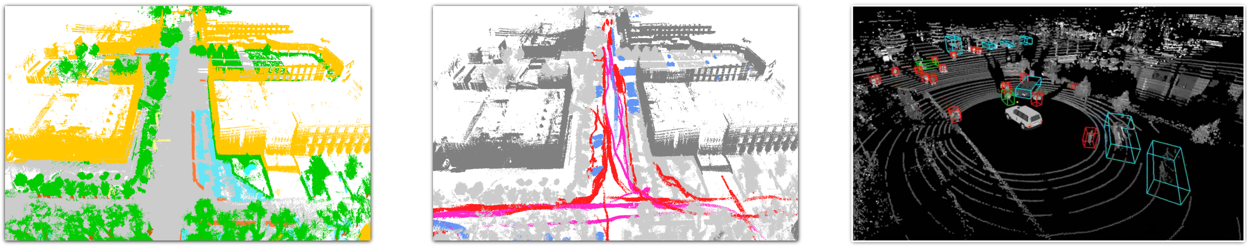
2020 IEEE Intelligent Vehicles Symposium (IV)
October 20-23, 2020. Las Vegas, USA

SemanticPOSS: A Point Cloud Dataset with Large Quantity of Dynamic Instances

Yancheng Pan*, Biao Gao*, Jilin Mei*, Sibo Geng*, Chengkun Li†, Huijing Zhao*
*Peking University, Beijing, China
†Beijing Institute of Technology, Beijing, China



(a) (b) (c)

Fig. 1: A typical scene of SemanticPOSS. (a) LiDAR points of only the static objects are visualized. (b) LiDAR points of the dynamic objects are highlighted. (c) Instances of the dynamic objects in a single LiDAR scan.

***Abstract*— 3D semantic segmentation is one of the key tasks for autonomous driving system. Recently, deep learning models for 3D semantic segmentation task have been widely researched, but they usually require large amounts of training data. However, the present datasets for 3D semantic segmentation are lack of point-wise annotation, diversiform scenes and dynamic objects.**

In this paper[1], we propose the SemanticPOSS dataset, which contains 2988 various and complicated LiDAR scans with large quantity of dynamic instances. The data is collected in Peking University and uses the same data format as SemanticKITTI. In addition, we evaluate several typical 3D semantic segmentation models on our SemanticPOSS dataset. Experimental results show that SemanticPOSS can help to improve the prediction accuracy of dynamic objects as people, car in some degree. SemanticPOSS will be published at `www.poss.pku.edu.cn`.

I. Introduction

Scene understanding is a vital part of autonomous driving system. In order to distinguish different objects in the environment, autonomous vehicles use sensors such as camera, LiDAR to perceive, and then use a semantic segmentation algorithm to learn how objects distribute in the environment. Compared to camera, LiDAR gives 3D information and it is not easily influenced by various light condition. Therefore, LiDAR has become an essential sensor in most autonomous driving system.

Semantic segmentation for 3D LiDAR point cloud has been widely researched in the past several years. Recently, with the development of deep learning methods, a lot of deep neural network models have been proposed and brought a great progress in 3D semantic segmentation. Some models use 3D point cloud as input directly and give every point a semantic label, such as PointNet[1], PointNet++[2], SPLATNet[3]. Others project 3D point cloud to a surface to generate a 2D range image as input, and then give every pixel a semantic label, such as PointSeg[4], SqueezeSeg[5].

However, the generalization performance of deep learning models depends on a large amount and diversity of manually labeled data. For 3D semantic segmentation, we need a large-scale point cloud dataset with point-wise annotation. Although we can find several relevant public datasets such as Semantic3D[6], SemanticKITTI[7], it is still hard to satisfy the data requirement of deep learning models. In this paper, we try to discuss three problems existing in public datasets for 3D semantic segmentation:

- Lack point-wise labeled data. Compared to the image datasets such as PASCAL VOC[8], ImageNet[9] and Cityscapes[10], the scale of point-wise labeled point cloud dataset is limited. As shown in Table.I, ImageNet contains 14197122 image samples and Cityscapes contains 52425M labeled pixels, much more than any existing 3D dataset. The difficulty of manually annotating 3D point cloud is a nonnegligible reason for the lack of 3D datasets.
- Lack scene diversity. A single 3D dataset usually have scene selection bias[11] in some degree. For example, most 3D datasets only contain scenes of structured urban road or highway environment. The lack of scene diversity will limit the generalization performance of deep learning models. If a model is trained at a dataset with low scene diversity, its performance will drop drastically when testing at new different scenes.
- Lack dynamic objects. For autonomous driving system, we are extremely concerned about pedestrians, cars in the surrounding. However, most existing 3D datasets contain rich static objects but few dynamic objects,

[1]This work is supported by the National Key Research and Development Program of China (2017YFB1002601) and the NSFC Grants (61973004). Correspondence: H. Zhao, `zhaohj@cis.pku.edu.cn`.

Authorized licensed use limited to: ShanghaiTech University. Downloaded on April 19,2025 at 07:50:21 UTC from IEEE Xplore. Restrictions apply.

	frames	points	classes	type	annotation	scene	average instances per frame		
							people	rider	car
PASCAL VOC[8]	9993	/	20	images	point-wise	/	/	/	/
ImageNet[9]	14,197,122	/	21841	images	bounding box	/	/	/	/
Cityscapes[10]	24998	52425M	30	images (sequential)	point-wise / instance	outdoor	6.16	0.68	9.51
NYU-Depth V2[12]	1449	445M	894	RGB-D	point-wise	indoor	/	/	/
SUN RGB-D[13]	10335	3175M	800	RGB-D	bounding box	indoor	/	/	/
ScanNet[14]	2,500,000	768000M	20	RGB-D	point-wise	indoor	/	/	/
ApolloScape[15]	146997	1322973M	25	RGB-D*	point-wise	outdoor	/	/	/
S3DIS[16]	5	215M	12	static point clouds	point-wise	indoor	/	/	/
Oakland[17]	17	1.6M	44	static point clouds	point-wise	outdoor	/	/	/
Paris-lille-3D[18]	3	143M	50	static point clouds	point-wise	outdoor	/	/	/
Semantic3D[6]	30	4009M	8	static point clouds	point-wise	outdoor	/	/	/
KITTI[19]	14999	1799M	8	sequential point clouds	bounding box	outdoor	0.63	0.22	4.38
nuScenes[20]	40000	2780M	23	sequential point clouds	bounding box	outdoor	5.45	0.62	**15.89**
SemanticKITTI[7]	43552	4549M	28	sequential point clouds	point-wise / instance	outdoor	0.63	0.18	10.09
SemanticPOSS	2988	216M	14	sequential point clouds	point-wise / instance	outdoor	**8.29**	**2.57**	15.02
GTA-V[21]	/	/	/	synthetic point clouds	point-wise	outdoor	/	/	/
SynthCity[22]	/	367.9M	9	synthetic point clouds	point-wise	outdoor	/	/	/

TABLE I: The existing main image datasets and 3D datasets.

which may make deep learning models detect and recognize the important dynamic instances not as well as other static objects. The quantities of dynamic instances of several datasets are also shown in Table.I.

In order to alleviate the problems mentioned above, we propose the SemanticPOSS dataset, which contains 2988 LiDAR scans with point-wise labeling and 14 classes. Meanwhile, we also give instance-level annotations. The data is collected in Peking University, China. Distinct from typical urban road scenes as most datasets, the campus scenes of SemanticPOSS are various with a large number of pedestrians, riders, cars, and so on. We select an example shown in Figure.1.

For convenience, the provided dataset follows the same data format and interface as SemanticKITTI. It almost does not need extra data processing for the model training on SemanticKITTI to use SemanticPOSS. We also evaluate some typical 3D semantic segmentation models on our SemanticPOSS dataset.

II. Related Works

The increasement and improvement of datasets tremendously contribute to the developing of deep learning models. 3D point cloud datasets also help the research of many tasks such as 3D classification, detection and semantic segmentation. Due to the difference of data collection approaches, the existing 3D point cloud datasets can be divided into 4 types: RGB-D, static point clouds, sequential point clouds, synthetic point clouds. Different type of datasets have different density and features of the point clouds.

A. RGB-D Datasets

RGB-D datasets are collected by RGB-D camera like Kinect. The RGB-D camera not only gets RGB images like common camera, but also perceive the depth for every pixel, to build 3D space information of the scenes. Because it is hard for RGB-D camera to get high-quality data in outdoor scenes, RGB-D datasets only have indoor scenes generally. NYU-Depth V2[12] contains 1449 RGB-D images from 364 different indoor scenes with pixel-level labels. SUN RGB-D[13] is similar to NYU-Depth V2, which includes 10335 RGB-D images with 64595 3D bounding box annotations. ScanNet[14] is much more large, which contains 2.5M RGB-D images, 1513 sequences shot in 707 different indoor scenes. ApolloScape[15] is a huge dataset for autonomous driving application, but for semantic segmentation task it uses the data format like RGB-D.

B. Static Point Clouds Datasets

Static point clouds are some discrete, independent point clouds without dynamic objects, which are collected by laser scanner in two ways. One can put a fixed laser scanner at a specific position to get a quite dense point cloud, or equip a mobile side looking laser scanner and a precise GPS/IMU localization system to generate a dense point cloud covering large areas. This type of datasets usually contains many points but few frames and no dynamic objects. Laser scanners are suitable for both indoor environment and outdoor environment. Stanford Large-Scale 3D Indoor Spaces (S3DIS)[16] is composed of 215M points from 5 indoor areas. Oakland[17] is an outdoor datasets with 1.6M labeled points. Paris-lille-3D[18] consists of 143M points from Paris and Lille. Semantic3D[6] contains 4009M points and different scenes captured in Central Europe.

C. Sequential Point Clouds Datasets

Sequential point clouds are collected by moving LiDAR. This type of datasets contains a great quantity of points and frames, but the point cloud of a frame is sparse. It is very hard to annotate so many point clouds, so some use 3D bounding boxes to label objects to lighten the workload, such as KITTI[19] and nuScenes[20]. However, 3D bounding boxes can not be used for 3D semantic segmentation task directly. Recently, SemanticKITTI[7] provides 4549M labeled points in 22 sequences of the KITTI Vision Odometry Benchmark.

Authorized licensed use limited to: ShanghaiTech University. Downloaded on April 19,2025 at 07:50:21 UTC from IEEE Xplore. Restrictions apply.

D. Synthetic Datasets

Because of the difficulty of the manual point clouds annotation, synthetic datasets provide an approach to rapidly generate considerable number of point clouds with accurate point-wise labels. GTA-V[21] is a popular video game with a virtual world, whose environment is similar to our real world. One can easily create virtual point clouds in GTA-V by some API. SynthCity[22] is a synthetic point cloud that contains 367.9M points with Gaussian noise. However, there is still a gap between synthetic data and realistic scenes, so synthetic datasets can not completely replace real datasets.

Obviously, it is important to choose proper datasets for different tasks. For autonomous driving system, we are more concerned about sequential point clouds. In addition, we pay more attention to the dynamic instances like pedestrians in the environment. Our SemanticPOSS dataset increases the scale of existing sequential point clouds data with point-wise labels. Besides, the scenes of SemanticPOSS is more complicated with considerable quantity of dynamic instances.

III. Dataset Construction

A. Sensors and Data Collection

We use a vehicle equipped with a Pandora[2] sensor module and a GPS/IMU localization system to collect point clouds data. The Pandora integrates cameras, LiDAR and data processing ability into the same module, together with advanced synchronization and calibration features. It consists of a 40-channel LiDAR with 0.33 degree vertical resolution, a forward-facing color camera, 4 wide-angle mono cameras covering 360 degrees around the car. In addition, the GPS/IMU is used for providing localization information to help manual annotation. More details are shown in Figure.2 and Table.II.

[2]For more details, please refer to www.hesaitech.com.

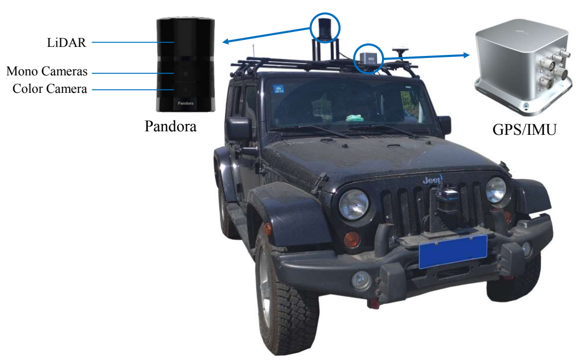


Fig. 2: Our data collection vehicle and sensors.

All of the data was collected in Peking University, China. It took about 5 minutes for the data collection vehicle to acquire point clouds, and the vehicle totally traveled about 1.5 kilometers. The vehicle was driven around the teaching buildings and along the way it passed the school gate, main road, parking lot and so on. There were large number of walking or riding students and moving vehicles on the road. Therefore, the scenes are dynamic, various and complicated, especially some road crossing. The route of data collection and some typical scenes are shown in Figure.3.

B. Data Annotation

We annotate every point with semantic label in SemanticPOSS. Moreover, every dynamic object (people, car, rider) has a unique instance label. It is more difficult to label every point in 3D point clouds than label every pixel in 2D images, which is a reason why large-scale 3D dataset is rare. In order to reduce the requirement of manpower and annotation time as much as possible while ensuring label quality, we use the following annotation process shown in Figure.4:

1) Spatial-Temporal Segmentation: First, we use a segmentation program to divide the point clouds into some segments. It projects the point cloud to a sphere surface to generate a range image. The grey value of a pixel describes the distance between the point and the sensor. It uses the region grow algorithm to divide the range image into several segments, and tracks the same segments in different frames by a simple data association algorithm.

2) Track-wise Annotation: Then, we use a pre-annotation program to give every tracked segment a semantic label by some artificially defined rules. For example, it measures the length, width, height of the segment and finds a proper label to assign from the defined rules. For this track-wise annotation, all points of a tracked segment will get the same semantic label. Meanwhile, it combines information of range images and the matched images captured by cameras to recognize a tracked segment. It uses a pre-trained Mask R-CNN[24] to annotate people, riders, cars in images and transfers the labels to range images. With the help of the matched images, it will give a relatively good annotation result if the segment is close to the sensor. However, it may give wrong labels or wrong segments, which must be corrected by human. For some unlabeled segments or

Sensor	Maker	Data	Specification	Note
Pandora	Hesai Tech	LiDAR point cloud	40-line 0.33 degree vertical resolution 200m measurement range	/
		Image	4 wide-angle mono cameras 1 forward-facing color camera	Only used for annotation, to be published in future.
XW-GI7660 GPS/IMU	StarNeto	Vehicle pose	RTK 2cm+1ppm(CEP) Pose accuracy 0.05 degree Location accuracy 2cm	Corrected by SLAM[23].

TABLE II: Sensor configuration.

Authorized licensed use limited to: ShanghaiTech University. Downloaded on April 19,2025 at 07:50:21 UTC from IEEE Xplore. Restrictions apply.

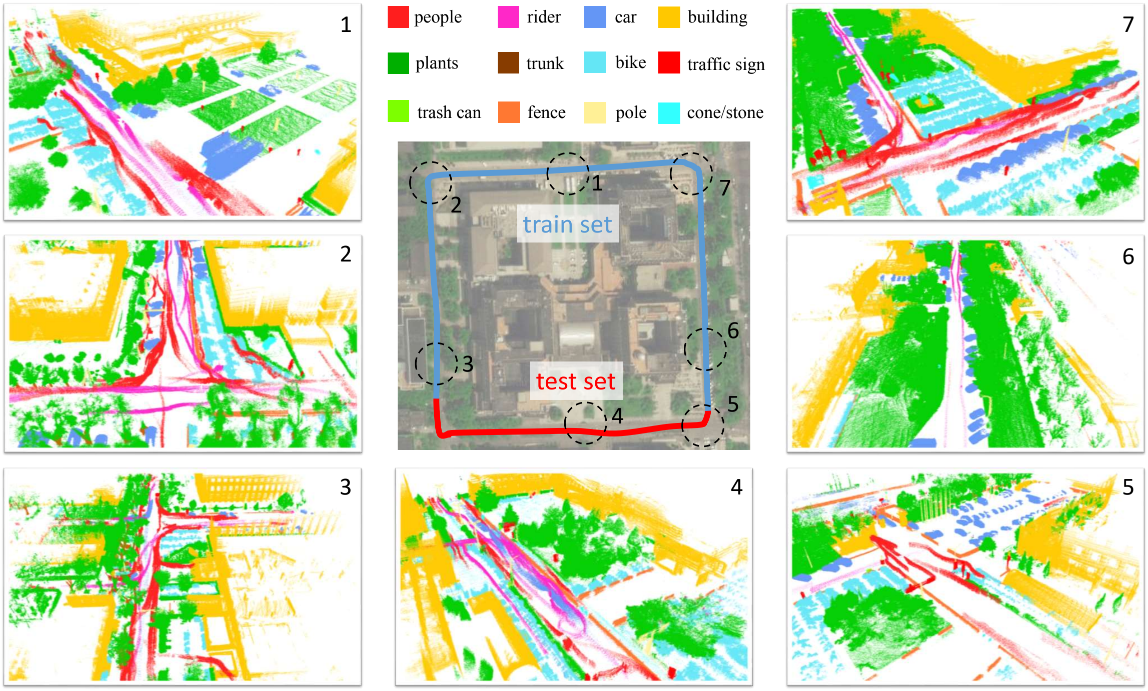


Fig. 3: SemanticPOSS dataset. The vehicle's driving route in the campus of PKU, some typical scenes, the train and test sets in experiments.

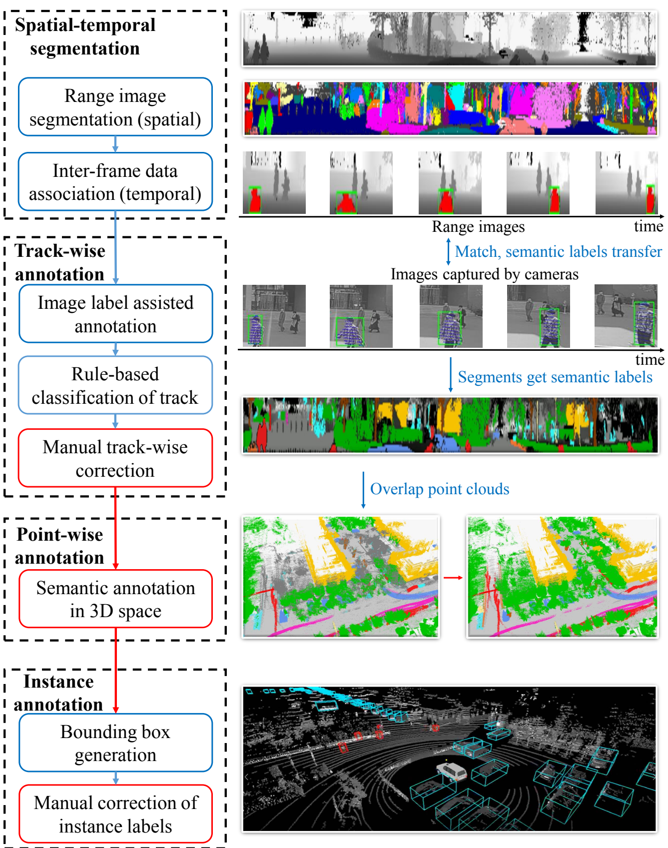


Fig. 4: Annotation process. The red boxes and texts mean the work is operated by human, and the blue means the work is operated by computer automatically.

obviously wrong annotated segments, we manually assign the correct labels to them. And we simply keep the *unlabeled* class to the segment which is hard for us to recognize what it is.

3) Point-wise Annotation: Next, we use the 3D point cloud annotating application provided by SemanticKITTI[7] to manually annotate the multiple overlapped point clouds. We overlap several single scan point clouds by calculating the sensor localization of every single scan from the locating information provided by GPS/IMU and calibration. We use SLAM algorithm[23] additionally to acquire a more accurate localization. The annotating application helps us conveniently visualize, operate and annotate the point clouds. Actually, we have already annotated lots of points after track-wise annotation, so we only need to annotate some unlabeled points and misclassified points according to the hint of track-wise annotation result. Because overlapped point clouds are much denser than single scan point cloud, the difficulty of manual annotation decreases remarkably with the help of pre-annotation. However, we need to change our view frequently to annotate points in 3D space. Though only part of the points are required to assign semantic labels, it still takes more than 70 percent of time for us to annotate.

4) Instance Annotation: Finally, we generate the instance labels. After all of the points get right semantic labels, we generate a bounding box for every dynamic instance and redo the data association as the first step. This step can be finished by a computer program automatically and it gives nearly completely correct instance labels without manual intervention. Annotators only need to modify some malposed bounding boxes.

Authorized licensed use limited to: ShanghaiTech University. Downloaded on April 19,2025 at 07:50:21 UTC from IEEE Xplore. Restrictions apply.

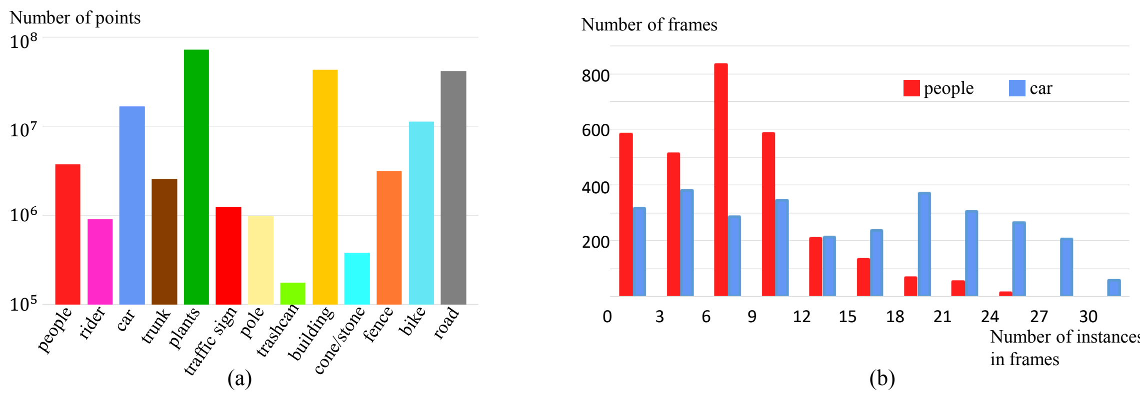


Fig. 5: Statistics of SemanticPOSS. (a) The number of points of each class. (b) Histogram of frames containing different number of instances

C. Dataset Analysis

We define 14 classes to label objects. Figure.5(a) shows the points statistics of every class in SemanticPOSS. There are 2988 point cloud frames with average 8.29 people instances and 15.02 car instances per frame in SemanticPOSS. Figure.5(b) shows how dynamic instances distribute in frames, where we can find that most frames contain 3-12 people and the number of car varies from frame to frame. The distribution of dynamic instances reflects the variety of the scenes. The large quantity of dynamic objects is the biggest difference between SemanticPOSS and other existing 3D point cloud datasets. For autonomous driving system, it is essential to accurately recognize and detect dynamic objects such as pedestrians and vehicles. Therefore, these complicated and dynamic scenes in SemanticPOSS may help researchers improve the performance and robustness of autonomous driving system in various environment.

IV. Experiments

A. Evaluation Metrics

Because our vehicle may pass a position twice or more when collecting data, to ensure the independence between training set and test set, we divide SemanticPOSS according to the location of our vehicle, as shown in Figure.3. For annotating convenience, we simply divide SemanticPOSS into 6 parts, 500 frames per part. Data part 3 is defined as test set, and the others are training set.

For 3D semantic segmentation task, deep learning models take point cloud as input and give a semantic prediction to every point. To evaluate the model performance, we use Intersection over Union (IoU) given by

$$IoU_c = \frac{TP_c}{TP_c + FP_c + FN_c} \tag{1}$$

where TP_c,FP_c,FN_c denote the number of true positive, false positive, false negative predictions of the class c. Let N be the number of classes used for measure, the mean IoU (mIoU) is defined as the arithmetic mean of IoU, namely

$$mIoU = \frac{1}{N}\sum_{c=1}^{N} IoU_c \tag{2}$$

We design two experiments to analyze the performance of different models and the feature of the dataset itself. The first experiment is training and testing on our SemanticPOSS dataset and comparing the generalization performance of different classes. We just ignore some confusing semantic labels like *unlabeled* when testing. The semantic labels used for evaluation are *people, rider, car, traffic sign, trunk, plants, pole, fence, building, bike, road*. The second experiment is cross-dataset generalization experiment[11] between SemanticPOSS and SemanticKITTI, namely training models on SemanticPOSS and testing on SemanticKITTI or contrary. The cross-dataset generalization experiment uses the same semantic labels to evaluate as the first experiment, but some semantic labels defined in SemanticKITTI are ignored or merged to keep classes coincident. Notice that all of the evaluation experiments use single scan point cloud as input, not overlapped point clouds, and weights of all classes are same when training.

B. Baseline Models

There are two main deep learning methods for 3D semantic segmentation task distinguished by input form. One uses the raw unordered point cloud as model input and another

Train and test on SemanticKITTI	people	rider	car	traffic sign	trunk	plants	pole	fence	building	bike	road	mIoU
PointNet++[2]	0.007	0.016	0.531	0.001	0.175	0.640	0.237	0.314	0.618	0.001	0.809	0.304
SequeezeSegV2[25]	0.154	0.393	0.784	0.201	0.322	0.726	0.192	0.404	0.697	0.122	0.882	0.470

Train and test on SemanticPOSS	people	rider	car	traffic sign	trunk	plants	pole	fence	building	bike	road	mIoU
PointNet++[2]	0.208	0.001	0.089	0.218	0.040	0.512	0.032	0.060	0.427	0.001	0.622	0.201
SequeezeSegV2[25]	0.184	0.112	0.349	0.110	0.158	0.563	0.045	0.255	0.470	0.324	0.713	0.298

TABLE III: Test results of the baseline models in SemanticKITTI and SemanticPOSS.

Authorized licensed use limited to: ShanghaiTech University. Downloaded on April 19,2025 at 07:50:21 UTC from IEEE Xplore. Restrictions apply.

test set \ model	mIoU		IoU of people	
	PN-KITTI	PN-POSS	PN-KITTI	PN-POSS
SemanticKITTI	**0.304**	0.164	0.007	**0.064**
SemanticPOSS	0.127	**0.201**	0.000	**0.208**

TABLE IV: Cross-dataset generalization experiment between SemanticPOSS and SemanticKITTI using PointNet++. PointNet++ training on SemanticKITTI is denoted as PN-KITTI, and PointNet++ training on SemanticPOSS is denoted as PN-POSS.

uses the range image as model input. We choose two typical deep learning models as baseline models, PointNet++[2] and SequeezeSegV2[25].

PointNet++ is the evolution of PointNet[1]. PointNet uses raw point cloud as input directly and outputs class of the whole point cloud or semantic labels of every points. It learns point features through max pooling, feature transformations, local and global features combination. PointNet++ ameliorates the problem that PointNet does not capture the contextual information of points. PointNet++ applies PointNet on a nested partitioning recursively to generate local features hierarchically, which significantly improve the robustness and generalization performance of the model.

SqueezeSegV2 is derived from SqueezeSeg[5] that uses 2D range image as input to give a point-wise prediction of the point cloud. It achieves a higher prediction accuracy by improving loss function, model structure and robustness to dropout noise of the point cloud. SqueezeSegV2 is based on Convolutional Neural Networks (CNN), Conditional Random Field (CRF) and adds Context Aggregation Module (CAM) to get further improvement. Besides, it boosts the model performance when training on a synthetic dataset and testing on a real dataset.

C. *Results and Discussion*

The results of the first experiment are shown in Table.III. For comparison we also show the performance of models training and testing on SemanticKITTI. In order to evaluate whether the variety of scenes have an effect on the performance of deep learning model, we do the second experiment using PointNet++. The results are shown in Table.IV, where we denote PointNet++ training on SemanticKITTI as PN-KITTI and PointNet++ training on SemanticPOSS as PN-POSS for simplicity. From the experimental results, we have the following findings:

1) There is high correlation between IoU and the scale of the corresponding class. For example, the number of road points is much more than the number of people points in datasets, and IoU of road is significantly higher than IoU of people whatever the model we use in Table.III. Therefore, sufficiency of data is a necessary condition for deep learning model to learn features of a class.
2) Both SemanticKITTI and SemanticPOSS reflect a degree of scene selection bias. As a consequence, mIoU of PN-KITTI drastically drops when testing on SemanticPOSS in Table.IV. Similarly, mIoU of PN-POSS also drops when testing on SemanticKITTI. The scene selection bias can be alleviated if multiple datasets are combines to train models.
3) PN-POSS performs better in labeling dynamic objects. The IoU of people improves when testing on both SemanticKITTI and SemanticPOSS if the model is training on SemanticPOSS. The results show that large number of dynamic instances really improve the abilities to label dynamic objects of deep learning models. Some examples of the model prediction is shown in Figure.6.

V. Conclusion and Future Work

In this work, we propose a large-scale 3D point cloud dataset with point-wise labels for 3D semantic segmentation task, SemanticPOSS. The dataset contains various and complicated scenes with large quantity of dynamic instances. Through experimental results we find that if training set contains more dynamic instances, deep learning models will get stronger ability to label dynamic objects. Therefore, SemanticPOSS can help deep learning models improve the prediction accuracy of people, car and so on in some degree.

We plan to give further extension to SemanticPOSS in future work. In addition, our Pandora sensor provides quite accurate match between point clouds from LiDAR and images from camera. So we may add images matched with point clouds to SemanticPOSS, to help the research of semantic segmentation methods using multi-sensor fusion.

References

[1] Charles R Qi, Hao Su, Kaichun Mo, and Leonidas J Guibas. PointNet: Deep learning on point sets for 3D classification and segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 652–660, 2017.

[2] 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.

[3] 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 *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2530–2539, 2018.

[4] Yuan Wang, Tianyue Shi, Peng Yun, Lei Tai, and Ming Liu. PointSeg: Real-time semantic segmentation based on 3D LiDAR point cloud. *arXiv preprint arXiv:1807.06288*, 2018.

[5] Bichen Wu, Alvin Wan, Xiangyu Yue, and Kurt Keutzer. SqueezeSeg: Convolutional neural nets with recurrent CRF for real-time road-object segmentation from 3D LiDAR point cloud. In *2018 IEEE International Conference on Robotics and Automation (ICRA)*, pages 1887–1893. IEEE, 2018.

[6] Timo Hackel, Nikolay Savinov, Lubor Ladicky, Jan D Wegner, Konrad Schindler, and Marc Pollefeys. Semantic3D. net: A new large-scale point cloud classification benchmark. *arXiv preprint arXiv:1704.03847*, 2017.

[7] Jens Behley, Martin Garbade, Andres Milioto, Jan Quenzel, Sven Behnke, Cyrill Stachniss, and Jurgen Gall. SemanticKITTI: A dataset for semantic scene understanding of LiDAR sequences. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 9297–9307, 2019.

[8] Mark Everingham, SM Ali Eslami, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman. The pascal visual object classes challenge: A retrospective. *International journal of computer vision*, 111(1):98–136, 2015.

Authorized licensed use limited to: ShanghaiTech University. Downloaded on April 19,2025 at 07:50:21 UTC from IEEE Xplore. Restrictions apply.

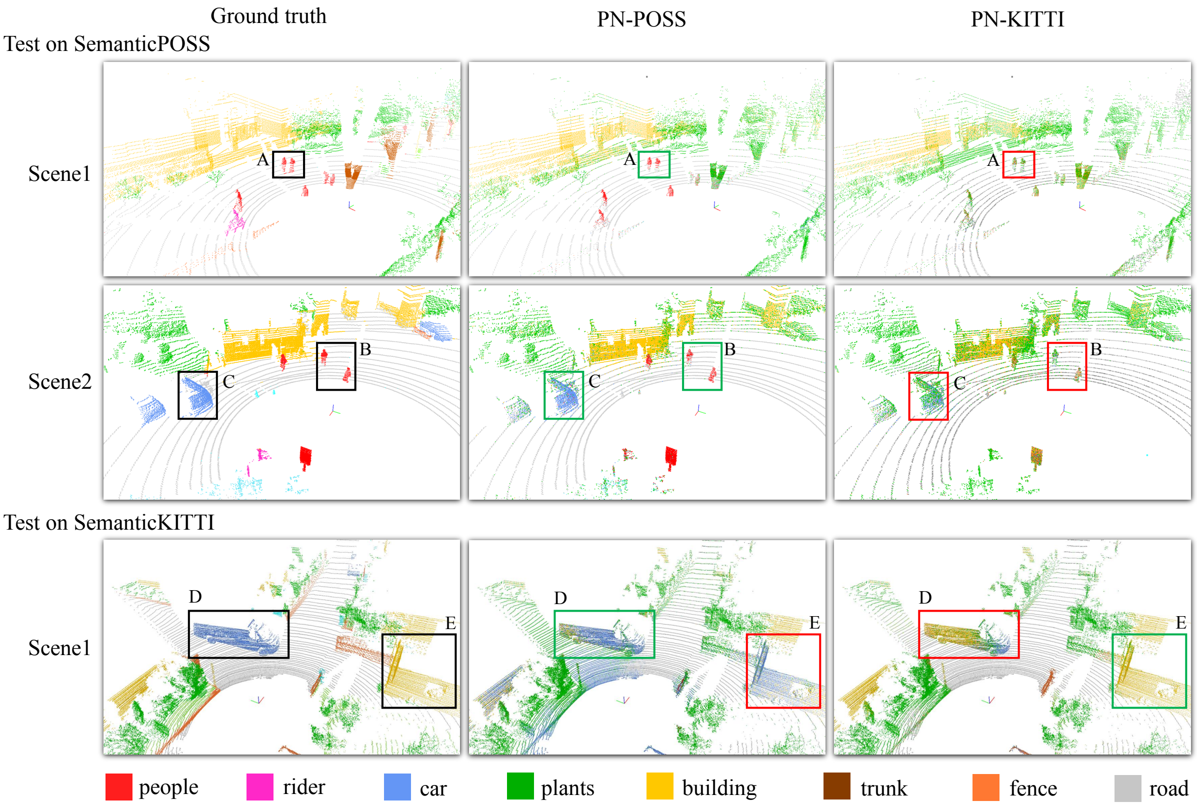


Fig. 6: Semantic segmentation results of some scene examples. The abilities to label static objects of PN-KITTI and PN-POSS are not much different. PN-POSS performs better in labeling dynamic objects, as shown in box A, B, C, D. However, it may make mistakes in some static objects, such as box E.

[9] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. ImageNet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. IEEE, 2009.

[10] 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 *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3213–3223, 2016.

[11] Antonio Torralba, Alexei A Efros, et al. Unbiased look at dataset bias. In *CVPR*, volume 1, page 7. Citeseer, 2011.

[12] Nathan Silberman, Derek Hoiem, Pushmeet Kohli, and Rob Fergus. Indoor segmentation and support inference from RGBD images. In *European Conference on Computer Vision*, pages 746–760. Springer, 2012.

[13] Shuran Song, Samuel P Lichtenberg, and Jianxiong Xiao. Sun RGB-D: A RGB-D scene understanding benchmark suite. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 567–576, 2015.

[14] Angela Dai, Angel X Chang, Manolis Savva, Maciej Halber, Thomas Funkhouser, and Matthias Nießner. Scannet: Richly-annotated 3D reconstructions of indoor scenes. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5828–5839, 2017.

[15] Xinyu Huang, Peng Wang, Xinjing Cheng, Dingfu Zhou, Qichuan Geng, and Ruigang Yang. The ApolloScape open dataset for autonomous driving and its application. *arXiv preprint arXiv:1803.06184*, 2018.

[16] Iro Armeni, Ozan Sener, Amir R Zamir, Helen Jiang, Ioannis Brilakis, Martin Fischer, and Silvio Savarese. 3D semantic parsing of large-scale indoor spaces. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1534–1543, 2016.

[17] Daniel Munoz, J Andrew Bagnell, Nicolas Vandapel, and Martial Hebert. Contextual classification with functional max-margin markov networks. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 975–982. IEEE, 2009.

[18] Xavier Roynard, Jean-Emmanuel Deschaud, and François Goulette. Paris-Lille-3D: A large and high-quality ground-truth urban point cloud dataset for automatic segmentation and classification. *The International Journal of Robotics Research*, 37(6):545–557, 2018.

[19] Andreas Geiger, Philip Lenz, and Raquel Urtasun. Are we ready for autonomous driving? the KITTI vision benchmark suite. In *2012 IEEE Conference on Computer Vision and Pattern Recognition*, pages 3354–3361. IEEE, 2012.

[20] 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.

[21] Xiangyu Yue, Bichen Wu, Sanjit A Seshia, Kurt Keutzer, and Alberto L Sangiovanni-Vincentelli. A LiDAR point cloud generator: from a virtual world to autonomous driving. In *Proceedings of the 2018 ACM on International Conference on Multimedia Retrieval*, pages 458–464. ACM, 2018.

[22] David Griffiths and Jan Boehm. Synthcity: A large scale synthetic point cloud. *arXiv preprint arXiv:1907.04758*, 2019.

[23] Huijing Zhao, Masaki Chiba, Ryosuke Shibasaki, Xiaowei Shao, Jinshi Cui, and Hongbin Zha. SLAM in a dynamic large outdoor environment using a laser scanner. In *2008 IEEE International Conference on Robotics and Automation*, pages 1455–1462. IEEE, 2008.

[24] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask R-CNN. In *Proceedings of the IEEE international conference on computer vision*, pages 2961–2969, 2017.

[25] Bichen Wu, Xuanyu Zhou, Sicheng Zhao, Xiangyu Yue, and Kurt Keutzer. SqueezeSegV2: Improved model structure and unsupervised domain adaptation for road-object segmentation from a lidar point cloud. In *2019 International Conference on Robotics and Automation (ICRA)*, pages 4376–4382. IEEE, 2019.

Authorized licensed use limited to: ShanghaiTech University. Downloaded on April 19,2025 at 07:50:21 UTC from IEEE Xplore. Restrictions apply.