The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, Volume XLVIII-G-2025
ISPRS Geospatial Week 2025 “Photogrammetry & Remote Sensing for a Better Tomorrow…”, 6–11 April 2025, Dubai, UAE

# Finding the Optimal Convolutional Kernel Size for Semantic Segmentation of Pole-like Objects in Lidar Point Clouds

Zezheng Zhang[1], Davood Shojaei[1], Kourosh Khoshelham[2]

[1] Centre for Spatial Data Infrastructures and Land Administration, Department of Infrastructure Engineering, The University of Melbourne, Melbourne, Victoria, Australia
[2] Department of Infrastructure Engineering, The University of Melbourne, Melbourne, Victoria, Australia
E-mails: zezzhang@student.unimelb.edu.au, (shojaeid, k.khoshelham)@unimelb.edu.au

**Keywords:** Mobile Laser Scanning, Street Furniture, Road Assets, Semantic Segmentation, Deep Learning.

**Abstract**

Pole-like objects (PLOs) are important street assets in urban environments, yet current deep learning methods often underperform in their segmentation compared to other objects. The main challenge is determining the right kernel size to effectively understand the unique structure of PLOs with an appropriate receptive field. In this study, we improve the segmentation performance of PLOs by optimizing the kernel size in a KPConv-based network. Our experiments show that kernel size of 9 yields an Intersection over Union (IoU) of 95.02% on the Parkville-3D dataset. We also develop a post-processing approach that transforms semantic segmentation outputs into panoptic segmentation results, enabling accurate detection of individual PLO instances. Furthermore, qualitative tests on an independent, unlabelled point cloud dataset from a different urban area demonstrate that our method consistently achieves accurate segmentation.

## 1. Introduction

Given the significance of PLOs in infrastructure maintenance (Cabo et al., 2014), 5G network planning (Gholampooryazdi et al., 2017), and the development of High Definition Maps (HD Maps) (Dong et al., 2023), there is a pressing need for a specialized deep-learning model that excels in recognizing those assets. Current deep learning approaches for point cloud semantic segmentation underperform when identifying Pole-like objects (PLOs). For example, as shown by Roynard et al. (2018), the top five methods in the Paris-Lille-3D benchmark achieved an 82.74% mean Intersection over Union (mIoU) across various classes but only secured a 73.94% mIoU for PLOs.

In our analysis of existing studies, we observed that while network designs are optimized for objects that are small and elongated, they struggle with the slender characteristics of poles. This limitation often results in the poor segmentation performance. To address the identified challenge, we have re-engineered the kernel of the selected network to feature a larger receptive field.

We reviewed the literature on various deep learning networks for point clouds and chose KPConv (Thomas et al., 2019) as a starting point. We noticed that KPConv is widely used as a baseline and has shown good performance on different datasets during our review. In addition, KPConv's design preserves many network settings that we can modify. Compared with other networks designed specifically for a few targeted datasets, where most of the architecture settings are fixed and difficult to change, KPConv provides more flexibility in adjusting the network. This flexibility allowed us to better explore the conditions under which the network can improve the segmentation of PLOs.

Our contributions are summarized as follows:

1. We propose a method to improve the performance of deep learning networks for PLOs.
2. We design an experiment to determine the optimal kernel size for KPConv in semantic segmentation of PLOs and verify its effectiveness.
3. We introduce a post-processing method for semantically segmented PLO point clouds to achieve panoptic segmentation and accurate detection of PLOs.

## 2. Related Work

In this section, we review deep neural networks designed for understanding point cloud features, including tasks such as classification and segmentation.

### 2.1 Projection-based Methods

The main idea behind these methods is to use well-established Convolutional Neural Networks (CNN) that have been successfully applied to 2D image classification to address the classification of 3D point clouds. These methods first project a 3D shape into several 2D images, extract features from each view, and then combine those features to classify the point cloud. Aggregating multiple view-wise features into an overall representation presents a key challenge for these methods.

MVCNN Su et al. (2015) is such a pioneering work that achieved that. It generated 80 simulated photos from different angles of the target 3D point cloud object, used the same trained CNN to extract their features, and then composed those 80 views of features into a global descriptor by max-pooling layers. However, simple max-pooling only retains the maximum elements, which may discard a significant amount of 3D point information. Some other methods have been proposed to improve classification accuracy, such as using multi-resolution filtering CNN (Qi et al., 2016), adding one more grouping layer after pooling each view (Feng et al., 2018), using Graph Convolutional Network (GCN) on the projected images (Wei et al., 2020), and leveraging relation networks to exploit relationships among views (Yang and Wang, 2019).

### 2.2 Voxelization-based Methods

As mentioned in the previous section, simply projecting a 3D point cloud to 2D and applying 2D CNN models does not capture the 3D features well. Therefore, researchers turned to 3D CNN models that directly extract 3D features. The basic idea is to convert the original point cloud into voxels, which are 3D pixels, and then directly extend 2D CNN methods into 3D, to extract features from these 3D voxels.

Voxnet (Maturana and Scherer, 2015) and 3D ShapeNets (Wu et al., 2015) are two pioneering studies in those 3D CNN methods. They introduced volumetric occupancy grids as an intermediate format for dense point clouds and designed three-layer and two-layer 3D CNN models, respectively. However, their performance was limited by the increased computation and VRAM cost when voxelizing point clouds at higher resolutions, since the number of training parameters grows cubically.

Later, studies such as OctNet (Riegler et al., 2017) and O-CNN (Wang et al., 2017) used hierarchical grid structures to lower the computational cost. Octree is a geometric modelling technique that represents 3D objects with a tree structure where each parent node has eight children (Meagher, 1982). In the context of 3D tasks, this data structure divides a cube recursively into eight equal sub-cubes. Using an Octree for feature learning allows for more compact storage and faster computation while maintaining similar accuracy.

### 2.3 Point-based Methods

To better use the information from dense point clouds, many studies have designed network structures that take points as input directly, rather than relying on intermediate representations such as 2D images or 3D voxels.

**Point-wise MLP Methods** PointNet innovatively pioneered this field, it processes each point using a multilayer perceptron (MLP) while taking into account three key characteristics of point clouds: their unordered nature, the interactions among points, and robustness to transformations (Qi et al., 2017a). Since all the points have no order, their designed network used MLPs to read points disorderly. All those MLPs shared the parameters to learn the relation among all the points. Then, they added T-Net which allows the model to ignore differences in rotation and focus on the underlying shape of the point cloud. In the backbone of Pointnet, points are transformed into feature vectors of dimensions 64, 128, and 1024, and a max-pooling operation then aggregates those features into a global descriptor of length 1024. That global feature vector is used by a fully connected network for object classification or by a similar decoder structure for point segmentation.

One critical limitation of Pointnet is it cannot extract local features at different scales. To overcome this, PointNet++ was developed (Qi et al., 2017b). The key difference is that PointNet++ does not directly reduce point features from an N-dimensional space to a single dimension. Instead, it hierarchically extracts features, similar to the layer-by-layer feature extraction in convolutional neural networks (CNNs), where higher-level features are derived from groups of lower-level features. Unlike the sliding stride in traditional CNNs, PointNet++ uses the farthest point sampling (FPS) algorithm to generate receptive fields. This sampling method has been widely adopted in many point cloud deep learning methods and continues to perform well in recent studies such as PointASNL (Yan et al., 2020) and Point-NN Zhang et al. (2023).

**Graph-based Methods** Earlier methods focused mainly on the x, y, and z coordinates of each point. In contrast, graph-based methods aim to represent and learn from the relationships between points. In these methods, a point cloud is treated as a graph, where each point is a vertex and each pairwise relationship is represented as an edge.

Dynamic Graph CNN (DGCNN) is such a graph network constructed based on the k-nearest neighbours of each point (Wang et al., 2019). The term ”dynamic” indicates that the graph structure is updated in each layer as the point features change. A key component of DGCNN is the EdgeConv operation, which applies convolution to the edges of the graph, capturing local geometric structures of the point cloud. This approach of learning relationships from key points makes DGCNN particularly well suited for point clouds representing individual objects and indoor scenes.

**Convolution-based Methods** Different from the methods in projection-based and Voxelization-based categories, methods in this section use point convolution that directly operates on a point cloud instead of intermediate representations. KPConv is a representative model in this area, which designed a ’Point kernel’ to eliminate the traditional ’Grid kernel’ (Thomas et al., 2019). They first created a rigid kernel with fixed point positions by solving the optimization problem, and that kernel performed very well when giving spherical point domains. Then, they also trained their network by leaving the position of the kernel point trainable and named it deformable KPConv. Their experiment found that the deformable KPConv outperformed the rigid version on large and diverse datasets.

## 3. Method

KPConv performs well on a variety of datasets and yields results comparable to state-of-the-art methods (Roynard et al., 2018; Xiang et al., 2023). However, its performance on PLOs is notably weak. In this study, we investigate the reasons behind this limitation and explore ways to improve the KPConv network for better feature understanding of PLO.

We observe that the KPConv model trained with the default settings produces segmentation results with many false negatives across different parts of PLOs. In our tests, two nearly identical PLOs display distinct error patterns, one shows false negatives mainly at the top, while the other exhibits them at the bottom. That indicates that the issue is not a failure to understand the individual parts, but rather an issue to ’see’ the complete structure of the object. Additional evidence supporting this assumption comes from cases where some tree trunks were segmented as PLOs. That may occur because the model only captures the middle part of the tree trunk, which is similar to the middle portion of many PLOs.

Therefore, we adjust the size of the ball point kernel to increase its receptive field and modify the parameters to fit within VRAM limits. This architectural refinement allows the model to capture the entirety of PLOs more effectively, thereby improving semantic segmentation accuracy as shown in Figure 1.

Unlike panoptic segmentation, semantic segmentation does not include instance information, making it difficult to separate objects such as vegetation, vehicles, and buildings that are often in close proximity. However, for PLOs, there is common knowledge that gaps typically exist between them. Even when

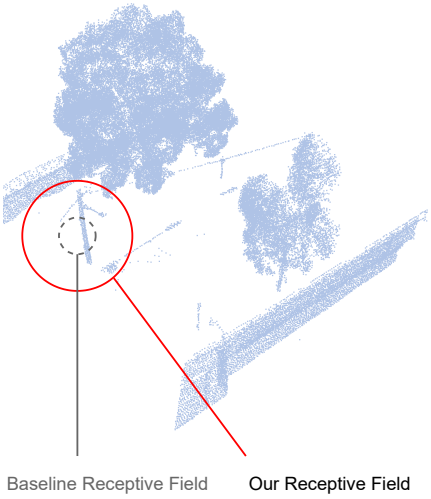


Figure 1. Example receptive fields of our method and baseline KPConv.

some PLOs are near trees or other roadside elements, the semantic segmentation results treat each PLO as a separate entity. This allows us to divide them into distinct instances using simple post-processing steps. To extract detailed instance information from our semantically segmented results, we employ connected-component labelling (He et al., 2017). This technique segments the point clouds of detected PLOs, enabling further analysis of instances. As an example, we estimate the height of each segmented instance by calculating their convex hulls.

## 4. Experiments and Results

### 4.1 Quantitative result on Parkville-3D dataset

We used the Parkville-3D dataset (Zhang et al., 2024) in this study because, compared with other datasets such as Paris-Lille-3D (Roynard et al., 2018) and Toronto-3D (Tan et al., 2020), it focuses more on PLOs and offers a greater variety of their shapes. The original dataset distinguishes among electrical poles, light poles and road signs. In our experiments, we combined those three types into a single ”Pole” category to better fit our method.

Table 1 presents our model achieved a 95.02% mIoU for PLOs, which is a significant improvement from 65.58%, the result of the default baseline. Comparative visualizations (Figure 2) highlight the classification errors of the original KPConv network at the attachments at the top and the base of PLOs as a result of the network’s previously smaller receptive field which failed to capture the full semantic scope of those structures. Our enhanced model significantly resolved this issue, providing a more comprehensive understanding of PLOs as cohesive units.

Moreover, the original baseline models offen misclassify tree trunks as poles, particularly in urban settings where utility poles often consist of unprocessed tree trunks. This resemblance poses a substantial challenge for deep learning models, leading to frequent misclassifications of orderly, plant-based structures as utility poles. Our model has markedly reduced these errors, demonstrating improved discrimination between natural tree trunks and man-made poles.

| First Kernel Radius | PLO IoU (%) |
|---|---|
| 3 | 65.580 |
| 6 | 85.657 |
| **9** | **95.020** |
| 12 | 94.410 |
| 15 | 93.821 |

Table 1. Semantic Segmentation Results for PLOs on the Parkville-3D Dataset: Comparison of KPConv with Kernel Size 3 (Default) and Kernel Size 9 (Optimized)

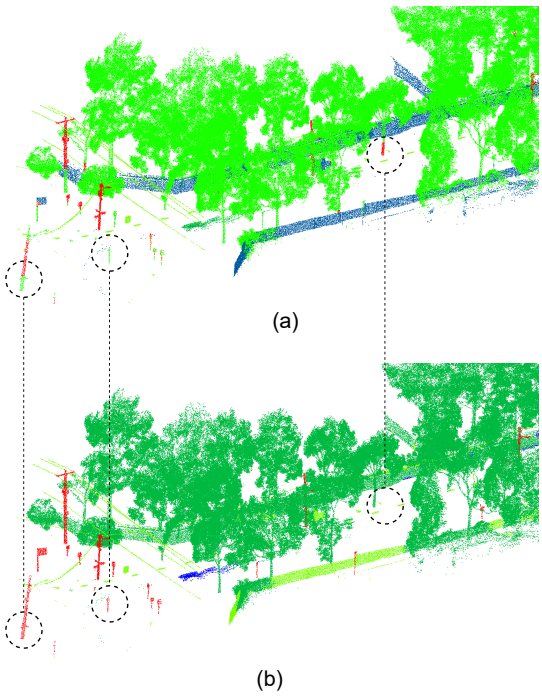


Figure 2. Comparison of semantic segmentation results for poles (highlighted in red) on Parkville-3D dataset: (a) KPConv, (b) proposed model.

### 4.2 Qualitative results

Domain shift is a common issue in deep learning models for point clouds, it refers to the situation where a model trained on one dataset performs noticeably worse when applied to a new point cloud environment (Luo et al., 2020). In our study, even though the training and testing sets are different portions of the Parkville-3D dataset, they still share many similarities. To further evaluate the performance and generalizability of our proposed method, we designed a qualitative experiment using an unlabelled point cloud dataset captured in another city within the greater Melbourne area as shown in Figure 3. This dataset contains more complex road assets, such as dense vegetation and a variety of PLO shapes, providing a more challenging

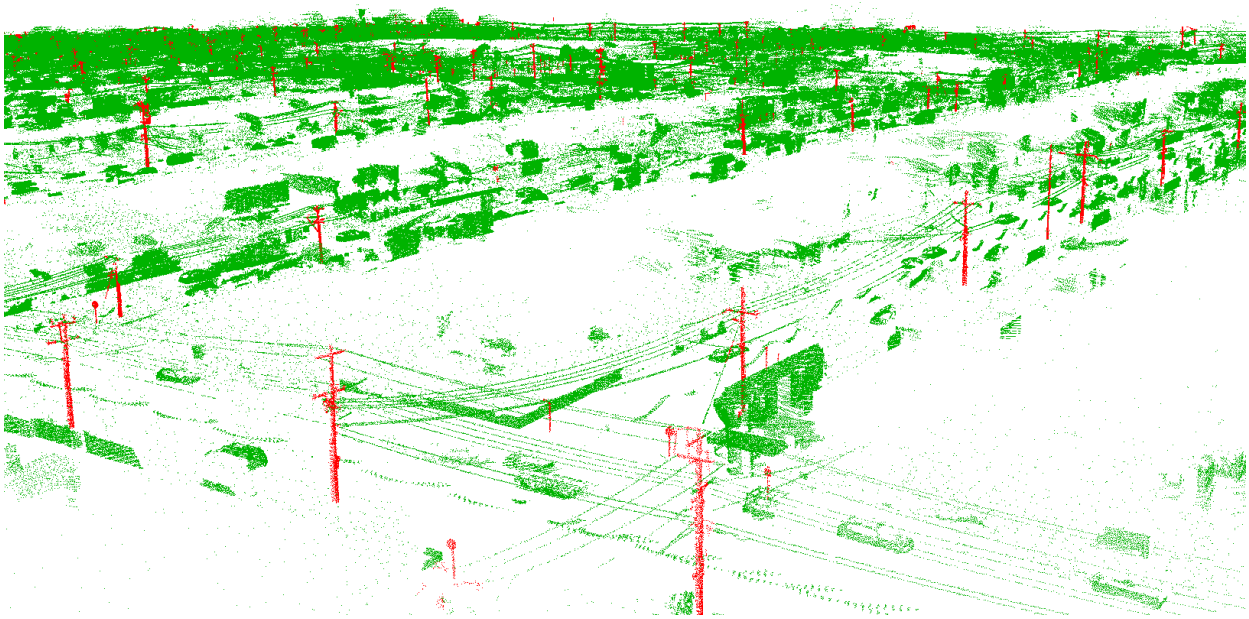

Figure 3. Qualitative Experimental Result (Down-sampling applied to other objects for better visualization of PLO point clouds).

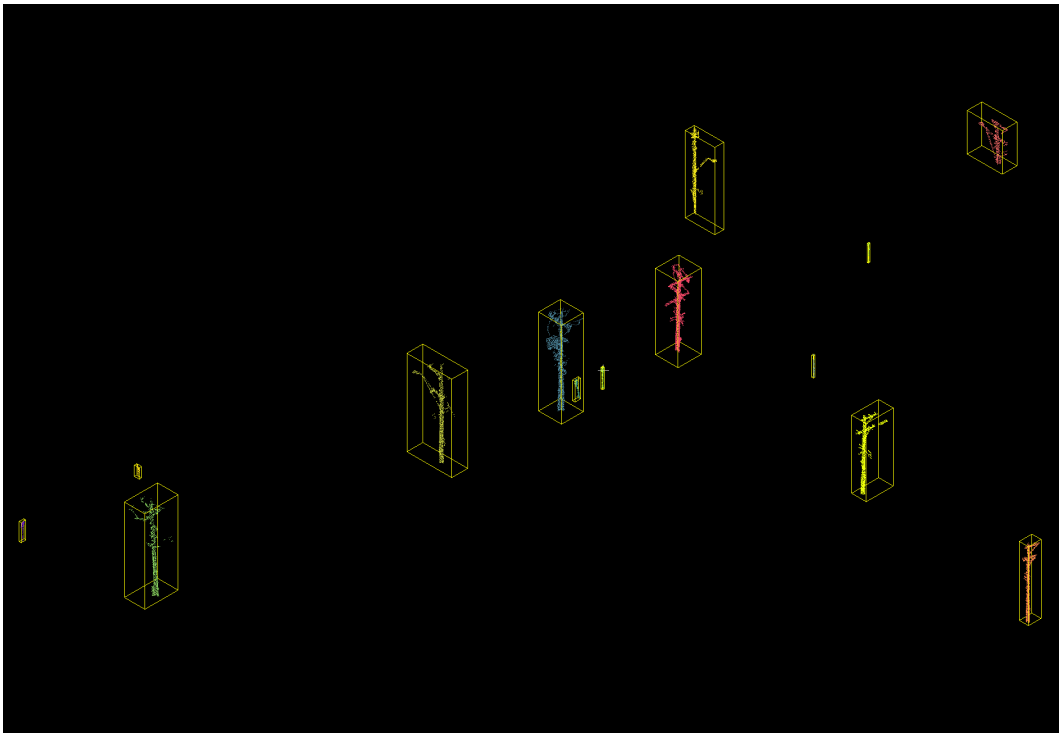

Figure 4. Instance segmentation results of connected-component labelling

scenario for our model.

Following semantic segmentation, we applied the Connected-component labelling algorithm (He et al., 2017) to the point clouds identified as belonging to the pole category, segmenting them into separate instances, as shown in figure 4. Next, we extracted the height of each PLO and used the center of each instance as its location. Because the point cloud is georeferenced, we can display those locations on a base map alongside reference results from satellite imagery, as shown in Figure 5. Out of 360 poles present, our method successfully detected 310, yielding an accuracy rate of 86.11%.

We analyzed the error cases and identified two main issues. Figure 6 demonstrates that when a PLO is surrounded by dense vegetation, the model sometimes fails to detect the PLO hidden among the thick branches and leaves. Figure 7 reveals that palm trees—which were absent from the training and validation sets—confuse the model, causing it to mistakenly segment the long, straight trunks of those palms as PLOs. One interesting observation is that even though our model has never encountered palm trees before, it tends to segment the top part as vegetation. In contrast, when faced with unseen man-made structures resembling PLOs, the model consistently segments

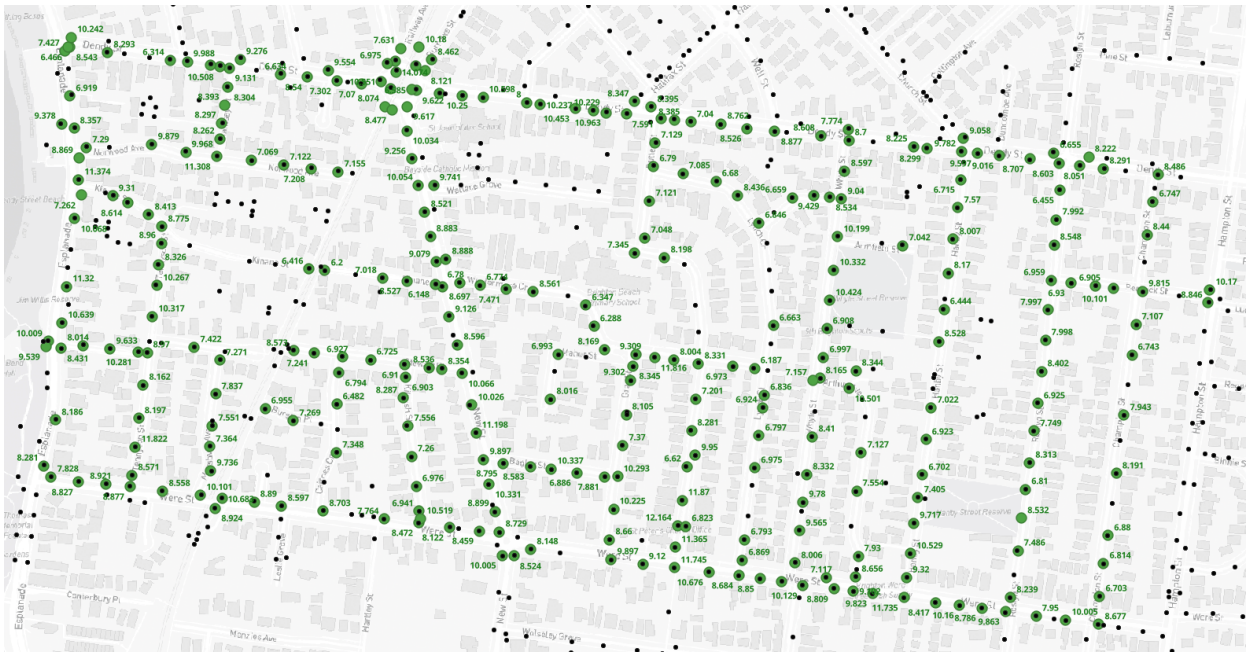

Figure 5. Extracted poles (in green) with their estimated height and reference pole location (in black) from satellite image on base map.

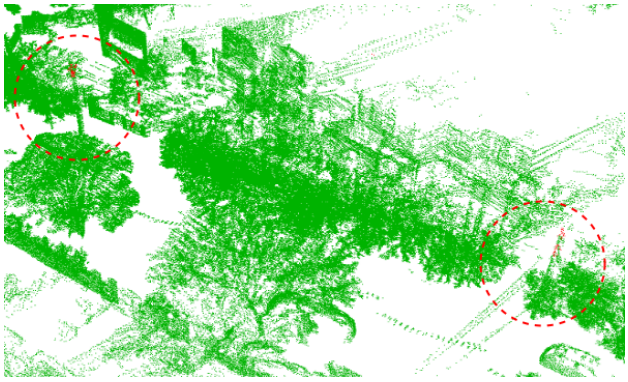

Figure 6. Example of undetected PLOs due to nearby dense vegetation (highlighted in red circle) in our qualitative experiment.

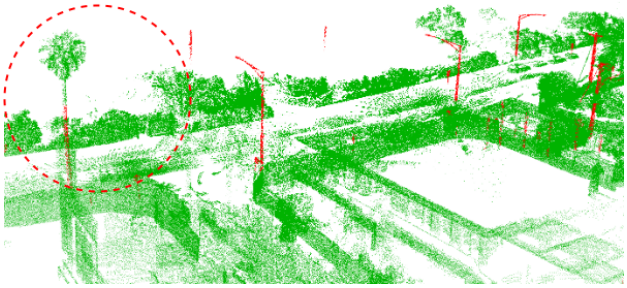

Figure 7. Example of undetected PLOs due to unseen vegetation (highlighted in red circle) in our qualitative experiment.

them as PLOs. This indicates that our model effectively understands the features of PLOs and, to a certain extent, demonstrates strong generalizability.

## 5. Discussions and Future Directions

Our experiments show that selecting an appropriate kernel size increases the receptive field and, in turn, improves semantic segmentation performance, especially for PLOs. This finding indicates that for a target-specific point cloud classification or segmentation task, tuning the kernel size according to the dimensions of the target data is an effective approach.

However, due to the limited availability of point cloud data, domain shift remains an issue when applying the trained model to unseen datasets. One solution is to expand the training data by capturing and labeling more point clouds in different environments, or by using synthetic point clouds while addressing the gap between synthetic and real data. Another approach is to investigate few-shot learning methods (Zhang et al., 2024) for point cloud classification and segmentation, which could achieve similar performance in new environments with only a few labelled examples. We believe those approaches offer promising directions to close the current research gap.

## 6. Conclusion

This paper presents a method to improve the accuracy of pole segmentation by focusing on optimizing the network’s receptive field. We also implement post-processing techniques that extract detailed information from the segmented point cloud, which enhances the model’s utility in urban mapping and infrastructure analysis. Experimental results indicate that our method outperforms existing approaches for semantic segmentation of PLOs. These findings demonstrate the potential of specialized deep learning models for addressing challenges in complex urban environments.

**Acknowledgment**

This research is supported by The University of Melbourne Graduate Research Scholarships and Research Computing Services.

**References**

Cabo, C., Ordoñez, C., García-Cortés, S., Martínez, J., 2014. An algorithm for automatic detection of pole-like street furniture objects from Mobile Laser Scanner point clouds. *ISPRS Journal of Photogrammetry and Remote Sensing*, 87, 47–56. Publisher: Elsevier.

Dong, H., Chen, X., Särkkä, S., Stachniss, C., 2023. Online pole segmentation on range images for long-term LiDAR localization in urban environments. *Robotics and Autonomous Systems*, 159, 104283.

Feng, Y., Zhang, Z., Zhao, X., Ji, R., Gao, Y., 2018. Gvcnn: Group-view convolutional neural networks for 3d shape recognition. *Proceedings of the IEEE conference on computer vision and pattern recognition*, 264–272.

Gholampooryazdi, B., Hämmäinen, H., Vijay, S., Savisalo, A., 2017. Scenario planning for 5G light poles in smart cities. *2017 Internet of Things Business Models, Users, and Networks*, 1–7.

He, L., Ren, X., Gao, Q., Zhao, X., Yao, B., Chao, Y., 2017. The connected-component labeling problem: A review of state-of-the-art algorithms. *Pattern Recognition*, 70, 25–43. Publisher: Elsevier.

Luo, H., Khoshelham, K., Fang, L., Chen, C., 2020. Unsupervised scene adaptation for semantic segmentation of urban mobile laser scanning point clouds. *ISPRS Journal of Photogrammetry and Remote Sensing*, 169, 253–267. Publisher: Elsevier.

Maturana, D., Scherer, S., 2015. Voxnet: A 3d convolutional neural network for real-time object recognition. *2015 IEEE/RSJ international conference on intelligent robots and systems (IROS)*, IEEE, 922–928.

Meagher, D., 1982. Geometric modeling using octree encoding. *Computer graphics and image processing*, 19(2), 129–147. Publisher: Elsevier.

Qi, C. R., Su, H., Mo, K., Guibas, L. J., 2017a. Pointnet: Deep learning on point sets for 3d classification and segmentation. *Proceedings of the IEEE conference on computer vision and pattern recognition*, 652–660.

Qi, C. R., Su, H., Nießner, M., Dai, A., Yan, M., Guibas, L. J., 2016. Volumetric and multi-view cnns for object classification on 3d data. *Proceedings of the IEEE conference on computer vision and pattern recognition*, 5648–5656.

Qi, C. R., Yi, L., Su, H., Guibas, L. J., 2017b. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. *Advances in neural information processing systems*, 30.

Riegler, G., Osman Ulusoy, A., Geiger, A., 2017. Octnet: Learning deep 3d representations at high resolutions. *Proceedings of the IEEE conference on computer vision and pattern recognition*, 3577–3586.

Roynard, X., Deschaud, J.-E., Goulette, F., 2018. 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. Publisher: SAGE Publications Sage UK: London, England.

Su, H., Maji, S., Kalogerakis, E., Learned-Miller, E., 2015. Multi-view convolutional neural networks for 3d shape recognition. *Proceedings of the IEEE international conference on computer vision*, 945–953.

Tan, W., Qin, N., Ma, L., Li, Y., Du, J., Cai, G., Yang, K., Li, J., 2020. Toronto-3D: A large-scale mobile LiDAR dataset for semantic segmentation of urban roadways. *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops*, 202–203.

Thomas, H., Qi, C. R., Deschaud, J.-E., Marcotegui, B., Goulette, F., Guibas, L. J., 2019. Kpconv: Flexible and deformable convolution for point clouds. *Proceedings of the IEEE/CVF international conference on computer vision*, 6411–6420.

Wang, P.-S., Liu, Y., Guo, Y.-X., Sun, C.-Y., Tong, X., 2017. O-cnn: Octree-based convolutional neural networks for 3d shape analysis. *ACM Transactions On Graphics (TOG)*, 36(4), 1–11. Publisher: ACM New York, NY, USA.

Wang, Y., Sun, Y., Liu, Z., Sarma, S. E., Bronstein, M. M., Solomon, J. M., 2019. Dynamic graph cnn for learning on point clouds. *ACM Transactions on Graphics (tog)*, 38(5), 1–12. Publisher: ACM New York, NY, USA.

Wei, X., Yu, R., Sun, J., 2020. View-gcn: View-based graph convolutional network for 3d shape analysis. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 1850–1859.

Wu, Z., Song, S., Khosla, A., Yu, F., Zhang, L., Tang, X., Xiao, J., 2015. 3d shapenets: A deep representation for volumetric shapes. *Proceedings of the IEEE conference on computer vision and pattern recognition*, 1912–1920.

Xiang, B., Yue, Y., Peters, T., Schindler, K., 2023. A Review of panoptic segmentation for mobile mapping point clouds. *ISPRS Journal of Photogrammetry and Remote Sensing*, 203, 373–391. https://www.sciencedirect.com/science/article/pii/S092427162300223X.

Yan, X., Zheng, C., Li, Z., Wang, S., Cui, S., 2020. Pointasnl: Robust point clouds processing using nonlocal neural networks with adaptive sampling. *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 5589–5598.

Yang, Z., Wang, L., 2019. Learning relationships for multi-view 3D object recognition. *Proceedings of the IEEE/CVF international conference on computer vision*, 7505–7514.

Zhang, R., Wang, L., Wang, Y., Gao, P., Li, H., Shi, J., 2023. Parameter is not all you need: Starting from non-parametric networks for 3d point cloud analysis. *arXiv preprint arXiv:2303.08134*.

Zhang, Z., Khoshelham, K., Shojaei, D., 2024. Pole-NN: Few-Shot Classification of Pole-Like Objects in Lidar Point Clouds. *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 10, 333–340. Publisher: Copernicus Publications Göttingen, Germany.