Fast Online Incremental Segmentation of 3D Point Clouds from Disaster Sites

Yosuke Yajima[a], Seongyong Kim[b], Jingdao Chen[a], and Yong K. Cho[b]

[a]Institute for Robotics and Intelligent Machines, Georgia Institute of Technology, U.S.A
[b]School of Civil and Environmental Engineering, Georgia Institute of Technology, U.S.A

E-mail: yyajima@gatech.edu, skim3310@gatech.edu, jchen490@gatech.edu, yong.cho@ce.gatech.edu

Abstract -

In monitoring disaster sites, mobile robots represent a fast, reliable, and practical option to remotely inspect active areas in disaster management for applications such as risk management, search and rescue, and structural assessment purposes. Mobile robots can efficiently collect laser scan data and reconstruct the state of ongoing disaster relief in the form of 3D point clouds. Current point cloud processing methods are mostly designed to work as a post-processing step and are inefficient when applied in real-time. Additionally, object segmentation on point cloud data from the disaster sites is challenging due to data impurities and occlusions. To overcome these issues, this paper proposes an instance point cloud segmentation method that incrementally builds a 3D map for robotic scans of infrastructure. In the first step, the proposed neural network, named Dynamic Graph PointNet (DGPointNet), is trained to classify objects in the disaster environment while building up a semantic 3D map of the environment. Additionally, the proposed method predicts object instance labels by using a sequence of predicted semantic point cloud data. The proposed method shows strong performance over the state of art segmentation models in terms of semantic segmentation, instance segmentation, and processing time using point cloud data collected from a custom-built laser-scanning robot at an outdoor simulated disaster site.

Keywords -

Point cloud segmentation; laser scanning; robotics; disaster site; deep learning

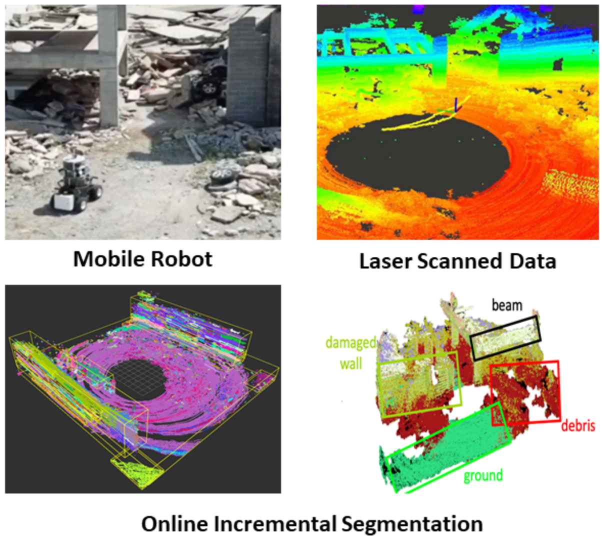

Figure 1. Overall framework for collecting and processing 3D point clouds from disaster sites.

1 Introduction

In disaster sites, mobile robots represent a safer and more labor-efficient option to carry out inspection tasks such as risk estimation and assessment of damaged building in post-disaster sites for reconstruction of damaged constructions. According to the NOAA's National Centers for Environmental Information, the natural disaster costs the U.S. over $600 billion in the last 5 years (2016-2020) [1]. Current damage assessment and quality inspections often involve a long inspection activity that requires inspectors to collect data on the disaster sites due to difficulty in accessing disaster sites. The inspection process often involves safety concerns such as exposing toxic chemical to human being and building collapse.

Due to the safety concerns and recent development in computing technology, mobile robots can be used to collect laser scan data and reconstruct the as-is state of various building entities on the site in the form of 3D point clouds [2, 3, 4]. Several methods use mobile robots to monitor the condition of damaged building in the disaster sites such as mapping disaster sites with ground robot [5] and with an aerial robot [6]. Even though a large amount of point cloud data can be easily collected, the process of automatically organizing and extracting useful information from the noisy data remains a challenging task. Especially in the disaster sites, the collected data often includes incomplete objects such as collapsed buildings and damaged objects. The mobile robot needs to perform segmentation of the acquired data to carry out obstacle detection, object recognition, and other scene understanding tasks. Current segmentation methods are mostly designed to process point cloud data one at a time and are applied only as a post-processing step. In addition, segmentation methods that are trained with complete models of objects do not work well because robotic scans are usually noisy or occluded.

This paper proposes an incremental point cloud segmentation method for robotic scans of disaster sites using a 3D Light Detection and Ranging (Lidar) sensor. The scans are first registered based on simultaneous localization and mapping (SLAM) and stored in a voxel-based lookup table. The registered point cloud is then passed to the proposed deep learning model, named Dynamic Graph PointNet (DGPointNet) that predicts semantic object labels. The DGPointNet is robust to detect objects with data impurities and occlusions by learning both local neighbor and global point features. The output of the deep learning model is then processed with incremental segmentation algorithm that merges new scan points into existing points to form instance clusters based on similarity in feature space. The proposed method is incremental in that each new scan is processed and combined with information from previous scans without having to recompute the entire scene. Results are then used to create a 3D object-level map of the disaster site. The proposed method is validated using point cloud data collected from a laser scanning robot at an outdoor simulated disaster site shown in Figure 1. The key contributions of the proposed method are summarized as follows:

- Fast online instance segmentation that directly takes an input of Lidar scanned data and outputs predicted instance object labels
- Development of a light weight deep learning model suitable for dataset from an outdoor environment with impurities and occlusions.
- Evaluation of semantic and instance segmentation methods using dataset from a real-world simulated post disaster sites environment.

2 Related work

2.1 Geometry-based segmentation

In the past, several approaches use geometric segmentation methods that rely on surface normals, curvatures, and edge. Hough Transform and the Random Sample Consensus have also been widely employed as fundamental algorithms for detecting simple geometric objects based on their model parameters [7, 8]. Until recently, RANSAC-based algorithms for the plane segmentation has continued to be improved [9, 10]. Clustering is another common step for point cloud segmentation. Region growing which progressively gathers nearby points regarded as the same class or regions with cohesive features, has been a widely used, and even a learnable model for region growing has been developed [11]. Using density-based spatial clustering of applications with noise (DBSCAN), Czerniawski *et al.* [12] proposed a method to detect planar objects in indoor scenes. However, the methods referring to geometry features have limitations in being robustly applied to a disaster site of highly unstructured environment, containing varied noise and deformed objects with complex geometry not fitting to predefined geometry features.

2.2 Data-driven models for segmentation

With the growing popularity of data-driven models, deep architectures for classifying each 3D point into semantic categories have made significant progress [13, 14]. Compared to ones using hand-crafted features, these deep architectures show better performance and robustness, but still are in active development due to issues such as sparsity, randomness, and the unstructured nature of point clouds. As an indirect method, to facilitate convolutional neural networks for the segmentation, researchers have converted a point cloud into a regular structure prior to the processing, such as multi-views [15], voxel grids [16]. However, since these data conversion cause information loss and computational complexity, PointNet [17] directly process each point by extracting robust features from the transformation and permutation of the point cloud. Since then, many methods based on PointNet have been proposed, referring to the direct processing of each point [14]. To find optimistic receptive field for the segmentation task, Qi *et al.* [18] adopted a multi-scale concept which recursively extracted local features using PointNet layer, and enhanced the results of segmentation, but, the inference time increased several times. Regarding a point cloud as structured graph data composed of a series of nodes and edges, Wang *et al.* [19] proposed DGCNN by replacing multi-layers of PointNet with stacks of edge convolution from Graph Convolutional Networks. It shows a significant effect in extracting analogous features but increases the complexity of the network as well as number of the training parameters. Since the methods based on GCNN, compare to others, have advantages in examining the boundary feature, researchers have now increasingly adopted these graph-based algorithms [20, 21]. Other data-driven models such as Point Transformer [22] and SCF-Net [23] are not considered in this study due to the slow inference speed of the models.

2.3 Incremental Segmentation

Although researchers have proposed many segmentation models so far, to leverage them integrated with robotic scanning in real-world sites, one thing that needs to be

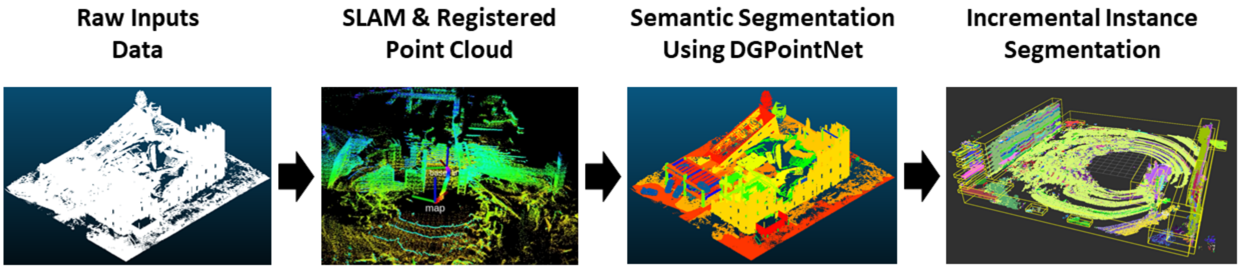

Figure 2. 3D point cloud processing framework.

added is incremental segmentation, which progressively updates class labels of the point in a dynamic fashion while collecting data simultaneously. To understand scenes even when the entire site is not scanned, researchers have proposed a method to supplement the segmentation results with online data in varied situation [24, 25, 26]. Finman *et al.* [24] proposed a method to combine partially segmented results by iterative voting from RGB-D maps. Dube *et al.* [25] selectively updated dynamic voxel grids online and adopted incremental fashion in region growing algorithm while caching the geometric consistencies. Unlike others, Multi-view Context Pooling Network based on PointNet, Chen *et al.* [27] proposed, implicitly updated an instance id of each point online using the deep network that was directly trained from the raw point cloud annotation. By suggesting a pooling operation to assemble local features from neighbor points to the global features, MCPNet updated each point while reflecting the contextual information of neighbors.

However, these methods were applied on limited dataset of indoor environments or confined outdoor sites. The goal of the proposed method is to perform incremental instance segmentation on real-world disaster sites. The proposed method directly consumes points from the scanned data and performs the incremental segmentation while fusing local features to the global features, especially using online data collected by the unmanned ground vehicle from challenging disaster site environments with a significant amount of debris and deformed objects.

3 Methodology

3.1 Overview

This research proposes a robotic scanning and point cloud segmentation framework for identifying construction related objects from post disaster sites. Figure 2 shows an overall data collection and processing framework where each distinct object is segmented into an object instance. In the first step, an input point cloud is captured by a laser scanning device using a mobile robot. Then, the simultaneous localization and mapping (SLAM) module takes the point cloud data and registers points to reconstruct a 3D map of the damaged building structures around the robot. For each point cloud scan, the semantic segmentation module (DGPointNet) classifies object classes in the registered point cloud and outputs a set of semantic object labels. In the final step, the instance segmentation module uses the predicted semantic labels to estimate object instance labels by grouping points from the current and previous laser scans. The end result of the point cloud data processing framework is a densely labeled 3D map of the disaster site that contains information about the obstacles, building elements, and debris in the surrounding environment. The following sections provide more details regarding methods and design choices.

3.2 3D input data and registration

The proposed 3D data registration method is visualized in Figure 3 where the input data is a raw laser scanned point cloud that is represented as unordered 3D points that contains position vectors (x, y, z). As the mobile robot is moving around the disaster site, laser scan data is collected along with Inertial Measurement Unit (IMU) and Global Positioning System (GPS) measurements. A SLAM module based on the LeGO-LOAM [28] algorithm is used to localize the robot using sensor measurements. Each laser scan is registered on to the global coordinate system by applying a transformation matrix based on the robot's pose at each time step. These registered point clouds are used to train and evaluate semantic segmentation model described in the next sections.

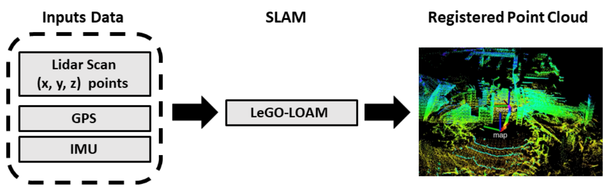

Figure 3. Automated point cloud registration step to generate the input dataset for our framework.

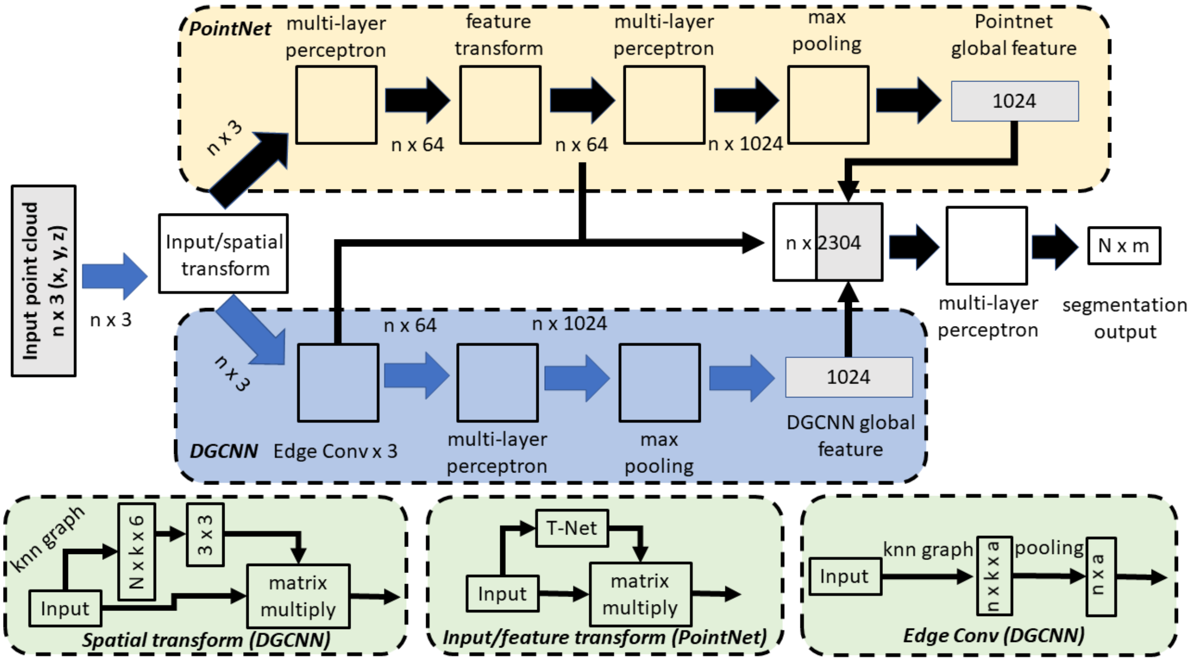

Figure 4. The proposed DGPointNet architecture for point cloud semantic segmentation.

3.3 Semantic segmentation with DGPointNet

Semantic segmentation is performed with a deep neural network to assign semantic class labels to each point. The semantic labels consist of classes such as *ground*, clutter, *wall*, etc. that are used to differentiate between different objects on the disaster site. The proposed deep learning architecture is shown in Figure 4. The proposed network combines elements from both PointNet [17] and DGCNN [19], integrated into a single neural network. The advantage of this model is that it combines the benefits of PointNet for learning the global context of the input point cloud data and DGCNN for learning the local context of the input point cloud using the edge convolution layers.

The proposed model first takes a set of n points with feature vectors including position vector (x, y, z) and normalized z position to each data scene (a real number ranging from 0 to 1) and passes them through input and spatial transform layers. In the PointNet module (yellow box in Figure 4), both input and feature transform use a T-Net that applies an affine transformation to the given point cloud inputs. The points are then passed through a series of convolution layers and multi-layer perceptron. The points features are aggregated using a max pooling layer to obtain a global feature. The outputs of the intermediate multi-layer perceptron and global features are concatenated to a shared multi-layer perceptron. Similarly, in the DGCNN module (blue box in Figure 4), the points are passed through a series of edge convolutions and multi-layer perceptrons. The edge convolution layer applies k nearest neighbor algorithm to group input points into k feature clusters and aggregates the features by a max pooling layer. The outputs of each edge conv layers and global features are concatenated to a shared multi-layer perceptron. The model predicts n x m vectors as outputs for n input points and m semantic classes.

3.4 Incremental instance segmentation

An incremental instance segmentation module is responsible for clustering the predicted points from the semantic segmentation module into individual object instances. For example, given a set of points corresponding to wall objects, the instance segmentation module will cluster the points into *wall #1*, *wall #2*, *wall #3*, etc. This step is important for identifying the number of building elements on the disaster site and for assigning damage information to each individual building element. The overall algorithm is specified in Algorithm 1. The method first initializes a voxel grid dictionary v to remove duplicate input points from the segmentation outputs. The voxel grid allows the proposed method to only update the instance labels for the newly-scanned regions and reduce computation time so that the method can run online on the mobile robot. The update list u is used to store new points that are used in the clustering method. The distance limit k is used a neighborhood threshold to append all neighboring points in the x, y, z position for each point in the updated points list. The surface normal threshold and semantic label are both used to group the updated point into existing clusters.

If the updated point does not match with an existing object instance, a new object instance is created.

Algorithm 1: Incremental instance segmentation

```
Input: Point cloud data with semantic
       segmentation labels
Result: Point cloud data with instance
        segmentation labels
Initialize updated points list u ← ∅
Initialize voxel grid dictionary v ← ∅
Initialize neighbor points list w ← ∅
for each point in input data do
    if point does not exists in voxel grid then
        Append point to the end of v and u
    end
end
for each updated point do
    Append all points within k distance from
      update point to w;
    for each nearest point do
        Compute surface normals
    end
    if updated point > surface normal threshold
      and updated point class ID == neighbor point
      class ID then
        Add to the existing object instance
    else
        Initialize a new object instance
    end
    u ← ∅ w ← ∅
end
```

4 Results

4.1 Experimental setup and dataset

A field experiment was conducted at the Guardian Centers disaster training facility (Figure 5). The Guardian Centers facility consists of numerous damaged concrete structures that are cracked, deformed, or collapsed. In this experiment, a 4-wheel ground mobile equipped with a 3D Lidar sensor, IMU, and GPS sensor is used to collect laser scan data at the site. The 3D Lidar sensor is a Velodyne VLP-16 which has 16 beams, angular horizontal and vertical resolution of $0.1° - 0.4°$ and $2°$, accuracy of ±3cm, and measurement range of 100 m. The semantic object classes consist of 8 different categories such as clutter, ground, wall, beam, girder, slab, column, door. The collected point cloud data is downsampled to 0.1m to reduce point cloud size. To eliminate sparse Lidar data, each scanned data is limited by ±15m in x, y, z coordinates.

To evaluate the performance of semantic and instance segmentation, a total of 9 robot scanned data with the Velodyne Lidar at a different part of the disaster sites scene are collected. 7 point cloud scenes are used as the training set and 2 point cloud scenes are used as the test set. Each scanned data consists of 2-3 minutes sequential points that cover a part of the simulated test sites.

Figure 5. A mobile robot deployed to collect laser scans at the Guardian Centers disaster training facility.

4.2 Semantic segmentation results

The proposed method is compared with PointNet and PointNet++ models in terms of point accuracy and average Intersection over Union (IoU) using the test dataset shown in Table 1. These baseline models are the state of art deep learning models that directly take point cloud data with geometry features and output predictions of semantic point labels. The input points size of these models are set to 4096 points with a block size of $10x10$ (m). The input points consist of feature vectors with x, y, z position and normalized z position to each test data scene. The clustering hyper-parameters k used in the DGPointNet is set to 20.

In terms of mean IoU and point accuracy, the proposed method shows the highest mIoU score of 55.1 % and the highest point accuracy of 77.8 %. Table 2 shows results of individual IoU performance per object class. The proposed method achieves the highest IoU for every object category. The proposed model shows strong performance on the small objects such as beam, girder, and door categories. The performance of the proposed method is visualized in Figure 6.

Table 1. Segmentation results evaluated in terms of mIoU and point accuracy on the Guardian Centers dataset using the test dataset.

Method	mIoU (%)	Point Acc.(%)
PointNet [17]	41.4	68.6
PointNet++ [18]	28.2	57.9
Proposed DGPointNet	55.1	77.8

4.3 Instance Segmentation Results

The instance segmentation results are evaluated using the metrics of normalized mutual information (NMI), ad-

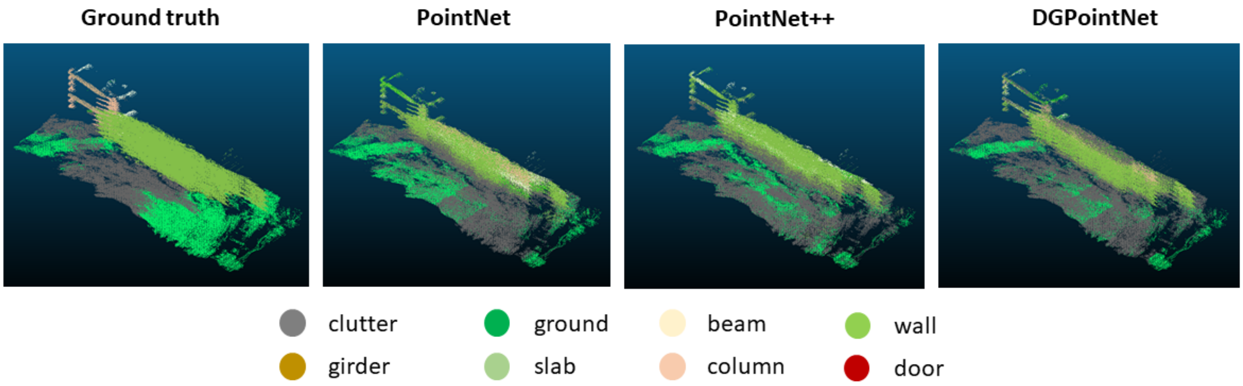

Figure 6. Results of semantic segmentation on the test dataset.

Table 2. Semantic segmentation results of each class on the Guardian Centers dataset. The evaluation metric is IoU(%) on the test dataset.

Object Class	total	clutter	ground	beam	wall	girder	slab	column	door
Test points	-	293673	127929	24495	261549	12212	147952	57449	2477
PointNet [17]	41.4	53.2	63.2	48.6	61.2	23.9	47.6	22.7	11.1
PointNet++ [18]	28.2	52.3	51.9	33.6	37.2	12.8	29.1	8.4	0.0
Proposed	55.1	63.3	68.7	55.5	68.2	45.7	66.0	46.1	27.3

justed rand index (ARI), and V measure score (VMS) [29, 30]. In Table 3, the proposed method is compared with other baseline deep learning models to validate the evaluation metrics. The result shows that the proposed method achieves the second-best result among the baseline models. The results of instance segmentation using DGPointNet is visualized in Figure 7. The bounding boxes show individual objects are separated from each other.

Table 3. Instance segmentation and classification results on the test dataset.

Method	NMI	AMI	VMS
PointNet [17]	34.4	33.7	34.4
PointNet++ [18]	37.1	35.7	37.1
Proposed DGPointNet	35.3	33.9	35.3

4.4 Computation time evaluation

Table 4 shows the computation time and average points processed per scan measured on Intel® Core™ i9-10980HK CPU and NVIDIA GeForce RTX 2080 GPU. The proposed method has a relatively fast computation speed of 1.18 s per scan compared to the PointNet++ models while the average processed points are 5742 points. This result shows that the proposed method can be applied in near real-time.

Table 4. Computation time and average points processed on the Guardian Centers dataset.

Method	Time (s)	Points per scan
PointNet [17]	0.43	5627
PointNet++ [18]	15.72	6052
Proposed DGPointNet	1.18	5742

5 Conclusions

In this work, the proposed method shows a novel process that directly takes raw point cloud data and predicts instance labels in near real-time. The semantic segmentation network learns from both local neighbor and global point features through PointNet and DGCNN. This model also shows strong performance on semantic segmentation of small object with data impurities and occlusions. Additionally, the incremental segmentation method clusters semantic prediction labels into an object instance by integrating a sequence of Lidar scanned data. The proposed method shows that the semantic and instance segmentation can both run in near real-time to classify objects on

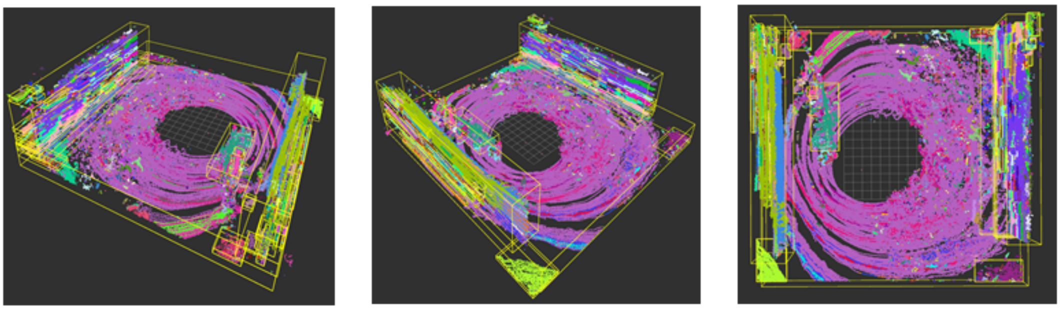

Figure 7. Results of Instance segmentation. Each object instance is assigned a different color.

disaster sites data.

While the proposed method shows a fast online segmentation performance, the experiment results indicate that the model could be revised to improve efficiency for real-time performance in future research. Additionally, ablation studies such as hyper parameter tuning of the segmentation model can be considered to optimize performance of the proposed method.

Acknowledgments

This material is based upon work supported by the U.S. Air Force Office of Scientific Research (AFOSR)(Award No. FA2386-17-1-4655); the Technology Innovation Program (No. 2017-10069072) funded by the Ministry of Trade, Industry & Energy (MOTIE, Korea); and National Science Foundation (Award No. 2040735). Any opinions, findings, conclusions, or recommendations expressed in this material are those of the authors and do not necessarily reflect AFOSR, MOTIE, or NSF's views.

References

[1] Adam B. Smith. 2020 U.S. billion-dollar weather and climate disasters in historical context, 2021. URL `https://www.climate.gov`.

[2] Mei-Po Kwan and Daniel M Ransberger. Lidar assisted emergency response: Detection of transport network obstructions caused by major disasters. *Computers, Environment and Urban Systems*, 34(3): 179–188, 2010.

[3] Soonwook Kwon, Mina Lee, Moonju Lee, Sukhan Lee, and Junbok Lee. Development of optimized point cloud merging algorithms for accurate processing to create earthwork site models. *Automation in Construction*, 35:618–624, 2013. ISSN 0926-5805. doi:https://doi.org/10.1016/j.autcon.2013.01.004.

[4] Seoungjun Lee, Dongsoo Har, and Dongsuk Kum. Drone-assisted disaster management: Finding victims via infrared camera and lidar sensor fusion. In *2016 3rd Asia-Pacific World Congress on Computer Science and Engineering (APWC on CSE)*, pages 84–89. IEEE, 2016.

[5] Jingdao Chen and Yong K. Cho. Detection of Damaged Infrastructure on Disaster Sites using Mobile Robots. *2019 16th International Conference on Ubiquitous Robots, UR 2019*, (July):648–653, 2019. doi:10.1109/URAI.2019.8768770.

[6] Yundong Li, Wei Hu, Han Dong, and Xueyan Zhang. Building damage detection from post-event aerial imagery using single shot multibox detector. *Applied Sciences (Switzerland)*, 9(6), 2019. ISSN 20763417. doi:10.3390/app9061128.

[7] Anh Nguyen and Bac Le. 3D point cloud segmentation: A survey. In *2013 6th IEEE conference on robotics, automation and mechatronics (RAM)*, pages 225–230. IEEE, 2013. ISBN 1479912018.

[8] E. Grilli, F. Menna, and F. Remondino. A review of point clouds segmentation and classification algorithms. *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences - ISPRS Archives*, 42(2W3):339–344, 2017. ISSN 16821750. doi:10.5194/isprs-archives-XLII-2-W3-339-2017.

[9] Bo Xu, Wanshou Jiang, Jie Shan, Jing Zhang, and Lelin Li. Investigation on the weighted ransac approaches for building roof plane segmentation from lidar point clouds. *Remote Sensing*, 8(1):5, 2016.

[10] Lin Li, Fan Yang, Haihong Zhu, Dalin Li, You Li, and Lei Tang. An improved ransac for 3d point cloud plane segmentation based on normal distribution transformation cells. *Remote Sensing*, 9(5): 433, 2017.

[11] Jingdao Chen, Zsolt Kira, and Yong K Cho. Lrgnet: Learnable region growing for class-agnostic point cloud segmentation. *IEEE Robotics and Automation Letters*, 6(2):2799–2806, 2021.

[12] T Czerniawski, B Sankaran, M Nahangi, C Haas, and F Leite. 6d dbscan-based segmentation of building point clouds for planar object classification. *Automation in Construction*, 88:44–58, 2018.

[13] J Zhang, X Zhao, Z Chen, and Z Lu. A Review of Deep Learning-Based Semantic Segmentation for Point Cloud. *IEEE Access*, 7: 179118–179133, 2019. ISSN 2169-3536 VO - 7. doi:10.1109/ACCESS.2019.2958671.

[14] Yulan Guo, Hanyun Wang, Qingyong Hu, Hao Liu, Li Liu, and Mohammed Bennamoun. Deep learning for 3d point clouds: A survey. *IEEE transactions on pattern analysis and machine intelligence*, 2020. ISSN 0162-8828.

[15] Hang Su, Subhransu Maji, Evangelos Kalogerakis, and Erik Learned-Miller. Multi-view convolutional neural networks for 3d shape recognition. In *Proceedings of the IEEE international conference on computer vision*, pages 945–953, 2015.

[16] Daniel Maturana and Sebastian Scherer. Voxnet: A 3d convolutional neural network for real-time object recognition. In *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 922–928. IEEE, 2015. ISBN 1479999946.

[17] Charles R. Qi, Hao Su, Kaichun Mo, and Leonidas J. Guibas. PointNet: Deep learning on point sets for 3D classification and segmentation. *Proceedings - 30th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017*, 2017-Janua:77–85, 2017. doi:10.1109/CVPR.2017.16.

[18] Charles R Qi, Li Yi, Hao Su, and Leonidas J Guibas. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. *arXiv preprint arXiv:1706.02413*, 2017.

[19] Yue Wang, Yongbin Sun, Ziwei Liu, Sanjay E Sarma, Michael M Bronstein, and Justin M Solomon. Dynamic graph cnn for learning on point clouds. *Acm Transactions On Graphics (tog)*, 38(5):1–12, 2019.

[20] Kuangen Zhang, Ming Hao, Jing Wang, Clarence W de Silva, and Chenglong Fu. Linked dynamic graph cnn: Learning on point cloud via linking hierarchical features. *arXiv preprint arXiv:1904.10014*, 2019.

[21] Gusi Te, Wei Hu, Amin Zheng, and Zongming Guo. Rgcnn: Regularized graph cnn for point cloud segmentation. In *Proceedings of the 26th ACM international conference on Multimedia*, pages 746–754, 2018.

[22] Hengshuang Zhao, Li Jiang, Jiaya Jia, Philip H. S. Torr, and Vladlen Koltun. Point transformer. *CoRR*, abs/2012.09164, 2020. URL `https://arxiv.org/abs/2012.09164`.

[23] Siqi Fan, Qiulei Dong, Fenghua Zhu, Yisheng Lv, Peijun Ye, and Fei-Yue Wang. Scf-net: Learning spatial contextual features for large-scale point cloud segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 14504–14513, June 2021.

[24] Ross Finman, Thomas Whelan, Michael Kaess, and John J. Leonard. Efficient incremental map segmentation in dense RGB-D maps. In *Proceedings - IEEE International Conference on Robotics and Automation*, pages 5488–5494. IEEE, 2014. ISBN 1479936855. doi:10.1109/ICRA.2014.6907666.

[25] Renaud Dubé, Mattia G Gollub, Hannes Sommer, Igor Gilitschenski, Roland Siegwart, Cesar Cadena, and Juan Nieto. Incremental-segment-based localization in 3-d point clouds. *IEEE Robotics and Automation Letters*, 3(3):1832–1839, 2018. ISSN 2377-3766.

[26] Anton Milan, Trung Pham, Kumar Vijay, Douglas Morrison, Adam W Tow, L Liu, Jordan Erskine, Riccardo Grinover, Alec Gurman, and T Hunn. Semantic segmentation from limited training data. In *2018 IEEE International Conference on Robotics and Automation (ICRA)*, pages 1908–1915. IEEE, 2018. ISBN 1538630818.

[27] Jingdao Chen, Yong Kwon Cho, and Zsolt Kira. Multi-View Incremental Segmentation of 3-D Point Clouds for Mobile Robots. *IEEE Robotics and Automation Letters*, 4(2):1240–1246, 2019. ISSN 23773766. doi:10.1109/LRA.2019.2894915.

[28] Tixiao Shan and Brendan Englot. Lego-loam: Lightweight and ground-optimized lidar odometry and mapping on variable terrain. In *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 4758–4765, 2018. doi:10.1109/IROS.2018.8594299.

[29] A. Rosenberg and Julia Hirschberg. V-measure: A conditional entropy-based external cluster evaluation measure. In *EMNLP-CoNLL*, 2007.

[30] Nguyen Xuan Vinh, Julien Epps, and James Bailey. Information theoretic measures for clusterings comparison: Variants, properties, normalization and correction for chance. *Journal of Machine Learning Research*, 11(95):2837–2854, 2010. URL `http://jmlr.org/papers/v11/vinh10a.html`.