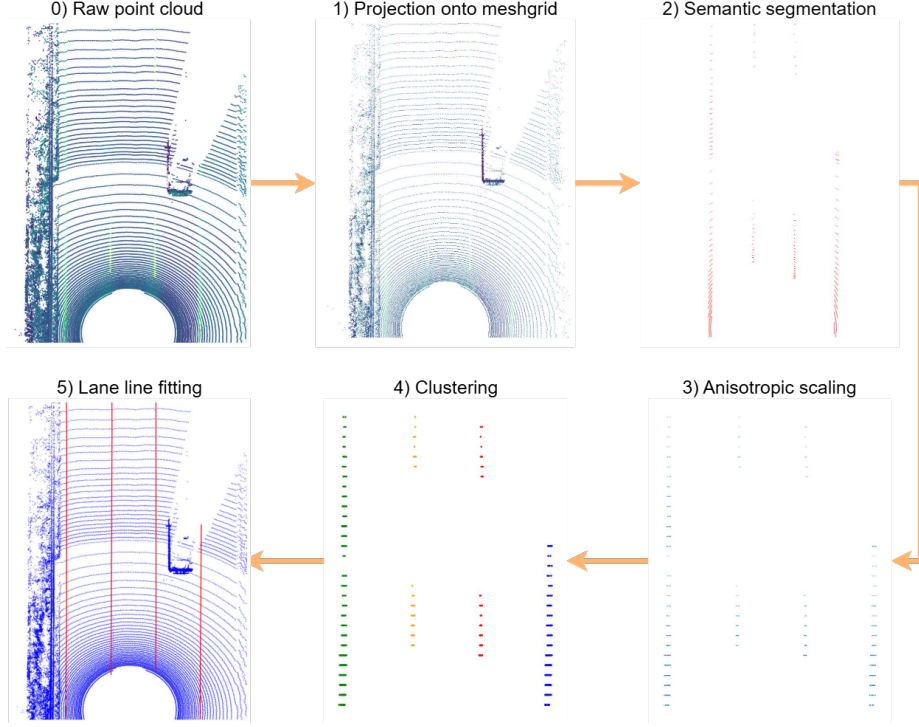


Fig. 5. Architecture of the proposed SimpleLidarLane pipeline.

4.2 Evaluation Metrics

We evaluate lane line extraction using both standard metrics and a novel polyline-based metric that we introduce in this work. The full evaluation framework is composed of:

- **Proposed metric:** Interpolation-Aware Matching F1 (IAM-F1).
- **Standard metrics:** semantic segmentation F1, raster-based polyline F1.

Interpolation-Aware Matching F1 (IAM-F1) We introduce the Interpolation-Aware Matching F1 (IAM-F1), a novel metric designed to quantify the geometric alignment between predicted and ground-truth polylines in the bird’s-eye view (BEV). IAM-F1 focuses on 2D lateral accuracy, which is critical for vehicle control and path planning, and operates exclusively in the (x, y) plane. The vertical (z) component is deliberately omitted, as lane lines are predominantly used for lateral positioning and guidance in the road plane.

Formally, let $\mathcal{G} = \{g_j = (x_j, y_j)\}_{j=1}^N$ denote a ground-truth polyline and $\mathcal{P} = \{p_k = (x_k, y_k)\}_{k=1}^M$ a predicted polyline. Both are represented as ordered sequences of 2D points in BEV space.

For each ground-truth point g_j , we compute the interpolated lateral position y_j^{interp} on $\mathcal{P}$ via linear interpolation at the same longitudinal position x_j :

$$y_j^{\text{interp}} = \text{interp}_{\mathcal{P}}(x_j), \quad \delta_j = |y_j^{\text{interp}} - y_j| \quad (1)$$

If $\delta_j < \tau$ (with $\tau = 0.2\text{m}$), it is considered as a true positive (TP). Otherwise, it is a false negative (FN). Points outside the interpolation domain are also counted as FN.

We repeat the process in reverse: interpolating $\mathcal{G}$ at each x_k from $\mathcal{P}$ to compute false positives (FP).

The metric is then computed as:

$$\text{TP} = |\{g_j \mid x_j \in \text{dom}(\mathcal{P}) \wedge \delta_j < \tau\}| \quad (2)$$

$$\text{FN} = |\{g_j \mid x_j \notin \text{dom}(\mathcal{P}) \vee \delta_j \geq \tau\}| \quad (3)$$

$$\text{FP} = |\{p_k \mid x_k \notin \text{dom}(\mathcal{G}) \vee |y_k - \text{interp}_{\mathcal{G}}(x_k)| \geq \tau\}| \quad (4)$$

In contrast to prior methods, including K-Lane [16], our approach adopts a stricter and explicitly defined matching threshold. IAM-F1 provides a transparent, geometry-aware metric for polyline alignment in BEV space and offers computational advantages by operating on sparse polylines instead of dense raster maps.

A schematic of the interpolation and thresholding process is shown in Fig. 6.

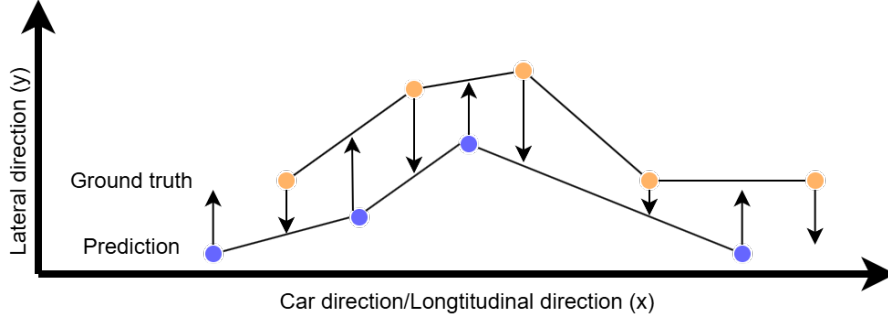


Fig. 6. Illustration of interpolation-aware lateral matching in IAM-F1 metric.

This makes IAM-F1 suitable for scalable evaluation on large datasets.

Standard Metrics In addition to our proposed IAM-F1 metric, we evaluate performance using two standard approaches for comparability with prior work:

1. **Semantic Segmentation F1** — This metric computes the F1 score over meshgrid cells corresponding to annotated lane markings. It reflects the network’s ability to localize lane areas in BEV space at the segmentation level.
2. **Polyline-Based Meshgrid Evaluation** — Inspired by K-Lane [16], we evaluate polylines by discretizing both predicted and ground-truth annotations onto a shared BEV meshgrid. Binary masks are generated to distinguish lane vs. background cells, and standard classification metrics (F1 score, precision, recall) are computed over the resulting occupancy grids. This method measures spatial agreement in a grid-based representation without requiring explicit instance matching.

5 Experiments

5.1 Experimental setup

We train only the semantic segmentation component of the SimpleLidarLane pipeline, which operates on bird’s-eye view (BEV) meshgrid representations of LiDAR point clouds. The BEV grid spans 0–40 meters longitudinally and ± 15 meters laterally with a resolution of 5 cm per cell similar to K-Lane[16], as described in Section 4.1.

We use a U-Net++ [28] segmentation architecture with a ResNet-18 [11] encoder pretrained on ImageNet. The model outputs three semantic classes: background, white lane line, and yellow lane line.

Given the strong class imbalance (lane lines occupy only a small portion of the BEV), we employ focal loss [13] with default settings to improve convergence. No data augmentations are applied.

Training is performed using the Adam optimizer ($\epsilon = 10^{-8}$) with an initial learning rate of 0.01. We apply ReduceLROnPlateau scheduling (patience = 5 epochs, $\epsilon = 0.2$) and early stopping (patience = 10 epochs) based on validation IoU. The batch size is 2 for training and 1 for validation.

The model is trained for up to 400 epochs on an NVIDIA RTX 4060 GPU with 8GB VRAM, 32 GB LPDDR5x RAM, and an AMD Ryzen 7 7840HS CPU. The split into train, validation, and test sets follows the dataset partitioning described in Section 3.4.

5.2 Quantitative Results

Table 3 reports performance across six traces from our test set, evaluated using three F1-based metrics:

- Semantic segmentation F1
- Rasterized polyline F1
- Our proposed IAM-F1.

Segmentation performs well in structured highway scenes but degrades in complex environments such as urban–highway transitions. IAM-F1 consistently yields the highest scores, confirming the benefits of explicit geometric matching.

filename	F1 score		
	Segmentation	Polyline (meshgrid)	Polyline (ours)
0208_105703_008	0.8434	0.8669	0.9136
0208_105703_012	0.7793	0.8547	0.9093
0208_105703_016	0.7419	0.8787	0.9263
0307_160641_111	0.7931	0.8790	0.9093
1019_114352_021	0.7139	0.7657	0.8384
1110_131951_005	0.4195	0.6118	0.5945
Average	0.7152	0.8095	0.8486

Table 3. Comparison of F1 scores for segmentation, meshgrid polyline rasterization, and our final polyline metric (rounded to 4 decimal places per trace).

Unlike raster-based metrics tied to fixed meshgrid resolution (5×5 cm), IAM-F1 compares sparse polylines directly, avoiding quantization artifacts and enabling sharper, more efficient evaluation.

5.3 Qualitative Evaluation

Our SimpleLidarLane achieves an average IAM-F1 score of **80.95%** on the test set, demonstrating consistent performance across diverse driving conditions. The method remains effective even when lane markings are barely visible due to glare, rain, or low illumination. Despite LiDAR sparsity at longer ranges, it reliably reconstructs lane lines up to 40 m, and under adverse weather up to 25 m.

Challenging scenarios include worn or missing markings (e.g., merging zones), sparse returns in rain, and complex road geometries. Fig. 7 visualizes both successful and failure cases. Top-row scenes show accurate polyline reconstruction; bottom-row examples reveal common failure modes, such as lane confusion or missed markings.

These results suggest the pipeline’s potential for practical deployment, while also highlighting the need for improved handling of occlusions, severe weather, and highly curved lane structures.

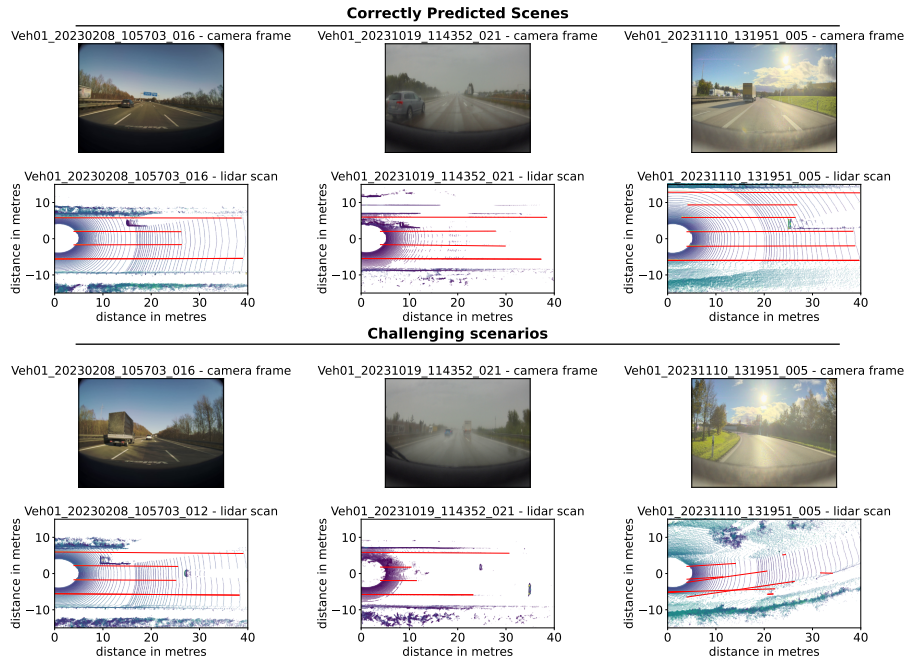


Fig. 7. Qualitative results showing successful (top row) and challenging (bottom row) predictions across different driving scenarios.

6 Conclusion

We introduced Pandar128, the largest public LiDAR-based dataset for lane line detection, featuring over 34k annotated LiDAR frames with high-resolution scans, synchronized odometry, and comprehensive calibration data. This enables advanced research in geometric lane modeling, sensor fusion, and long-range perception.

To benchmark this dataset, we proposed SimpleLidarLane, a lightweight baseline pipeline combining BEV segmentation with clustering and polyline fitting. Despite its simplicity, it achieves strong performance across diverse driving scenarios, including challenging weather and illumination conditions.

In addition, we introduced IAM-F1, a novel interpolation-aware metric for evaluating polyline alignment in BEV space. Unlike raster-based metrics, IAM-F1 directly compares sparse polylines, offering improved geometric fidelity and computational efficiency.

Together, the dataset, baseline, and evaluation protocol establish a new foundation for reproducible and interpretable research in LiDAR-based lane detection. All resources are publicly available to encourage further development in this critical area of autonomous driving.

References

1. Tusimple (2017), <https://github.com/TuSimple/tusimple-benchmark>
2. Arias-Castro, E., Wang, J.: Ransac algorithms for subspace recovery and subspace clustering (2017), <https://arxiv.org/abs/1711.11220>
3. Bai, M., Mattyus, G., Homayounfar, N., Wang, S., Lakshmikanth, S.K., Urtasun, R.: Deep multi-sensor lane detection (2019), <https://arxiv.org/abs/1905.01555>
4. Certad, N., Morales-Alvarez, W., Olaverri-Monreal, C.: Road markings segmentation from lidar point clouds using reflectivity information (2025), <https://arxiv.org/abs/2211.01105>
5. Commission, E.: New rules to improve road safety and enable fully driverless vehicles in the eu (2022), https://ec.europa.eu/commission/presscorner/api/files/document/print/en/ip_22_4312/IP_22_4312_EN.pdf
6. Deng, G., Wu, Y.: Double lane line edge detection method based on constraint conditions hough transform. In: 2018 17th International Symposium on Distributed Computing and Applications for Business Engineering and Science (DCABES). pp. 107–110 (2018). <https://doi.org/10.1109/DCABES.2018.00037>
7. Doe, J., Smith, J.: Robust lidar-based slam under low-light and glare conditions. *Journal of Autonomous Vehicle Systems* **12**(3), 123–135 (2024). <https://doi.org/10.1016/j.javs.2024.01.015>, placeholder reference for LiDAR robustness under low-light and glare
8. Dong, Y., Xiong, J., Li, L., Yang, J.: Robust lane detection and tracking for lane departure warning. In: 2012 International Conference on Computational Problem-Solving (ICCP). pp. 461–464 (2012). <https://doi.org/10.1109/ICCP.2012.6384266>
9. Ester, M., Kriegel, H.P., Sander, J., Xu, X., et al.: A density-based algorithm for discovering clusters in large spatial databases with noise. In: *kdd*. vol. 96, pp. 226–231 (1996)
10. Fathy, M., Ashraf, N., Ismail, O., Fouad, S., Shaheen, L., Hamdy, A.: Design and implementation of self-driving car. vol. 175, pp. 165–172 (01 2020). <https://doi.org/10.1016/j.procs.2020.07.026>

11. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR). pp. 770–778 (2016)
12. Liao, Y., Xie, J., Geiger, A.: Kitti-360: A novel dataset and benchmarks for urban scene understanding in 2d and 3d (2022), <https://arxiv.org/abs/2109.13410>
13. Lin, T.Y., Goyal, P., Girshick, R., He, K., Dollár, P.: Focal loss for dense object detection (2018), <https://arxiv.org/abs/1708.02002>
14. Liu, L., Chen, X., Zhu, S., Tan, P.: Condlanenet: a top-to-down lane detection framework based on conditional convolution (2023), <https://arxiv.org/abs/2105.05003>
15. Mi, X., Dong, Z., Cao, Z., Yang, B., Cao, Z., Zheng, C., Stoter, J., Nan, L.: A benchmark approach and dataset for large-scale lane mapping from mls point clouds. *International Journal of Applied Earth Observation and Geoinformation* **133**, 104139 (2024). <https://doi.org/https://doi.org/10.1016/j.jag.2024.104139>, <https://www.sciencedirect.com/science/article/pii/S156984322400493X>
16. Paek, D.H., Kong, S.H., Wijaya, K.T.: K-lane: Lidar lane dataset and benchmark for urban roads and highways. In: 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW). p. 4449–4458. IEEE (Jun 2022). <https://doi.org/10.1109/cvprw56347.2022.00491>, <http://dx.doi.org/10.1109/CVPRW56347.2022.00491>
17. Paek, D.H., Wijaya, K.T., Kong, S.H.: Row-wise lidar lane detection network with lane correlation refinement (2022), <https://arxiv.org/abs/2210.08745>
18. Pan, X., Shi, J., Luo, P., Wang, X., Tang, X.: Spatial as deep: Spatial cnn for traffic scene understanding (2017), <https://arxiv.org/abs/1712.06080>
19. Qin, Z., Wang, H., Li, X.: Ultra fast structure-aware deep lane detection (2020), <https://arxiv.org/abs/2004.11757>
20. Ronneberger, O., Fischer, P., Brox, T.: U-net: Convolutional networks for biomedical image segmentation (2015)
21. Sarker, S., Maples, B., Li, W.: A comprehensive review on traffic datasets and simulators for autonomous vehicles. *arXiv preprint arXiv:2412.14207* (2024)
22. Sun, P., Kretschmar, H., Dotiwalla, X., Chouard, A., Patnaik, V., Tsui, P., Guo, J., Zhou, Y., Chai, Y., Caine, B., et al.: Scalability in perception for autonomous driving: Waymo open dataset. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 2446–2454 (2020)
23. Tabelini, L., Berriel, R., Paixão, T.M., Badue, C., Souza, A.F.D., Oliveira-Santos, T.: Keep your eyes on the lane: Real-time attention-guided lane detection (2020), <https://arxiv.org/abs/2010.12035>
24. Yogamani, S., Hughes, C., Horgan, J., Sistu, G., Varley, P., O’Dea, D., Uricar, M., Milz, S., Simon, M., Amende, K., Witt, C., Rashed, H., Chennupati, S., Nayak, S., Mansoor, S., Perroton, X., Perez, P.: Woodscape: A multi-task, multi-camera fisheye dataset for autonomous driving (2021)
25. Zhang, Y.: Perception and sensing challenges for adas under adverse weather. *Sensors* (2023)
26. Zhao, R., Heng, Y., Wang, H., Gao, Y., Liu, S., Yao, C., Chen, J., Cai, W.: Advancements in 3d lane detection using lidar point clouds: From data collection to model development. *arXiv preprint arXiv:2309.13596* (2023)
27. Zhao, R., Heng, Y., Wang, H., Gao, Y., Liu, S., Yao, C., Chen, J., Cai, W.: Advancements in 3d lane detection using lidar point clouds: From data collection to model development. In: 2024 IEEE International Conference on Robotics and Automation (ICRA). pp. 5382–5388. IEEE (2024)

28. Zhou, Z., Siddiquee, M.M.R., Tajbakhsh, N., Liang, J.: Unet++: A nested u-net architecture for medical image segmentation (2018)
29. Ziebinski, A., Cupek, R., Grzechca, D., Chruszczyk, L.: Review of advanced driver assistance systems (adas). vol. 1906, p. 120002 (11 2017). <https://doi.org/10.1063/1.5012394>
30. Żywanowski, K., Banaszczyk, A., Nowicki, M.: Comparison of camera-based and 3d lidar-based loop closures across weather conditions. arXiv preprint arXiv:2009.03705 (2020)