Cirrus: A Long-range Bi-pattern LiDAR Dataset

Ze Wang[1], Sihao Ding[2], Ying Li[2], Jonas Fenn[2], Sohini Roychowdhury[2], Andreas Wallin[2], Lane Martin[3], Scott Ryvola[3], Guillermo Sapiro[4], and Qiang Qiu[1]

ra
ta
ar
a
w
th
si
pa
ra
re
ar
er
by
ac
Promising results show the great potential of this new dataset to the robotics and computer vision communities.

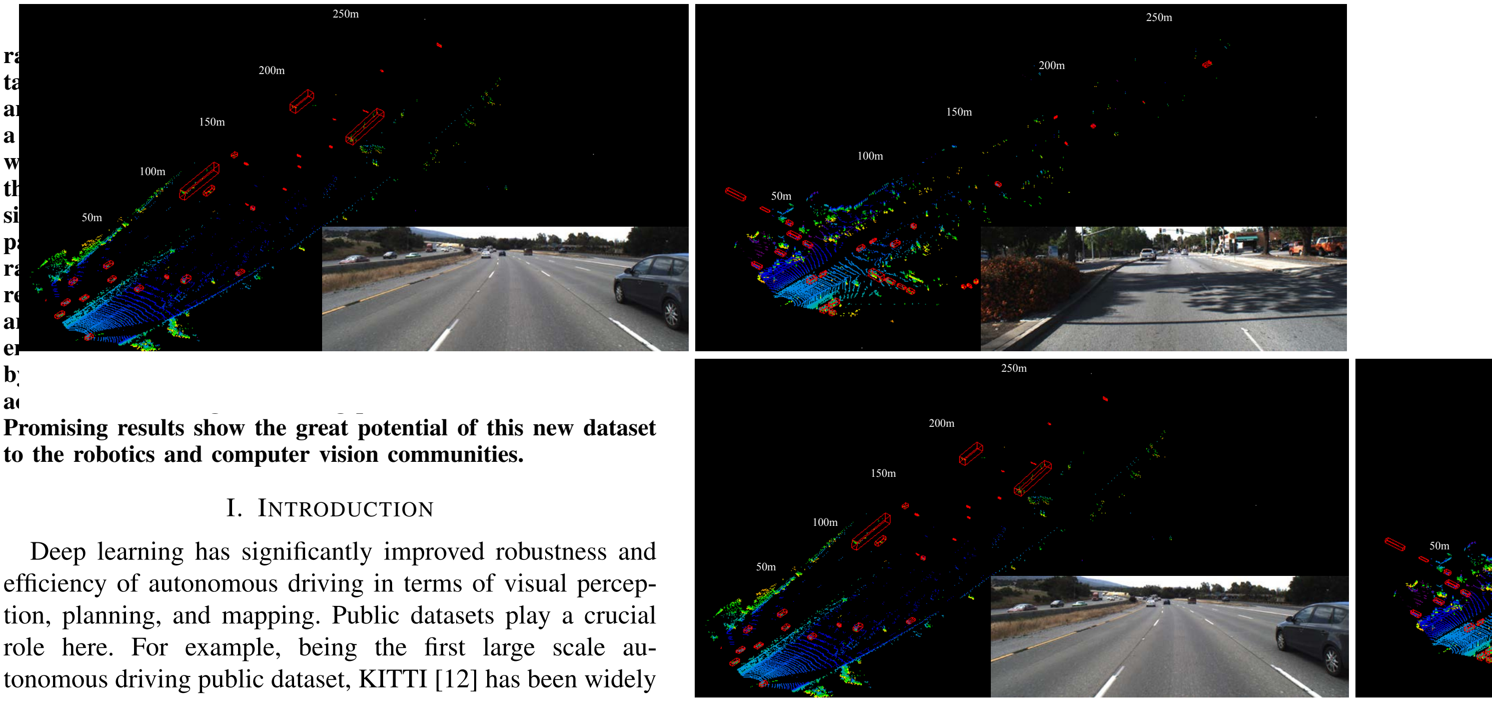


Fig. 1: Example LiDAR point clouds from the Cirrus dataset with bounding boxes. Distance is marked in white.

I. Introduction

Deep learning has significantly improved robustness and efficiency of autonomous driving in terms of visual perception, planning, and mapping. Public datasets play a crucial role here. For example, being the first large scale autonomous driving public dataset, KITTI [12] has been widely adopted for developing and evaluating many state-of-the-art autonomous driving algorithms, or their key components such as object recognition, in the past several years. However, it remains unconfirmed if these algorithms generalize well to unseen scenarios, e.g., with different scanning patterns or range, mostly due to the lack of corresponding datasets for validation.

In this paper, we introduce Cirrus, a new long-range bi-pattern LiDAR dataset for autonomous driving tasks. Cirrus is developed to enhance existing public LiDAR datasets with additional diversity in terms of sensor model, range, and scanning pattern. A long effective range allows object detection at a far distance and leaves sufficient time to react, especially in high-speed driving scenarios. While being constrained by sensor capability, existing datasets usually contain point clouds of limited ranges, e.g., 120m for KITTI, and 70m for nuScenes [3], and largely restrict trained algorithms to low-speed driving scenarios. When cars drive at 75 mph, the 120m effective range of KITTI and the 70m of nuScenes allow only 3.5s and 2s reaction time, respectively. Thus, in Cirrus, we adopt LiDAR sensors of a 250-meter effective range, as shown in Figure 1, to better support developing and evaluating algorithms for high-speed scenarios. We present a side-by-side visualization of point clouds from different datasets in a bird-eye view in Figure 2. Note that, in a point cloud, with respect to distance, the object size stays constant, but the point density varies. Therefore the long effective range of the new dataset provides rich samples with various degrees of point densities, serving a good benchmark for developing algorithms robust across ranges.

In the Cirrus dataset, data are collected using two scanning patterns. Besides the standard uniform pattern, the Gaussian pattern gives extra flexibility by enabling sampling with a focus on a particular direction. For example, on urban roads, cars drive in a relatively complex environment space, but at a relatively low speed. In this case, the uniform scanning pattern can provide perception from a wide viewing angle at a reduced range. While driving on highways, cars move at a much higher speed, but the environment is considerably

[1] Ze Wang and Qiang Qiu are with Purdue University, USA. {zewang, qqiu}@purdue.edu

[2] Sihao Ding, Ying Li, Jonas Fenn, Sohini Roychowdhury, and Andreas Wallin are with Volvo Cars Technology, USA. {sihao.ding, ying.li.5, jonas.fenn, sohini.roy.chowdhury, andreas.wallin1}@volvocars.com

[3] Lane Martin and Scott Ryvola are with Luminar Technologies, Inc., USA. {lane, scott.ryvola}@luminartech.com

[4] Guillermo Sapiro is with Duke University, USA. guillermo.sapiro@duke.edu

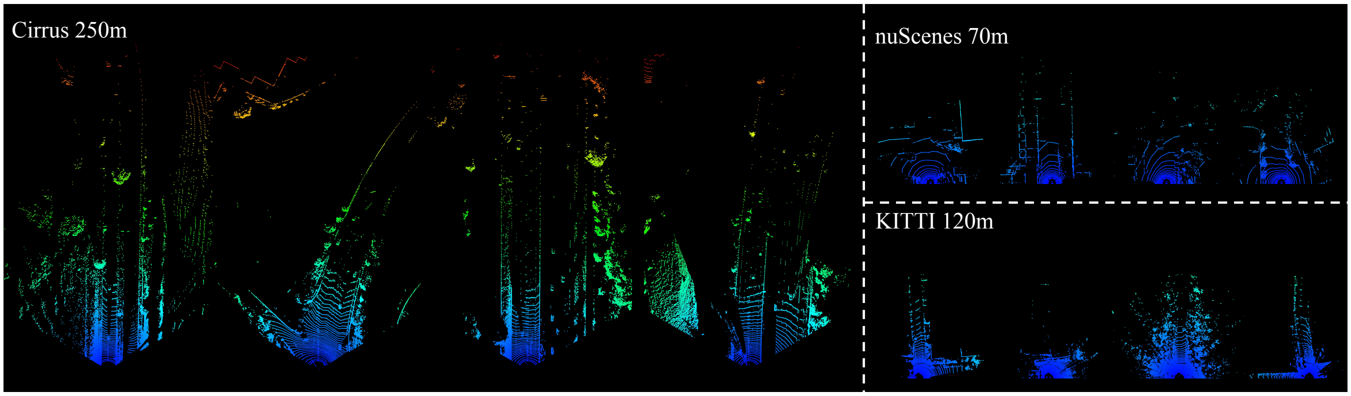


Fig. 2: Point cloud samples from different datasets. Warmer colors indicate longer point distances. The respective effective ranges are marked. When cars drive at 75mph, the 120m, 70m, and 250m effective ranges allow 3.5s, 2s and 7.5s reaction time, respectively.

simpler. Gaussian scanning pattern can enable longer range detection by focusing on the forward direction, thus allowing for longer reaction time in cases of emergency. This new dataset enables exploring the merits of each scanning pattern, so that two patterns can be adaptively switched, ideally with a single underlying analysis model through adaptation, based on the driving scenarios for an optimized range-angle trade-off.

The here introduced Cirrus public[1] dataset provides fully annotated long-range and bi-pattern paired point clouds, and enables several potential research topics with great practical impacts. Based on the aforementioned unique properties, we adopt 3D object detection as a sample application, and perform a series of preliminary experiments on cross-range, cross-device, and cross-pattern adaptations, to illustrate important properties and potential usages of this dataset. We report results on standard 3D object detection in LiDAR, to serve as a baseline for follow-up studies. We invite the community to together explore the Cirrus dataset on various tasks for autonomous driving.

Our main contributions are summarized as follows:

- We introduce a new long-range bi-pattern LiDAR dataset with exhaustive eight-category object annotations in point clouds over the entire 250-meter sensor effective range.
- The proposed dataset contains paired point clouds collected simultaneously with Gaussian and uniform scanning patterns, which enables studies on cross-pattern adaptation in point clouds.
- We adopt 3D object detection in LiDAR as a sample task, and conduct extensive experiments to study model adaptations across ranges, devices, and scanning patterns. Promising results show the great value of this new dataset for future research in the vision and robotics communities.

II. Related Work

In the past few years, large scale annotated datasets have greatly boosted the research on the perception of the autonomous driving. Datasets with various sensor setups have been introduced as the development tools for autonomous driving systems. An RGB camera is the most prevalent sensor thanks to its advantages including the low cost in terms of both hardware and annotations, and the tremendous existing research achievements in computer vision, thus is widely adopted in public datasets [2], [4], [9], [36]. The rich appearance information in RGB images makes it a suitable choice for inferring semantic. The works [2], [9] provide high-quality pixel-level annotations, and are widely adopted in the research of semantic segmentation for autonomous driving. Especially the 5k images with fine annotations and large-scale coarsely annotated samples have paved the way for deep learning based driving support algorithms [5], [22], [24]. Recently, newly released large scale datasets like BDD100K [36] and D^2−city [4] further enrich the diversity of public datasets by including samples collected under different weathers. Recent datasets like Apolloscape [17] with 144k annotated samples, BDD100K [36] with 100k annotated samples, and Mapillary Vistas [23] with 25k samples, significantly enlarge the scale of data for model training. However, the significant drawbacks of images-only datasets largely restrict the real-world performance of the image-only systems. First of all, the inference of distance information from the images is inherently non-trivial and the precision cannot be guaranteed. And the fastly decreased object size in images with respect to the object distance makes it an undesired choice for detecting long-range objects, and therefore is unsuitable for high-speed scenarios where ahead planning is crucial.

To compensate the drawbacks of RGB cameras, object detection with multiple cameras or sensors other than RGB cameras became a popular direction. LiDAR is widely adopted for the perception of autonomous driving for the significant advantages including precise localizing and distance measurement, relatively lower noise comparing to RGB

[1] The full dataset will be released upon manuscript acceptance. Examples are available at `https://developer.volvocars.com/open-datasets`.

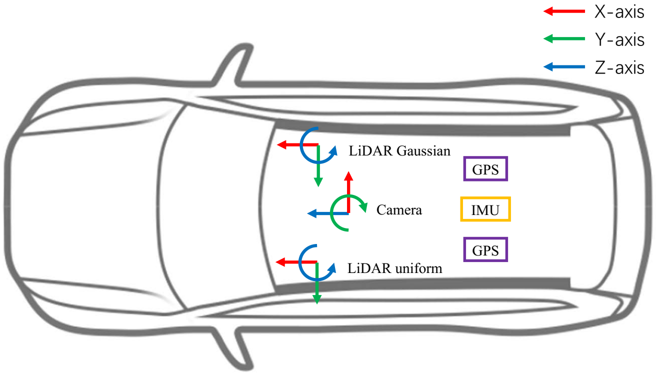


Fig. 3: Sensor placements for the dataset collection. All coordination axes follow the right hand rule.

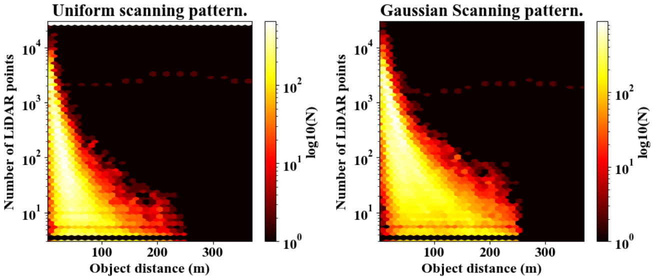


Fig. 4: Hexbin log-scaled density plots of the number of LiDAR points inside annotation boxes.

images, and the ability to fully perceive multiple angles with a single sensor. With the fast development of the deep network based point cloud processing methods [26], [27], [33] and sparse convolution [21], [13], [14], [16], [15], various algorithms [37], [31], [35], [19] are developed to efficiently detect object in point clouds. The advantages of the LiDAR sensors and the algorithms make them indispensable components for modern autonomous driving systems, algorithms [1], [6], [19], [20], [32], [35], [37], and multi-modality datasets [3], [7], [8], [12], [25]. KITTI [12] provides over 7K annotated samples collected with cameras and LiDAR. The stereo images and GPS/IMU data further enables various tasks for autonomous driving. The KAIST dataset [8] provides data with RGB/thermal camera, RGB stereo, LiDAR, and GPS/IMU, and high diversity with samples collected in both daytime and nighttime. However, the practical value of KAIST is largely restricted by its limited size and 3D annotations. The intrinsic limitation of LiDAR also cannot be neglected. Apart from the high cost on hardware, the effective range becomes a bottleneck to the wide adoption of LiDAR in all environments. Radar sensors, as another popular range sensor, have much longer perception range (250m typically), and lower hardware costs. But the low point density of radar sensors prevents it from being a qualified replacement to LiDAR. So far only the nuScenes dataset [3] provides point clouds collected with radars.

With the help of modern computer graphics and game engines, synthesized datasets like Playing for Data [28], CARLA [10], Virtual KITTI [11], and SYNTHIA [29], reduce the cost of collecting data, although the performance are sometimes restricted due to the domain shifts between synthesized and real-world data.

III. The Cirrus Dataset

The Cirrus dataset contains 6,285 synchronized pairs of RGB, LiDAR Gaussian, and LiDAR uniform frames. All samples are fully annotated for eight object categories across the entire 250-meter LiDAR effective range.

A. Sensor Placements

The data collection car is equipped with the following sensors:

- An RGB camera with a resolution of 1920 × 650.
- A Luminar Model H2 LiDAR sensor with the Gaussian scanning pattern, 10Hz, 64 lines per frame, 1550-nm, 250m effective range, $>$ 200 meters range to 10% reflective target (Lambertian), 120^o horizontal FOV, 30^o vertical FOV.
- A Luminar Model H2 LiDAR sensor with the uniform scanning pattern, 10Hz, 64 lines per frame, 1550-nm, 250m effective range, $>$ 200 meters range to 10% reflective target (Lambertian),120^o horizontal FOV, 30^o vertical FOV.
- IMU and GPS × 2.

The sensor placements are illustrated in Figure 3.

B. Scanning Patterns

Two LiDAR sensors mounted on the car are of the identical model, each running a particular scanning pattern. Point clouds are simultaneously captured using both uniform and Gaussian scanning patterns. These two sensors are calibrated to have synchronized and aligned point clouds, and thus annotations can be shared across patterns.

For the LiDAR with the Gaussian scanning pattern, we set the focus of sweeps to the forward direction of the car, which in return, gives higher point density to objects ahead than the uniform pattern. We plot the Hexbin log-scaled density for both patterns in Figure 4. It is clearly shown in the plot that point clouds collected with the Gaussian pattern have higher overall point density inside annotation boxes. For long-range objects that are more than 200 meters away, the significantly higher point density in Gaussian pattern point clouds can potentially enable more accurate estimation of object attributes and categories.

C. Sensor Synchronization

The exposure of the camera is triggered first with the corresponding time stamp captured. When the LiDAR sensors start firing and getting returns, the times stamps are generated as well. Then these separate sets of timestamps are sent to the computing platform (such as NVIDIA DRIVE PX2) and the gPTP protocol [18] is followed to sync these time stamps. For instance, the camera yields 30 time stamps per second, and each LiDAR sensor gives 10 per second. The nearest matching/syncing across timestamps happens in the PX2. This is a continuous streaming process.

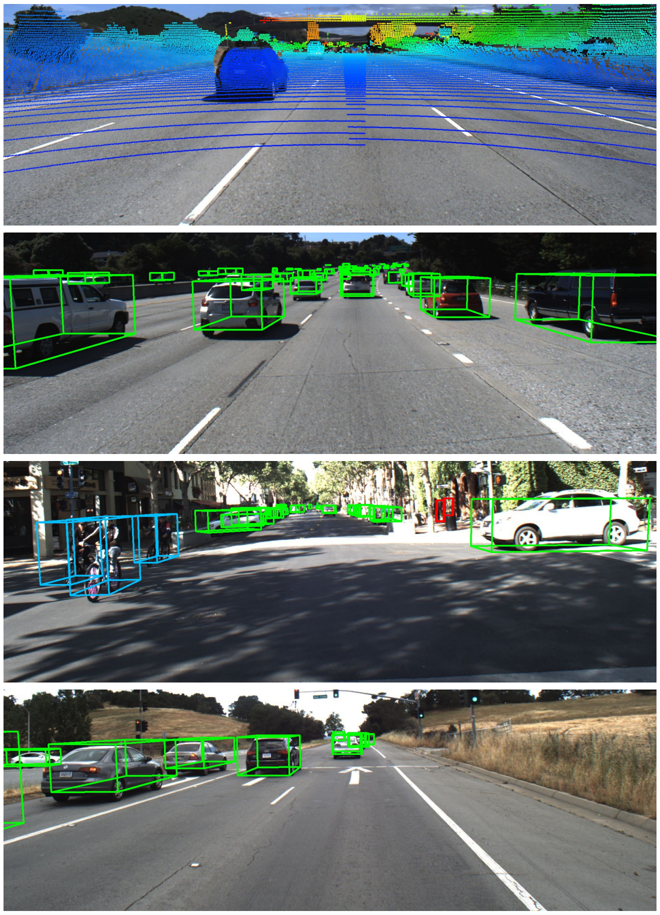
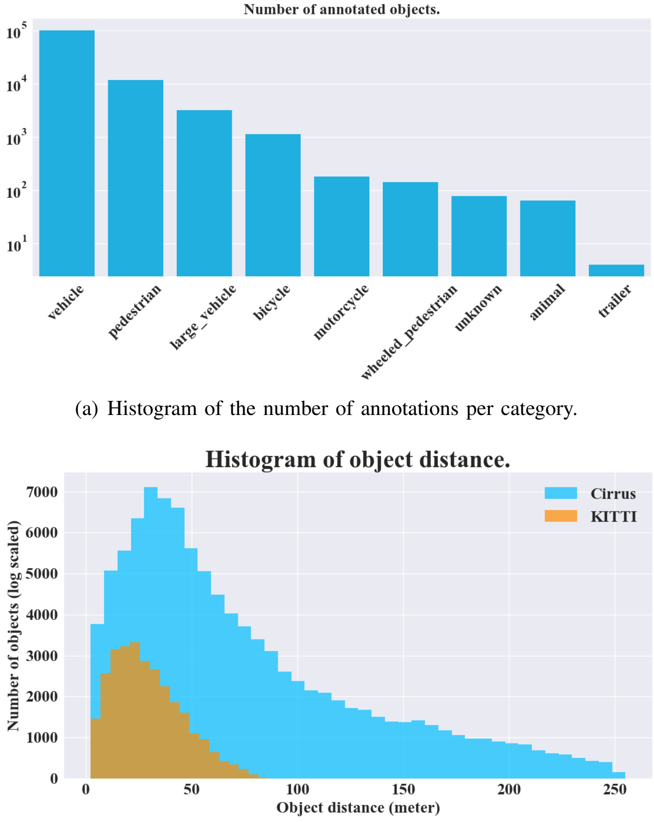


Fig. 5: Diverse scenes including both highway and urban-road scenarios are included in the Cirrus dataset. We include in the first sample the projected point cloud collected using Gaussian scanning pattern, and then we visualize all the projected boxing boxes in the rest of the samples, where red, blue, and green boxes denote pedestrians, bicycles, and vehicles, respectively. The top-right sample clearly shows the high detection range of the Cirrus dataset where large amount of objects are annotated exhaustively.



(a) Histogram of the number of annotations per category.

(b) Histogram of object distance and comparison against KITTI.

Fig. 6: Histogram of annotations (top) and comparison with KITTI (bottom).

D. Annotations

Cirrus provides 6,285 frames from 12 segments of videos. Both high-speed highway and low-speed urban-road scenarios are included. Example scenes are presented in Figure 5. All images have gone through an anonymization process blurring faces and license plates to eliminate personally identifiable information.

We annotate 8 categories of objects: *vehicle, large vehicle, pedestrian, bicycle, animal, wheeled pedestrian, motorcycle*, and *trailer*. Objects that do not belong to the aforementioned categories are annotated as *unknown*. The statistics of the annotated categories are plotted in Figure 6(a).

For each object from known categories, we annotate its spatial position as $\{x, y, z\}$ in the LiDAR coordinate. The shape of each object is represented by its length, width, and height, as well as the rotation angle (yaw) represented using a quaternion. Different from the previous datasets, where the full body of each object is tightly annotated with a 3D bounding box, the boxes in our dataset contain object parts that are visible in the point clouds. Since we focus particularly on far-range object detection, and many annotations are beyond visual range, it is hard to infer the full body of every object especially when an object is at a long distance and cannot be seen clearly in the RGB image.

We plot the histogram of object distances for the vehicle (car) category in Figure 6(b), where it is clearly shown that a large amount of objects appear across the 250-meter effective range. Note that the farthest annotation reaches a distance of over 350 meters. The histogram comparison against KITTI [12] is also included in Figure 6(b). Cirrus provides significantly larger amount of vehicles and objects are widely spread across the longer effective range comparing to KITTI.

IV. 3D Object Detection and Model Adaptation

In this section, we present sample tasks, 3D object detection and model adaptation on the newly introduced Cirrus dataset, to illustrate its unique properties and potential usages. We start this section with evaluation metrics we adopt for this new dataset, and then briefly introduce a benchmark setting for detection and adaptation experiments. All baseline results will be presented in Section V.

TABLE I: 3D object detection with VoxelNet and the Cirrus dataset.

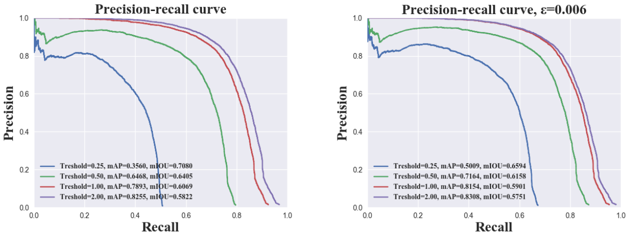
Pattern	Metric	Near range	Mid range	Far range	Overall
Gaussian	DDS	0.6954	0.5594	0.3412	0.6444
	wDDS ($\epsilon = 0.003$)	0.7112	0.5612	0.3517	0.6546
	wDDS ($\epsilon = 0.006$)	0.7232	0.5885	0.3884	0.6630
	wDDS ($\epsilon = 0.009$)	0.7421	0.5904	0.3908	0.6691
Uniform	DDS	0.6297	0.4888	0.3012	0.5672
	wDDS ($\epsilon = 0.006$)	0.6618	0.5013	0.3234	0.5908

TABLE II: 3D object detection with state-of-the-art methods. $\epsilon = 0.006$ is set as default for wDDS.

Methods	DDS	wDDS
VoxelNet	0.6444	0.6630
SECOND	0.6621	0.6819
PointRCNN	0.6672	0.6788

TABLE III: Cross-range adaptation. Model adaptation improves the detection performance across the entire effective range.

Range	DDS	wDDS
Near range	0.7237	0.7291
Far range	0.6018	0.6098
Overall	0.6543	0.6825



(a) Mean Average Precision.

(b) Mean Average Precision with weighted threshold ($\epsilon = 0.006$).

Fig. 7: Precision-recall curves.

A. Evaluation Metrics

3D object detection requires accurate estimation of object category, location, and pose. As pointed out in [3], 3D intersection over union (IoU) is sensitive to minor drifts for objects with small spatial dimensions such as pedestrians. In our case, since we adopt partial annotations for objects that are not fully visible in the point clouds, we observe unstable assessments for long-range object detection using IoU, which is thus unreliable to faithfully quantify the algorithm performance. Inspired by the nuScenes detection score (NDS) in [3], we propose a new decoupled detection score (DDS) that independently evaluates mean average precision for spatial locations, and box attribute estimation for spatial dimensions and poses. DDS is computed as

$$\mathrm{DDS} = \frac{1}{2}(\mathrm{mAP} + \mathrm{aIoU}), \tag{1}$$

where mAP and aIoU are mean average precision and aligned intersection over union, respectively, as detailed next.

Mean Average Precision. The partial annotations for the limited-visibility objects including small, occluded, and far distant objects, make the standard IoU based mAP metric over-sensitive to small spatial drifts on the prediction. In order to decouple the object localization and the box attribution estimation, we use mAP to independently evaluate the precision of object location prediction. Specifically, following [3], a match in average precision is defined by threshholding the 3D spatial distance between the ground-truth and predicted object center d in 3D coordinates. AP is calculated as the normalized area under the precision-recall curve. The final mAP is calculated by the average over the multiple matching threshold $\mathbb{T} = \{0.5, 1, 2, 4\}$ as

$$\mathbf{mAP} = \frac{1}{|\mathbb{T}|}\sum_{t\in\mathbb{T}} \mathbf{AP}_t. \tag{2}$$

Aligned Intersection over Union. With the precision of location prediction measured by mAP, we now introduce aligned intersection over union (aIoU) as the metric to measure the precision of box attribute estimation. We calculate aIoU as the IoU after aligning the 3D center point of the predicted and the ground-truth object boxes. In this way, aIoU only considers the precision of the box shape in terms of both the object dimensions and yaw angle. The spatial drifts are not included in the calculation of aIoU since they are already measured by mAP.

Weighted Decoupled Detection Score. One of the unique properties of our new dataset is the long effective range and the exhaustive annotations across the entire range. In practice, we consistently observe that the overall performance on the Cirrus dataset is largely constrained by the far-ranges objects which have low-density points and severe occlusions. To fairly evaluate the performance cross all ranges, we further propose a weighted decoupled detection score (wDDS), which uses a dynamic threshold for objects at different distances. Specifically, we introduce a new weight factor ω, which is calculated as

$$\omega = \exp\left(\epsilon \cdot d\right), \tag{3}$$

where d is the distance to an object on a 2D plane, ϵ is a positive constant, and $\epsilon = 0$ equals to DDS without weighting. The weight factor ω is applied by multiplying it with the matching threshold $\mathbb{T}$, so that objects at a far distance will have relatively large thresholds, and the matching thresholds for the objects at a near distance remain close to $\mathbb{T}$.

The proposed DDS and wDDS consider only single object category without taking into account the classification accuracy. When jointly detecting multiple classes of objects, DDS and wDDS for each category are calculated and reported separately.

V. Experiments

In this section, we adopt 3D object detection as a sample application, and perform a series of model adaptation experiments. *Vehicle* (including *large vehicle*) is selected as the target category to detect. Vehicles dominate the objects annotated in our dataset, and their wide existence across the entire effective range in the current version of our dataset provides reliable assessments to the algorithm performance. We start with standard 3D object detection and report baseline performance obtained on state-of-the-art 3D object detection methods including VoxelNet [37], SECOND [35], and PointRCNN [30]. Various model adaptation experiments, including cross-device, cross-pattern, and

TABLE IV: Cross-device adaptation experiments performed on KITTI → Cirrus. Different amounts of annotated data in Cirrus are marked in the table.

Methods	DDS	wDDS
Pretrained	0.6506	0.6612
Model adaptation (25%)	0.6511	0.6606
Model adaptation (50%)	0.6609	0.6824
Model adaptation (full)	0.6688	0.6904

TABLE V: Cross-pattern compatibility. G and U denote Gaussian and uniform, respectively.

Pattern	DDS	wDDS
G → U	0.5492	0.5523
U → G	0.5514	0.5713

TABLE VI: Cross-pattern model performance after joint training and model adaptation.

Method	Joint training		Adaptations	
Metric	DDS	wDDS	DDS	wDDS
Gaussian	0.6692	0.6770	0.6865	0.6991
Uniform	0.5914	0.6107	0.6057	0.6212

cross-range, are then conducted to validate the value of the proposed benchmark for future research in LiDAR. VoxelNet [37] provides a principle way of efficient object detection in point clouds, and is used as the baseline network for performing adaptation experiments. And a Gaussian scanning pattern is selected as the default pattern besides the cross-pattern adaptation, where data for both patterns are used.

A. 3D Object Detection

We start with standard 3D object detection using the new Cirrus dataset. We train VoxelNet to detect vehicles represented by bounding boxes with 3D location, 3D dimension, and yaw angle in LiDAR. To produce gridded features for convolutional layers, following [37], we convert the point clouds into equally spaced 3D voxels. The detection range is set to be $[0, 250] \times [-50, 50] \times [-3, 1]$ meters along the X, Y, Z axis, respectively. The voxel size is set to be vW = 0.2, vH = 0.2, vD = 0.4 meters, which leads to gridded feature maps with a size of $1250 \times 500 \times 10$ that allow accurate box location estimation at high resolution feature maps.

Models for point clouds with Gaussian and uniform scanning patterns are trained separately. The results are presented in Table I. To comprehensively evaluate the algorithm robustness across range, we divide the 250 meter effective range into three levels: 0-70 meters as the near range, 70-150 meter as the mid range, and 150-250 meters as the far range. The performance for each range is reported separately, followed by an overall performance across the entire detection range. We report performance measured by both DDS and wDDS with 4 values of ϵ, and the precision-recall curves with $\epsilon = 0$ and $\epsilon = 0.006$ are plotted in Figure 7(a) and Figure 7(b). We select $\epsilon = 0.006$ as the default setting of wDDS in the following experiments.

We further provide results on more state-of-the-art methods in Table II for benchmarking future methods.

B. LiDAR Model Adaptations

Cross-range Adaptation We firstly show that, for a long effective range, we can improve the overall algorithm robustness by encouraging consistent deep features across the entire range. We adopt the framework of range adaptation proposed in our previous work [34] for promoting consistent feature across range both locally and globally. We use point clouds with Gaussian pattern, and divide the 250-meter effective range into two areas, with 0-100 meter as the near range and 100-250m as the far range to perform cross-range adaptation from near range to far range. The results are presented in Table III; we report DDS and wDDS after adaptation on near-range, mid-range, and far-range areas. The performance improvements indicates that the invariant feature benefits object detection across the entire effective range.

Cross-device Adaptation. We now consider a more challenging setting, where the cross-domain data is collected by different sensor models. We adopt point clouds in KITTI, which are collected using LiDAR sensor with shorter effective range and uniform scanning pattern, and perform the cross-device adaptation experiments. To further validate the practical value of model adaptation against insufficient annotated data, we progressively remove annotated data from Cirrus to show the model adaptation performance with insufficient annotated data. Note that different from the previous two adaptation experiments, where the network parameters are shared across domains completely, we train domain-specific detection heads for each domain due to the difference on annotation protocol (full-body annotation for KITTI and partially annotation for Cirrus). The results are presented in Table IV. We also present results on training the network using reduced amount of data from Cirrus alone to show the performance improvement with model adaptation.

Cross-pattern Adaptation. In this experiment, we perform model adaptation across the Gaussian and the uniform scanning patterns, so that one common model supports dynamic switching between different scanning patterns. We start with directly feeding a model trained using one scanning pattern with point clouds from the other pattern. As shown in Table V, accuracies drop for point clouds collected with different scanning pattern compared to overall accuracies in Table I, which indicates that the two scanning patterns are inherently different and the model cannot be shared across patterns directly. Based on the aforementioned empirical observations, we perform model adaptation with both paired and unpaired cross-pattern point clouds.

Cross-pattern Adaptation with Paired Data. Since we collect our dataset using two LiDAR sensors with different scanning patters simultaneously, and the coordinates are well-calibrated, we have paired data with consistent annotations. For cross-pattern adaptation with paired data, we directly feed the network with paired point clouds with two patterns and minimize the distance between two features. Feature extractor and detection heads are shared across two patterns. The results are presented in Table VI as joint training.

Cross-pattern Adaptation with Unpaired Data. Paired cross-domain data is expensive to collect. In practice, unpaired data is usually more accessible. In this experiment, we manually shuffle the data collected using Gaussian and uniform scanning patterns, and adopt the adaptation framework to encourage invariant features for both patterns. The results are presented in Table VI as adaptations. Training the network with model adaptation outperforms joint training, indicating the explicit invariant feature imposed by model adaptation improves the generalization of deep networks to different scanning patterns.

VI. Conclusion

In this paper, we introduced Cirrus, a new long-range bi-pattern LiDAR dataset for autonomous driving. The new dataset significantly enriches the diversity of public LiDAR datasets by providing point clouds with 250-meter effective range, as well as Gaussian and uniform scanning patterns. We presented details on the dataset collection and object annotation. 3D object detection in LiDAR is presented as an example task using the Cirrus dataset, and various model adaptation experiments are performed to illustrate important properties and sample usages of this new public dataset.

References

[1] Waleed Ali, Sherif Abdelkarim, Mahmoud Zidan, Mohamed Zahran, and Ahmad El Sallab. Yolo3d: End-to-end real-time 3d oriented object bounding box detection from lidar point cloud. In *European Conference on Computer Vision*, pages 0–0, 2018.

[2] Gabriel J Brostow, Julien Fauqueur, and Roberto Cipolla. Semantic object classes in video: A high-definition ground truth database. *Pattern Recognition Letters*, 30(2):88–97, 2009.

[3] Holger Caesar, Varun Bankiti, Alex H Lang, Sourabh Vora, Venice Erin Liong, Qiang Xu, Anush Krishnan, Yu Pan, Giancarlo Baldan, and Oscar Beijbom. nuScenes: A multimodal dataset for autonomous driving. *arXiv preprint arXiv:1903.11027*, 2019.

[4] Zhengping Che, Guangyu Li, Tracy Li, Bo Jiang, Xuefeng Shi, Xinsheng Zhang, Ying Lu, Guobin Wu, Yan Liu, and Jieping Ye. D ^{2}-city: A large-scale dashcam video dataset of diverse traffic scenarios. *arXiv preprint arXiv:1904.01975*, 2019.

[5] Liang-Chieh Chen, George Papandreou, Florian Schroff, and Hartwig Adam. Rethinking atrous convolution for semantic image segmentation. *arXiv preprint arXiv:1706.05587*, 2017.

[6] Xiaozhi Chen, Huimin Ma, Ji Wan, Bo Li, and Tian Xia. Multi-view 3d object detection network for autonomous driving. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1907–1915, 2017.

[7] Yiping Chen, Jingkang Wang, Jonathan Li, Cewu Lu, Zhipeng Luo, Han Xue, and Cheng Wang. Lidar-video driving dataset: Learning driving policies effectively. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 5870–5878, 2018.

[8] Yukyung Choi, Namil Kim, Soonmin Hwang, Kibaek Park, Jae Shin Yoon, Kyounghwan An, and In So Kweon. KAIST multi-spectral day/night data set for autonomous and assisted driving. *IEEE Transactions on Intelligent Transportation Systems*, 19(3):934–948, 2018.

[9] Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, Markus Enzweiler, Rodrigo Benenson, Uwe Franke, Stefan Roth, and Bernt Schiele. The cityscapes dataset for semantic urban scene understanding. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 3213–3223, 2016.

[10] Alexey Dosovitskiy, German Ros, Felipe Codevilla, Antonio Lopez, and Vladlen Koltun. CARLA: An open urban driving simulator. In *1st Annual Conference on Robot Learning*, pages 1–16, 2017.

[11] A Gaidon, Q Wang, Y Cabon, and E Vig. Virtual worlds as proxy for multi-object tracking analysis. In *IEEE Conference on Computer Cision and Pattern Recognition*, 2016.

[12] Andreas Geiger, Philip Lenz, Christoph Stiller, and Raquel Urtasun. Vision meets robotics: The KITTI dataset. *The International Journal of Robotics Research*, 32(11):1231–1237, 2013.

[13] Benjamin Graham. Spatially-sparse convolutional neural networks. *arXiv preprint arXiv:1409.6070*, 2014.

[14] Ben Graham. Sparse 3d convolutional neural networks. *arXiv preprint arXiv:1505.02890*, 2015.

[15] Benjamin Graham, Martin Engelcke, and Laurens van der Maaten. 3d semantic segmentation with submanifold sparse convolutional networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 9224–9232, 2018.

[16] Benjamin Graham and Laurens van der Maaten. Submanifold sparse convolutional networks. *arXiv preprint arXiv:1706.01307*, 2017.

[17] Xinyu Huang, Xinjing Cheng, Qichuan Geng, Binbin Cao, Dingfu Zhou, Peng Wang, Yuanqing Lin, and Ruigang Yang. The apolloscape dataset for autonomous driving. In *IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 954–960, 2018.

[18] Jeff Laird. PTP background and overview. 2012.

[19] Alex H Lang, Sourabh Vora, Holger Caesar, Lubing Zhou, Jiong Yang, and Oscar Beijbom. PointPillars: Fast encoders for object detection from point clouds. *arXiv preprint arXiv:1812.05784*, 2018.

[20] Bo Li, Tianlei Zhang, and Tian Xia. Vehicle detection from 3d lidar using fully convolutional network. *arXiv preprint arXiv:1608.07916*, 2016.

[21] Baoyuan Liu, Min Wang, Hassan Foroosh, Marshall Tappen, and Marianna Pensky. Sparse convolutional neural networks. In *IEEE conference on computer vision and pattern recognition*, pages 806–814, 2015.

[22] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 3431–3440, 2015.

[23] Gerhard Neuhold, Tobias Ollmann, Samuel Rota Bulo, and Peter Kontschieder. The mapillary vistas dataset for semantic understanding of street scenes. In *IEEE International Conference on Computer Vision*, pages 4990–4999, 2017.

[24] Hyeonwoo Noh, Seunghoon Hong, and Bohyung Han. Learning deconvolution network for semantic segmentation. In *IEEE International Conference on Computer Vision*, pages 1520–1528, 2015.

[25] Abhishek Patil, Srikanth Malla, Haiming Gang, and Yi-Ting Chen. The H3D dataset for full-surround 3d multi-object detection and tracking in crowded urban scenes. *arXiv preprint arXiv:1903.01568*, 2019.

[26] Charles R Qi, Hao Su, Kaichun Mo, and Leonidas J Guibas. Pointnet: Deep learning on point sets for 3d classification and segmentation. In *IEEE conference on computer vision and pattern recognition*, pages 652–660, 2017.

[27] Charles Ruizhongtai Qi, Li Yi, Hao Su, and Leonidas J Guibas. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. In *Advances in neural information processing systems*, pages 5099–5108, 2017.

[28] Stephan R Richter, Vibhav Vineet, Stefan Roth, and Vladlen Koltun. Playing for data: ground truth from computer games. In *European Conference on Computer Vision*, pages 102–118. Springer, 2016.

[29] German Ros, Laura Sellart, Joanna Materzynska, David Vazquez, and Antonio M Lopez. The synthia dataset: A large collection of synthetic images for semantic segmentation of urban scenes. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 3234–3243, 2016.

[30] Shaoshuai Shi, Xiaogang Wang, and Hongsheng Li. Pointrcnn: 3d object progposal generation and detection from point cloud. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 770–779, 2019.

[31] Shaoshuai Shi, Xiaogang Wang, and Hongsheng Li. Pointrcnn: 3d object proposal generation and detection from point cloud. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 770–779, 2019.

[32] Martin Simon, Stefan Milz, Karl Amende, and Horst-Michael Gross. Complex-YOLO: An euler-region-proposal for real-time 3d object detection on point clouds. In *European Conference on Computer Vision*, pages 197–209. Springer, 2018.

[33] Hang Su, Varun Jampani, Deqing Sun, Subhransu Maji, Evangelos Kalogerakis, Ming-Hsuan Yang, and Jan Kautz. Splatnet: Sparse lattice networks for point cloud processing. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 2530–2539, 2018.

[34] Ze Wang, Sihao Ding, Ying Li, Minming Zhao, Sohini Roychowdhury, Andreas Wallin, Guillermo Sapiro, and Qiang Qiu. Range adaptation for 3d object detection in lidar. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV) Workshops*, Oct 2019.

[35] Yan Yan, Yuxing Mao, and Bo Li. SECOND: Sparsely embedded convolutional detection. *Sensors*, 18(10):3337, 2018.

[36] Fisher Yu, Wenqi Xian, Yingying Chen, Fangchen Liu, Mike Liao, Vashisht Madhavan, and Trevor Darrell. BDD100K: A diverse driving video database with scalable annotation tooling. *arXiv preprint arXiv:1805.04687*, 2018.

[37] Yin Zhou and Oncel Tuzel. VoxelNet: End-to-end learning for point cloud based 3d object detection. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 4490–4499, 2018.