

# Automatic Generation of Synthetic LiDAR Point Clouds for 3-D Data Analysis

Fei Wang, Yan Zhuang<sup>id</sup>, *Member, IEEE*, Hong Gu<sup>id</sup>, and Huosheng Hu<sup>id</sup>, *Senior Member, IEEE*

**Abstract**—The recent success of deep learning in 3-D data analysis relies upon the availability of large annotated data sets. However, creating 3-D data sets with point-level labels are extremely challenging and require a huge amount of human efforts. This paper presents a novel open-sourced method to extract light detection and ranging point clouds with ground truth annotations from a simulator automatically. The virtual sensor can be configured to simulate various real devices, from 2-D laser scanners to 3-D real-time sensors. Experiments are conducted to show that using additional synthetic data for training can: 1) achieve a visible performance boost in accuracy; 2) reduce the amount of manually labeled real-world data; and 3) help to improve the generalization performance across data sets.

**Index Terms**—Deep learning, semantic segmentation, synthetic light detection and ranging (LiDAR) point clouds.

## I. INTRODUCTION

DEEP learning technology has become the research focus of various light detection and ranging (LiDAR)-based perception tasks, such as recognition [1], semantic segmentation [2], and scene understanding [3]. These methods often require a large amount of labeled training data which means a huge amount of human efforts. Tracing accurate object boundaries in 3-D is very difficult. This is particularly true for data sets acquired with real-time LiDAR sensors. As a result, there are very limited LiDAR point cloud data sets available. To solve this problem, one possible way is to use simulators to automatically generate annotated LiDAR point clouds.

The idea of using synthetic data starts from computer vision communities. Different methods have been presented to extract synthetic images from simulators [4], [5]. In terms of LiDAR simulation, Yue *et al.* [6] presented a framework to generate synthetic point clouds from a video game for road objects segmentation. They observed a +9% improvement in accuracy when augmenting training data with the synthetic data. In [7], more realistic data were generated by using point clouds collected from the real world as a static background. But it

required human interactions to generate dynamic traffic scenes. In [9], synthetic data were used to train a deep model for segmentation of road objects in urban environments.

To the best of our knowledge, limited works have been done on automatic generation of simulated LiDAR point clouds with point-level labels for 3-D data analysis. The main contributions of this paper are as follows.

- 1) An open-sourced method to automatically generate 3-D annotated LiDAR point clouds with highly configurable parameters to simulate various real devices. The code is available at <https://github.com/ZhuangYanDLUT/carla>.
- 2) Showing by experiments that using synthetic data can: 1) achieve a visible performance boost of a deep model; 2) significantly reduce the amount of manually labeled data for training; and 3) help to avoid the data set bias problem when real data contains limited types of scenes.

## II. GENERATION OF SYNTHETIC 3-D DATA

Our synthetic data generation method is based on CARLA [9], a simulator with various digital assets (such as urban layouts, buildings, and vehicles). The virtual LiDAR sensor uses ray casting to simulate a laser ray. The ray casting API, provided by the engine, takes the sensor location and the 3-D coordinates of the ending point of a ray as inputs and returns descriptions about the first point it hits. If a ray does not hit anything, it is filtered out in the outputs. Three-Dimensional coordinates of all hit points are then transformed from the world frame to the sensor frame and sent to the client.

We use the returned descriptions of ray casting to extract point-level labels. The descriptions include a pointer to an actor/component that the hit point belongs to. The actor is the base class for an object that can be placed in the simulation. An actor may have several components to control its behaviors or rendering processes. Both actors and components contain an array of tags. These tags are generated by the simulator during rendering. They are divided into 13 categories, e.g., buildings, fences, and others. Thus, we extract these tags as semantic labels for the hit points.

The virtual sensors allow for flexible configurations to simulate various types of real devices. Typical configurable parameters are shown in Fig. 1. Users can set these parameters to generate synthetic data of different sensors and settings, such as two vertically side-faced 2-D scanners (used in Oakland data set), a Velodyne HDL-64 laser scanner (used in KITTI data set), or a high-resolution static laser scanner (used in Semantic3d.net data set). Fig. 2 shows the examples of the synthetic data generated from these settings.

Manuscript received February 7, 2019; accepted March 04, 2019. Date of publication April 15, 2019; date of current version June 7, 2019. This work was supported by the National Natural Science Foundation of China under Grant 61375088 and Grant U1608253. The Associate Editor coordinating the review process was Dr. Christoph Baer. (*Corresponding author: Yan Zhuang.*)

F. Wang and Y. Zhuang are with the School of Control Science and Engineering, Dalian University of Technology, Dalian 116024, China (e-mail: feiwang@mail.dlut.edu.cn; zhuang@dlut.edu.cn).

H. Gu is with the Faculty of Electronic Information and Electrical Engineering, Dalian University of Technology, Dalian 116024, China (e-mail: guhong@dlut.edu.cn).

H. Hu is with the School of Computer Science and Electronic Engineering, University of Essex, Colchester CO4 3SQ, U.K. (e-mail: hhu@essex.ac.uk).

Color versions of one or more of the figures in this paper are available online at <http://ieeexplore.ieee.org>.

Digital Object Identifier 10.1109/TIM.2019.2906416

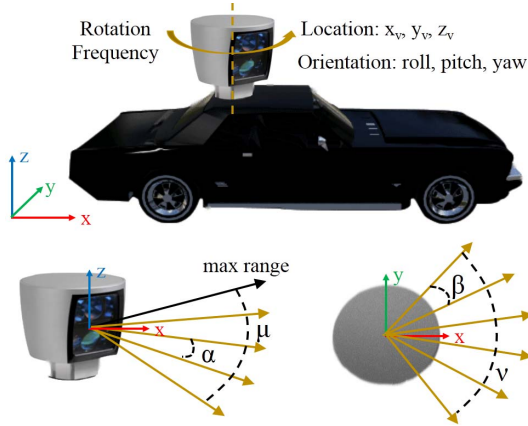


Fig. 1. Configurable parameters of the virtual sensor.  $\mu$  and  $\nu$  denote vertical and horizontal FOV;  $\alpha$  and  $\beta$  are vertical and horizontal resolutions.

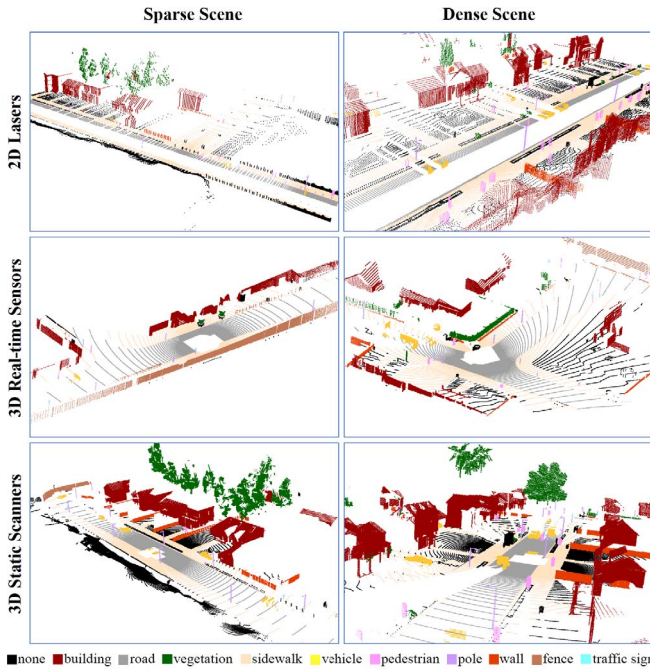


Fig. 2. Simulated LiDAR point clouds of different sensors from different scenes.

Besides, the scenes are also configurable. Users can specify the car model, number of cars, number of pedestrians, and weather and time of day in the simulation. The virtual world map can also be modified or customly built if it is needed.

Our method is fully automatic and no human interaction is needed during data collection. Once the parameters are configured, a vehicle equipped with sensors will drive around autonomously. If it crashes into other objects, the simulation will restart within a few seconds. Measurements are saved automatically when the vehicle has moved a certain distance from the last saved point.

### III. EXPERIMENTAL RESULTS

In our experiments, we use the same sensor configurations as those used in KITTI data set to collect a synthetic data set. The simulations run three times with a different number of dynamic objects to make the data more diverse. Totally

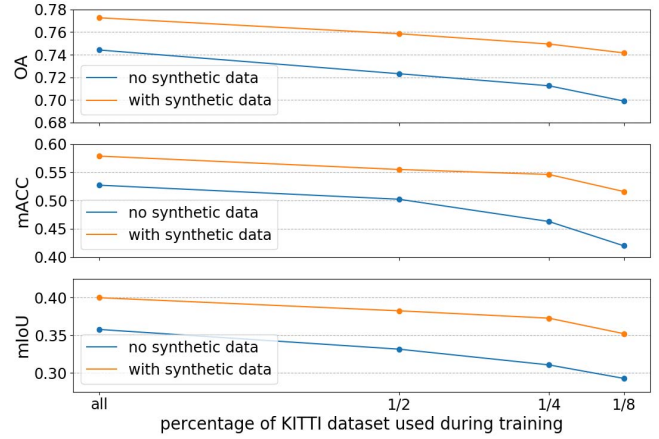


Fig. 3. Comparisons of training with and without synthetic data when using different amounts of real-world data (140 frames in total).

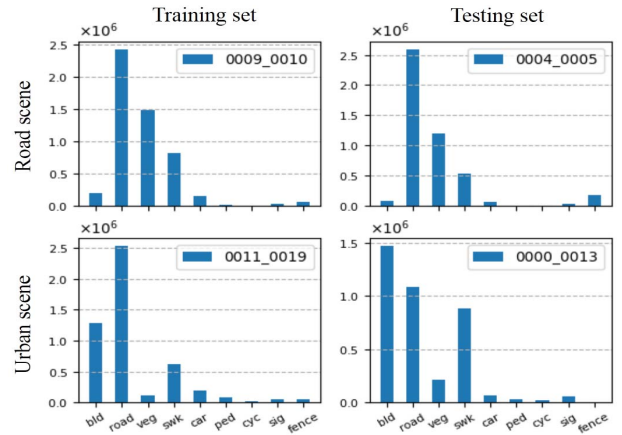


Fig. 4. Label distribution of subsets of the KITTI data set. “0009\_0010” denotes data from drive sequence 0009 and 0010. The entire data set is separated into two types of natural scenes—road and urban.

TABLE I  
COMPARISONS OF TRAINING WITHOUT/WITH SYNTHETIC DATA ON THE SUBSETS

Train \ Test	Road Scenes			Urban Scenes		
	mACC	mIoU	OA	mACC	mIoU	OA
Road Scenes	51 / 52	29 / 29	75 / 76	32 / 34	18 / 19	54 / 57
Urban Scenes	44 / 46	17 / 18	60 / 62	39 / 45	24 / 29	64 / 69










1703 frames are stored. It takes about 15 h to generate the entire synthetic data set.

The usage of synthetic data is mainly tested on semantic segmentation. A deep model, similar to [2], is jointly trained on the synthetic and real-world data set. The performances are evaluated in terms of overall accuracy (OA), mean of per class accuracy (mACC), and mean of per class intersection over union (mIoU).

#### A. Limited Real Data

In this experiment, we use different amounts of real data to train a deep model. The results are shown in Fig. 3. For models trained only on real data, significant performance degradation is observed when reducing the size of the data set. If synthetic

TABLE II  
QUANTITATIVE RESULTS OF DIFFERENT APPROACHES ON THE KITTI DATA SET

Method	 building	 road	 vegetation	 sidewalk	 car	 pedestrian	 cyclist	 signage	 fence	mIoU	mACC
Zhang et al.[10]	<b>86.90</b>	<b>89.20</b>	55.00	26.20	50.00	49.00	<b>19.30</b>	<b>51.7</b>	<b>21.10</b>	-	49.80
3D-FCNN-TI [2]	85.83	<b>90.57</b>	70.50	25.56	65.68	46.35	<b>7.78</b>	28.40	4.51	35.65	47.24
SEGCloud [2]	85.86	88.84	68.73	29.74	67.51	<b>53.52</b>	7.27	39.62	4.05	<b>36.78</b>	49.46
Ours-All_K	76.73	87.43	<b>75.33</b>	46.74	<b>68.55</b>	39.98	5.45	<b>55.77</b>	18.53	35.75	<b>52.72</b>
Ours-1/8_K-S	79.67	86.42	73.40	<b>52.23</b>	66.09	41.84	4.28	43.99	15.20	35.19	51.46
Ours-All_K-S	<b>86.56</b>	82.06	<b>75.66</b>	<b>66.73</b>	<b>72.42</b>	<b>68.65</b>	2.57	44.97	<b>21.21</b>	<b>39.97</b>	<b>57.87</b>

\* Ours-All\_K, ours-1/8\_K-S and ours-All\_K-S denote our model trained on KITTI dataset, 1/8 subset of KITTI dataset (17 frames) and the synthetic dataset, and the entire KITTI dataset and the synthetic dataset, respectively. Top two performances are shown in bold.

data is used, this degradation would be slowed down. One interesting thing is that the model jointly trained on synthetic data and the 1/8 subset of KITTI data set shows a performance very close to the model trained on the entire KITTI data set. This indicates that it is possible to train a model using a limited number of manually labeled data without the loss of accuracy. Thus, training with synthetic data could significantly reduce the required amount of manually labeled real-world data.

### B. Limited Scene Types

In this experiment, we investigate the impact of training with synthetic data on the generalization performance of a deep model across different data sets. To this end, training and testing set of KITTI data set are separated into two subsets according to their label distributions, as shown in Fig. 4. Note that it is much challenging to train a deep model on these subsets because not only the scene type is limited but also the number of training examples is reduced.

Table I reports the performance of training with and without synthetic data on the subsets. The model trained on road scenes shows a significant degradation on urban scenes as not enough buildings and dynamic objects have been seen from the training set; while the model trained on urban scenes generalizes well on road scenes. For both cases, we observed a performance boost when synthetic data was used during training. Besides, models tested on the same scene type as the training set also benefit from the use of synthetic data.

### C. Comparison With State-of-the-Art Methods

We choose three state-of-the-art methods for comparison. The first one [10] is a feature-based approach of which a random forest classifier is used. The second one, 3D-FCNN-TI [2], is an end-to-end deep-learning-based framework. The third one, SEGCloud, is the combination of 3D-FCNN-TI and a CRF. No synthetic data are used in these approaches.

Table II presents a quantitative comparison among different approaches. Our model trained only on the KITTI data set is also presented as a baseline for comparison. When jointly trained on the synthetic and real-world data sets, our model shows a +4% mIoU and +5% mACC performance boost. Besides, with 1/8 of real data, our model can achieve similar

performance to the other approaches trained on the entire data set if synthetic data are used during training.

## IV. CONCLUSION

In this paper, we presented a new method to automatically extract 3-D LiDAR point clouds with point-level ground truth labels from an autonomous driving simulator for 3-D data analysis. Comprehensive evaluation results demonstrate that: 1) training with synthetic data can improve the performance of a deep model and this improvement becomes more prominent as the real-world data set becomes smaller; 2) using synthetic data can significantly reduce the amount of manually labeled real data that required to train a deep model and thus release human efforts to create of 3-D data sets; and 3) using synthetic data also helps to improve generalization performance when real data contains limited types of scenes.

## REFERENCES

- [1] C. Ma, Y. Guo, Y. Lei, and W. An, "Binary volumetric convolutional neural networks for 3-D object recognition," *IEEE Trans. Instrum. Meas.*, vol. 68, no. 1, pp. 38–48, Jan. 2019.
- [2] L. Tchapmi, C. Choy, I. Armeni, J. Gwak, and S. Savarese, "SEGCloud: Semantic segmentation of 3D point clouds," in *Proc. Int. Conf. 3D Vis.*, Qingdao, China, Oct. 2017, pp. 537–547.
- [3] Z. Qiu, Y. Zhuang, F. Yan, H. Hu, and W. Wang, "RGB-DI images and full convolution neural network-based outdoor scene understanding for mobile robots," *IEEE Trans. Instrum. Meas.*, vol. 68, no. 1, pp. 27–37, Jan. 2019.
- [4] S. R. Richter, V. Vineet, S. Roth, and V. Koltun, "Playing for data: Ground truth from computer games," in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, Sep. 2016, pp. 102–118.
- [5] G. Ros, L. Sellart, J. Materzynska, D. Vazquez, and A. M. Lopez, "The SYNTHIA dataset: A large collection of synthetic images for semantic segmentation of urban scenes," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Las Vegas, NV, USA, Jun. 2016, pp. 3234–3243.
- [6] X. Yue, B. Wu, S. A. Seshia, K. Keutzer, and A. L. Sangiovanni-Vincentelli. (Mar. 2018). "A LiDAR point cloud generator: From a virtual world to autonomous driving." [Online]. Available: <https://arxiv.org/abs/1804.00103>
- [7] J. Fang *et al.*. (Nov. 2018). "Simulating LIDAR point cloud for autonomous driving using real-world scenes and traffic flows." [Online]. Available: <https://arxiv.org/abs/1811.07112>
- [8] B. Wu, X. Zhou, S. Zhao, X. Yue, and K. Keutzer. (Sep. 2018). "SqueezeSegV2: Improved model structure and unsupervised domain adaptation for road-object segmentation from a LIDAR point cloud." [Online]. Available: <https://arxiv.org/abs/1809.08495>
- [9] A. Dosovitskiy, G. Ros, F. Codevilla, A. Lopez, and V. Koltun, "CARLA: An open urban driving simulator," in *Proc. Conf. Robot Learn.*, Mountain View, CA, USA, Nov. 2017, p. 1–16.
- [10] R. Zhang, S. A. Candra, K. Vetter, and A. Zakhor, "Sensor fusion for semantic segmentation of urban scenes," in *Proc. IEEE Int. Conf. Robot. Automat.*, Seattle, WA, USA, May 2015, pp. 1850–1857.