Eleventh Hungarian Conference on Computer Graphics and Geometry, Budapest, 2024

Evaluating the Impact of Point Cloud Downsampling on the Robustness of LiDAR-based Object Detection

Marcell Golarits,[12] Zoltan Rozsa,[12] Raouf Hamzaoui,[3] Tanvir Allidina,[3] Xin Lu[3] and Tamas Sziranyi,[12]

[1] Machine Perception Research Laboratory of Hungarian Research Network (HUN-REN)
Institute for Computer Science and Control (SZTAKI), H-1111 Budapest, Kende u. 13-17., Hungary
[2] Faculty of Transportation Engineering and Vehicle Engineering, Budapest University of Technology and Economics (BME KJK),
H-1111 Budapest, Műegyetem rkp. 3., Hungary
[3] De Montfort University (DMU), The Gateway, Leicester, LE1 9BH, United Kingdom

Abstract
LiDAR-based 3D object detection relies on the relatively rich information captured by LiDAR point clouds. However, computational efficiency often requires the downsampling of these point clouds. This paper studies the impact of downsampling strategies on the robustness of a state-of-the-art object detector, namely PointPillars. We compare the performance of the approach under random sampling and farthest point sampling, evaluating the model's accuracy in detecting objects across various downsampling ratios. The experiments were conducted on the popular KITTI dataset.

1. Introduction

Autonomous vehicles and advanced driver-assistance systems (ADAS) rely heavily on robust and accurate 3D object detection for safe navigation. High-resolution LiDAR (Light Detection and Ranging) sensors play a vital role in this task by providing rich spatial information about the environment, although the point resolution is lower compared to, e.g., cameras. PointPillars [1], a popular deep learning framework, has emerged as a leader in LiDAR-based object detection [2] due to its efficiency and accuracy. However, processing raw LiDAR point clouds can be computationally expensive. To address this challenge, one can downsample the point cloud to reduce the number of data points before feeding them into the detection model.

While downsampling may increase processing speed, its impact on detection performance remains a concern. This paper explores the effect of downsampling strategies on the robustness of PointPillars for 3D object detection. We focus on two common downsampling methods: random sampling (RS) and farthest point sampling (FPS). By evaluating the model's accuracy at various downsampling ratios for each technique (Fig. 1), we aim to quantify the trade-off between computational or storage efficiency and detection performance.

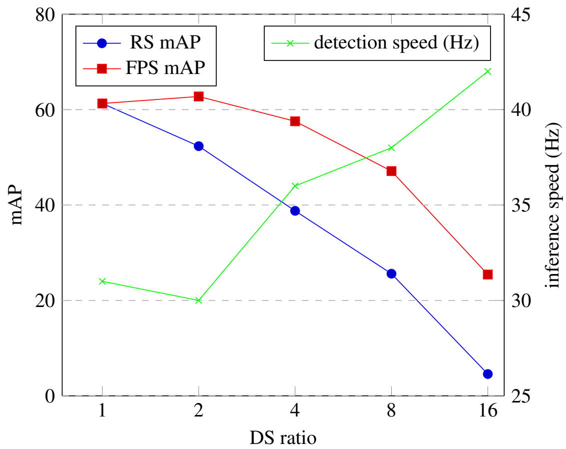


Figure 1: *3D Bounding box mean Average Precision (mAP) and detection speed of PointPillars-based object detection. The results are shown for two popular downsampling techniques (RS and FPS) and four downsampling ratios.*

There are several benefits of building a 3D object detection system that is robust against downsampling:

- Increased processing speed: Downsampling reduces the number of data points that need to be processed, which can significantly improve the speed of object detection. This is especially important for real-time applications, such as autonomous driving, where low latency is critical.
- Reduced memory requirements: Storing and processing large LiDAR point clouds can be memory-intensive. Downsampling can significantly reduce memory requirements, making it possible to run object detection on devices with limited memory resources.
- Improved generalization: Downsampling can help to improve the generalization performance of object detection models as different datasets have different point cloud resolutions and characteristics. For example, a model trained with high-resolution data is unlikely to generalize well to low-resolution ones.
- Enhanced robustness to data loss: Models that are robust against downsampling should also be resilient to data loss resulting from phenomena such as weather conditions (e.g., rain [3]) or targeted attacks [4].

This research offers valuable insights for researchers and developers working on optimizing LiDAR-based object detection for real-world applications in autonomous driving. Understanding the sensitivity of PointPillars to downsampling allows for a more informed approach when balancing the need for speed or generalization with the requirement for accurate object detection in complex environments.

1.1. Contributions

The contributions of the paper are as follows:

- We provide an in-depth analysis of the robustness of LiDAR-based object detection, especially in the case of PointPillars by quantifying the trade-off between efficiency and accuracy. This information can be crucial to practitioners.
- We identify optimal downsampling strategies for PointPillars. By comparing the performance of PointPillars under various downsampling scenarios, we point out which is the most effective at maintaining accuracy while achieving significant computational gains.
- Guiding the development of robust and efficient PointPillars variants. By analyzing our results, we aim to guide the enhancement of PointsPillars to be more robust against data loss.

1.2. Outline of the Paper

The paper is organized as follows: Section 2 surveys the related work. Section 3 describes our experiments' methodology, including downsampling techniques. Section 4 presents our experimental results and evaluates them. Finally, Section 5 draws some conclusions and suggests future work.

2. Related Work

In this section, we first introduce the literature on LiDAR-based object detection and motivate on chosing PointPillars as a representative; then works about performance analysis of object detection - with similar aim as our paper - are discussed.

2.1. LiDAR-based Object Detectors

Three main types of LiDAR-based object detectors can be distinguished today: voxelization-based, point-based and projection-based methods. Voxelization (e.g., [5], [6]) converts the point cloud into a 3D voxel grid, where each voxel represents a small region in space and aggregates the point cloud data within it. Point-based approaches directly operate on the raw LiDAR point cloud, treating each point as a separate entity with spatial coordinates and additional information (e.g., intensity). Voxelization offers the advantage compared to point-based approaches (e.g., [7], [8]) in that the voxelized point clouds can be processed by 3D convolutional neural networks. However, the conversion process can lead to the loss of some details, and it can also be computationally intensive compared to point-based approaches. Projection-based approaches (e.g., [9], [10]) project the LiDAR point cloud onto a 2D image plane from a specific viewpoint (e.g., bird's-eye view). They can use well-established 2D convolutional neural networks for object detection in the projected image domain. In this way, they can be computationally efficient. However, this simplification ignores the available information from the 3rd dimension.

Instead of using 3D convolutions, PointPillars [1] treats the pseudo-bird's-eye view map as a virtual voxelized representation. This allows the entire model to be trained with efficient 2D convolutions. To achieve this, PointPillars uses a simplified PointNet [11] architecture to extract features for each individual point within vertical columns (pillars) of this virtual voxel space.

As the review paper [2] and the current leaderboard of the KITTI dataset state, PointPillars (,which merges the advantages of different methodologies) continus to be of the fastest methods (it enables real-time operation on computers with relatively weak resources). It is also one of the most commonly used object detectors because of its easy and efficient implementation. Therefore, we chose this algorithm to examine 3D object detection robustness against downsampling. On the one hand, as it is so popular in the research community and among practitioners, studying it can be the interest to a large community. On the other hand, having already real-time running capabilities indicates that if the method can be accelerated, then it can be adapted to new sensors with short integration time (high measuring frequency) and low data point number (e.g., LIVOX Avia †) and in small computers like Nvidia Jetson Nano ‡.

2.2. Surveys and Performance Comparisons related to 3D Object Detection

Resolution-agnostic object detection is not a new research topic in autonomous driving [12]. Previously, state-of-the-art 3D object detectors were evaluated in different conditions. E.g., in [13] VoteNet [14], MLCVNet [15], Groupfree [16] and 3DETR [17] were tested in different corruption severity level. In [18] 64-channel LiDAR data was downsampled to 32-channel to test the compatibility of different data in object detection. The same reason (generalization between different datasets and LiDAR point cloud characteristics) guided the research of [19], where different data was used for training. The performance of PointPillars was measured in [20] and [4] against adversarial attacks and common corruptions. The above articles either examine different resolution LiDARs or specific corruption types. The work in [21] downsamples after pillarization, affecting only the consecutive processing steps. To the best of our knowledge, the impact of downsampling on Pointpillars have not been studied.

3. Methodology

This section provides a comprehensive analysis of our methodology for evaluating the robustness of 3D object detection against a decreased number of data points. First, we introduce the downsampling approaches. Next, we describe the dataset used and the implementation details.

3.1. Downsampling Approaches

Nowadays, machine learning-based approaches (e.g., CAS-net [22] or Feat-FPS [23])can be applied to the point cloud simplification problem. However, these are most often either task-specific or learning-based solutions (do not help in generalization). For this reason, we used random sampling [24] and farthest point sampling [25], which are still relevant and are still very popular today [26] [11].

Random sampling [24] is a simple approach that involves randomly selecting a certain number of points from the entire dataset. Each point has an equal chance of being chosen. Its main advantage is the computational efficiency. However, it can lead to unevenly distributed points, especially in the case of sparse datasets (like LiDAR data). Thus, points might cluster in certain areas, leaving other regions unrepresented.

Farthest point sampling [25] is a more strategic approach that aims for a more uniform distribution of the selected points. It works iteratively by repeatedly selecting the point that is farthest away from the already chosen points until the desired number of points is reached. It presents contrasting features compared to random sampling; it is more computationally expensive, but it results in a more uniform distribution.

3.2. Dataset and Experiment Details

Dataset: In our experiments, we used the popular KITTI Vision Benchmark Suite [27], specifically, its 3D object detection dataset. This benchmark consists of 7481 training images, 7518 test images, together with same number of point clouds, totaling about 80000 labeled objects. In our experiments, only the labeled point cloud parts were used in the usual division of 50-50 % split to training and validation data of the original training set. The average precision was calculated to evaluate different type and degree downsampling.

The evaluation was done as the original KITTI evaluation suggests. Thus, detections are considered only in the camera field of view (FoV). For cars at least 70 %, while for pedestrians and cyclists, at least 50 % of 3D bounding box overlap was categorized as successful detection. Three difficulties were defined according to the benchmark proposal (Easy, Moderate and Hard).

Experiment details: In our experiments, we used the pytorch implementation of PointPillars § algorithm and our trained model on the original KITTI object detection dataset. All experiments were performed using a computer equipped with Intel® Core™ i7-7820X CPU @ 3.60GHz × 16 and NVIDIA GeForce GTX 1080 Ti 12GB. The point clouds of the KITTI Object detection dataset were downsampled with ratios of 2, 4, 8 and 16, using both random sampling and farthest point sampling. After the downsampling (as the KITTI dataset is labeled only in the camera FoV), only the data points with positive X values were selected. (This resulted in about only 1000 points as the input of the detection model, in the case of the downsampling with the highest ratio.) Finally, the detection model was evaluated according to the KITTI's proposal (introduced in the previous subsection) for each downsampled validation point cloud.

4. Results and Discussion

The results of our analysis - according to the experiments detailed in Section 3 - are shown in Table 1 and 2. The precision of each category and different difficulty levels and also mean Average Precision (mAP) across all categories are reported in these tables. Besides, running time is provided for both the downsampling methods (DS) and for the detection (Det.).

† https://www.livoxtech.com/avia

‡ https://developer.nvidia.com/embedded/jetson-nano-developer-kit

§ https://github.com/zhulf0804/PointPillars

Ratio	Method	DS (s)	Det. (s)	mAP	Car			Pedestrian			Cyclist		
				Mod.	Easy	Mod.	Hard	Easy	Mod.	Hard	Easy	Mod.	Hard
1			0.032	68.85	90.03	87.72	85.39	60.83	54.87	50.86	79.31	63.98	60.23
2	RS	0.004	0.032	60.08	89.32	79.41	78.10	53.52	48.67	44.78	73.43	52.17	50.08
	FPS	13.74	0.034	69.75	89.98	87.61	85.25	60.53	55.12	51.08	82.24	66.53	62.34
4	RS	0.003	0.028	47.78	87.53	69.72	67.78	39.46	37.38	34.00	52.90	36.23	34.57
	FPS	6.89	0.028	65.68	89.98	87.14	79.50	49.11	46.30	43.18	78.83	63.62	59.00
8	RS	0.002	0.026	33.73	78.01	58.95	55.49	25.54	24.27	22.86	26.45	17.98	17.46
	FPS	3.48	0.026	55.59	88.53	78.86	77.01	39.77	37.46	34.86	68.24	50.46	46.65
16	RS	0.001	0.024	7.65	19.45	13.77	13.04	9.09	9.09	9.09	0.09	0.08	0.08
	FPS	1.80	0.024	37.89	81.18	72.49	65.50	19.78	17.28	16.86	33.66	23.89	22.16

Table 1: *BEV detection results on the validation dataset of KITTI object detection benchmark*

Ratio	Method	DS (s)	Det. (s)	mAP	Car			Pedestrian			Cyclist		
				Mod.	Easy	Mod.	Hard	Easy	Mod.	Hard	Easy	Mod.	Hard
1			0.032	61.29	86.15	76.53	69.10	51.77	46.27	42.75	78.17	61.06	58.73
2	RS	0.004	0.032	52.34	83.09	68.47	65.58	45.49	40.36	36.10	68.59	48.19	45.34
	FPS	13.74	0.034	62.76	86.57	76.60	68.94	53.11	47.46	43.70	79.68	64.21	60.17
4	RS	0.003	0.028	38.78	74.63	56.56	49.12	29.24	27.11	25.12	48.67	32.67	30.92
	FPS	6.89	0.028	57.56	86.38	73.56	67.50	40.91	38.94	35.49	76.39	60.18	56.52
8	RS	0.002	0.026	25.60	56.15	41.77	36.46	18.66	17.94	16.93	24.03	17.10	16.17
	FPS	3.48	0.026	47.12	75.26	64.71	56.88	31.12	29.75	27.50	64.07	46.89	43.82
16	RS	0.001	0.024	4.57	2.98	4.55	4.55	9.09	9.09	9.09	0.07	0.07	0.07
	FPS	1.80	0.024	25.43	48.28	43.52	38.48	11.54	11.90	10.65	28.63	20.86	19.56

Table 2: *3D bounding box detection results on the validation dataset of KITTI object detection benchmark*

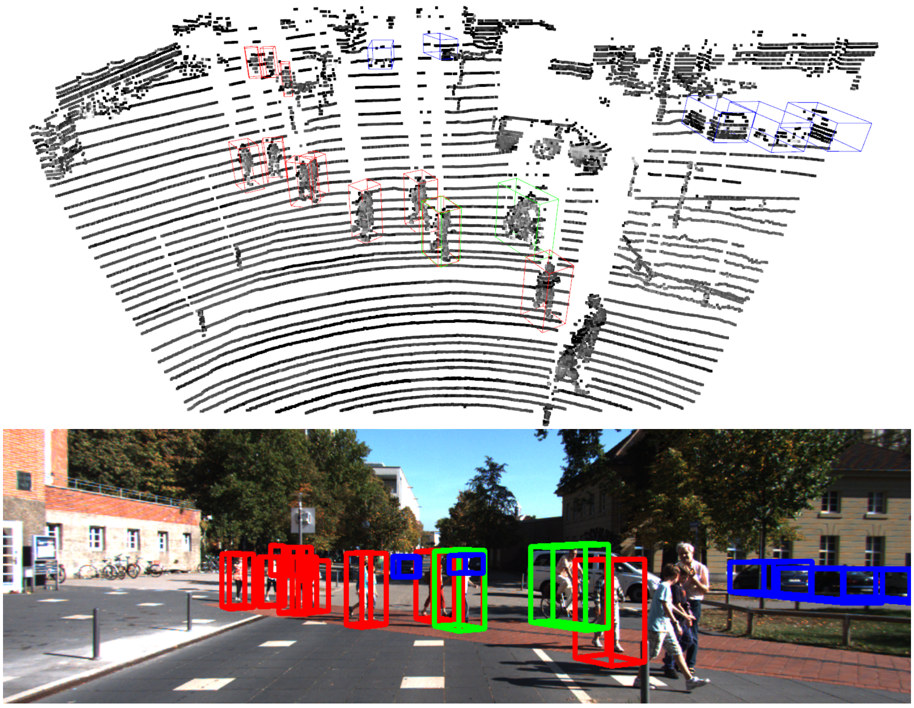


Figure 2: *Example detections on unsampled point cloud*

Table 1 contains Bird's eye view (BEV) bounding box evaluations, while Table 2 shows the results of 3D Bounding Box evaluations.

The tables reveal the following key observations:

1. Downsampling significantly affects precision.
   For both downsampling methods (RS and FPS), precision decreased as the downsampling ratio increased. The precision dropped considerably when the ratio was reached 16. For example, for 3D bounding box detection the mAP value dropped to 4.57 (RS) and 25.43 (FPS) from the 61.29.
2. FPS generally outperformed RS.
   FPS consistently produced higher mAP values compared to RS. The advantage of FPS was more pronounced as the ratio increased (Fig. 1). This is due to FPS generating a more uniform point distribution than RS.
3. Precision can be even increased with downsampling.
   This phenomenon of a slight increase of mAP can be observed in the case of FPS and the downsampling ratio 2. This is beacuse ignoring less important points (points close to each other can be redundant in terms of descriptiveness) can reduce noise.
4. Downsampling, using either RS or FPS, can reduce inference time. As expected, both the downsampling time and the detection time decreased with increasing downsampling ratio. The decrease in inference time from 32 to 24 ms for a sampling ratio of 16, was small but significant but significant. It increased the detection frequency from about 31 Hz to 42 Hz.
5. Total processing time can be decreased with RS.
   Only RS decreased the combined time for downsampling and detection, this means reduction in overall processing time.

In terms of detection accuracy, FPS generally appears to be a better choice for downsampling in PointPillars than RS. However, FPS does not enable real-time processing. For this reason, new downsampling algorithms should be developed to provide both computationally efficient and accurate 3D object detection.

Fig. 3 illustrates downsampled point clouds with different approaches and qualitative results about detections on them from the original point cloud shown in Fig. 2.

Inspecting Figs. 2 and 3 together, one can see that the number of detections significantly decreases to the point of reaching the highest downsampling ratio. This is especially true for the pedestrian class (which has the lowest number of points in general) and less true for the car category (which has the highest number of points in general). Naturally, the distance from the sensor also impacts the detections as the point density of LiDAR point clouds decreases with increasing range. It is also worth noting that the reliability of the classification drops significantly, even in the cases of smaller downsampling ratios. With a downsampling ratio of 2, classes are often confused. For example, pedestrians are frequently categorized as cyclists. This can be explained by the fact that objects (above ground) can be found, but boundaries between neighboring pedestrians are blurred by losing points. Thus, human features in a bigger bounding box can be mixed with cyclists.

5. Conclusions

This paper studied the impact of downsampling strategies on the robustness of PointPillars for 3D object detection. We evaluated the performance of PointPillars under random and farthest point sampling techniques for various downsampling ratios. Our analysis revealed that while FPS may improve detection accuracy compared to random sampling, the associated computational cost renders it impractical for real-time applications. On the other hand, while random downsampling can improve overall runtime performance, its resulting detection accuracy makes it unsuitable.

This work highlights the critical trade-off between accuracy and efficiency in LiDAR-based object detection. While PointPillars exhibits some resilience to downsampling, significant reductions in point cloud density ultimately lead to performance degradation. Future research directions could explore techniques to enhance PointPillars' ability to handle sparse data while maintaining real-time performance. This could involve investigating lightweight network architectures specifically designed for downsampled point clouds or incorporating mechanisms that prioritize informative data points during downsampling.

By optimizing 3D object detection for efficient processing of LiDAR data, we can pave the way for its better generalization and wider adoption in real-world applications.

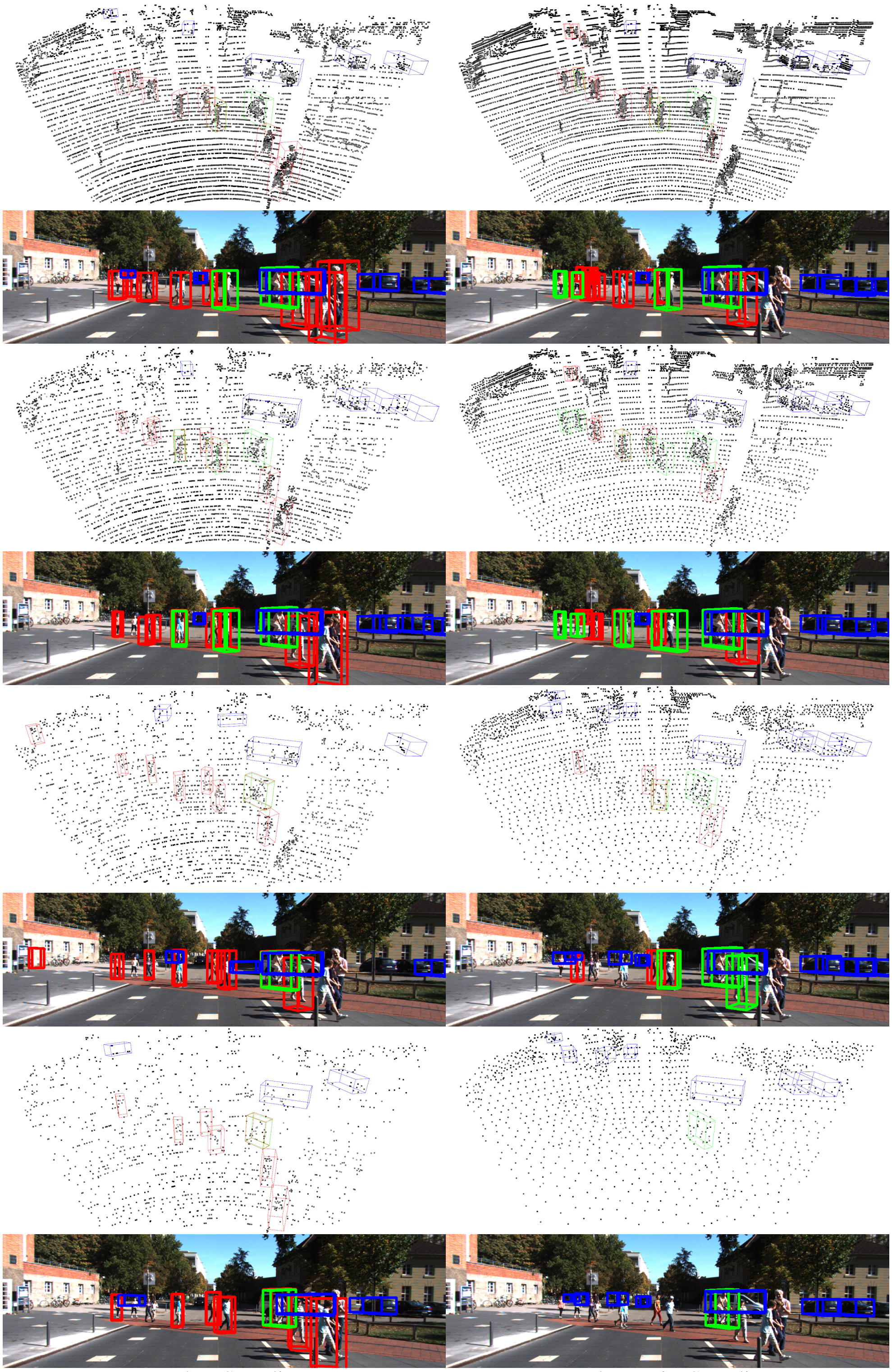
Acknowledgements

This work was supported by the ÚNKP-23-3-I-BME-285 and ÚNKP-23-5-BME-463 New National Excellence Program of the Ministry for Culture and Innovation from the source of the National Research, Development and Innovation Fund. It was also supported by the European Union within the framework of the National Laboratory for Autonomous Systems (RRF-2.3.1-21-2022-00002) and by the Royal Society (IES\R2\232053).

References

1. Alex H. Lang, Sourabh Vora, Holger Caesar, Lubing Zhou, Jiong Yang, and Oscar Beijbom. Pointpillars: Fast encoders for object detection from point clouds. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019.

2. Yutian Wu, Yueyu Wang, Shuwei Zhang, and Harutoshi Ogai. Deep 3d object detection networks using lidar data: A review. *IEEE Sensors Journal*, 21(2):1152–1171, 2021.



Random Sampling Farthest Point Sampling

Figure 3: *Detection with RS and FPS sampled point clouds. The original point cloud is shown of in Fig. 2. The sampling ratios 2, 4, 8, 16 from top to bottom.*

3. Arsalan Haider, Marcell Pigniczki, Shotaro Koyama, Michael H. Köhler, Lukas Haas, Maximilian Fink, Michael Schardt, Koji Nagase, Thomas Zeh, Abdulkadir Eryildirim, Tim Poguntke, Hideo Inoue, Martin Jakobi, and Alexander W. Koch. A methodology to model the rain and fog effect on the performance of automotive lidar sensors. *Sensors*, 23(15), 2023.

4. Yifan Zhang, Junhui Hou, and Yixuan Yuan. A comprehensive study of the robustness for lidar-based 3d object detectors against adversarial attacks. *International Journal of Computer Vision*, pages 1–33, 11 2023.

5. Yilun Chen, Shu Liu, Xiaoyong Shen, and Jiaya Jia. Fast point r-cnn. *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 9774–9783, 2019.

6. Jiajun Deng, Shaoshuai Shi, Peiwei Li, Wengang Zhou, Yanyong Zhang, and Houqiang Li. Voxel r-cnn: Towards high performance voxel-based 3d object detection. *Proceedings of the AAAI Conference on Artificial Intelligence*, 35:1201–1209, 05 2021.

7. Kiwoo Shin, Youngwook Paul Kwon, and Masayoshi Tomizuka. Roarnet: A robust 3d object detection based on region approximation refinement. *2019 IEEE Intelligent Vehicles Symposium (IV)*, pages 2510–2515, 2018.

8. Shaoshuai Shi, Xiaogang Wang, and Hongsheng Li. Pointrcnn: 3d object proposal generation and detection from point cloud. pages 770–779, 06 2019.

9. Charles R Qi, Wei Liu, Chenxia Wu, Hao Su, and Leonidas J Guibas. Frustum pointnets for 3d object detection from rgb-d data. *arXiv preprint arXiv:1711.08488*, 2017.

10. Xiaozhi Chen, Huimin Ma, Ji Wan, Bo Li, and Tian Xia. Multi-view 3d object detection network for autonomous driving. *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6526–6534, 2016.

11. Charles Ruizhongtai Qi, Li Yi, Hao Su, and Leonidas J Guibas. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. In I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017.

12. Ruddy Théodose, Dieumet Denis, Thierry Chateau, Vincent Fremont, and Paul Checchin. A deep learning approach for lidar resolution-agnostic object detection. *IEEE Transactions on Intelligent Transportation Systems*, 23(9):14582–14593, 2022.

13. Fatima Albreiki, Sultan Abu Ghazal, Jean Lahoud, Rao Anwer, Hisham Cholakkal, and Fahad Khan. On the robustness of 3d object detectors. In *Proceedings of the 4th ACM International Conference on Multimedia in Asia*, MMAsia '22, New York, NY, USA, 2022. Association for Computing Machinery.

14. Zhipeng Ding, Xu Han, and Marc Niethammer. Votenet: A deep learning label fusion method for multi-atlas segmentation. In *Medical Image Computing and Computer Assisted Intervention – MICCAI 2019: 22nd International Conference, Shenzhen, China, October 13–17, 2019, Proceedings, Part III*, page 202–210, Berlin, Heidelberg, 2019. Springer-Verlag.

15. Qian Xie, Yu-Kun Lai, Jing Wu, Zhoutao Wang, Yiming Zhang, Kai Xu, and Jun Wang. Mlcvnet: Multi-level context votenet for 3d object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020.

16. Ze Liu, Zheng Zhang, Yue Cao, Han Hu, and Xin Tong. Group-free 3d object detection via transformers. *arXiv preprint arXiv:2104.00678*, 2021.

17. Ishan Misra, Rohit Girdhar, and Armand Joulin. An End-to-End Transformer Model for 3D Object Detection. In *ICCV*, 2021.

18. Jihwan You and Young-Keun Kim. Up-sampling method for low-resolution lidar point cloud to enhance 3d object detection in an autonomous driving environment. *Sensors*, 23(1), 2023.

19. Louis Soum-Fontez, Jean-Emmanuel Deschaud, and François Goulette. Mdt3d: Multi-dataset training for lidar 3d object detection generalization. pages 5765–5772, 10 2023.

20. Yinpeng Dong, Caixin Kang, Jinlai Zhang, Zijian Zhu, Yikai Wang, Xiao Yang, Hang Su, Xingxing Wei, and Zhu Jun. Benchmarking robustness of 3d object detection to common corruptions in autonomous driving. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023.

21. Xunli Zhong, Qiang Li, Enfu Ye, Lei Huang, and Weize Sun. Swfnet: Efficient sampling and weighted fusion for enhanced pointpillars in 3d object detection. *2023 6th International Conference on Information Communication and Signal Processing (ICICSP)*, pages 1018–1022, 2023.

22. Chen Chen, Hui Yuan, Hao Liu, Junhui Hou, and Raouf Hamzaoui. Cas-net: Cascade attention-based sampling neural network for point cloud simplification. In *2023 IEEE International Conference on Multimedia and Expo (ICME)*, pages 1991–1996, 2023.

23. Zetong Yang, Yanan Sun, Shu Liu, and Jiaya Jia. 3dssd: Point-based 3d single stage object detector. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 11037–11045, 2020.

24. François Pomerleau, Francis Colas, Roland Siegwart,

and Stéphane Magnenat. Comparing icp variants on real-world data sets. *Autonomous Robots*, 04 2013.

25. Yuval Eldar, Michael Lindenbaum, Moshe Porat, and Yehoshua Zeevi. The farthest point strategy for progressive image sampling. *IEEE transactions on image processing : a publication of the IEEE Signal Processing Society*, 6:1305–15, 02 1997.

26. Qingyong Hu, Bo Yang, Linhai Xie, Stefano Rosa, Yulan Guo, Zhihua Wang, Niki Trigoni, and Andrew Markham. Randla-net: Efficient semantic segmentation of large-scale point clouds. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2020.

27. A. Geiger, P. Lenz, and R. Urtasun. Are we ready for Autonomous Driving? The KITTI Vision Benchmark Suite. In *Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, pages 3354–3361, 2012.