AN ANALYSIS OF NEIGHBOURHOOD TYPES FOR POINTNET++ IN SEMANTIC SEGMENTATION OF AIRBORNE LASER SCANNING DATA

Z. Akbulut [1]*, S. Ozdemir [1], F. Karsli [2], M. Dihkan [2]

[1] Gumushane University, Faculty of Engineering and Natural Sciences, Department of Geomatics Engineering, Gumushane, Turkey – (zeynepakbulut, samed.ozdemir)@gumushane.edu.tr
[2] Karadeniz Technical University, Faculty of Engineering, Department of Geomatics Engineering, Trabzon, Turkey – (fkarsli, mdihkan)@ktu.edu.tr

KEY WORDS: Airborne Laser Scanning, Deep Neural Network, Neighbourhood Types, PointNet++, Semantic Segmentation.

ABSTRACT:

The objective of the study is to conduct a comprehensive examination of how different neighbourhood types, namely spherical, cylindrical, and k-nearest neighbour (kNN), influence the feature extraction capabilities of the PointNet++ architecture in the semantic segmentation of Airborne Laser Scanning (ALS) point clouds. Two datasets are utilized for semantic segmentation analysis: the Dayton Annotated LiDAR Earth Scan (DALES) and the ISPRS 3D Semantic Labelling Benchmark datasets. In the experiments, the kNN method exhibited approximately 1% higher accuracy in weighted mean F1 and intersection over union (IoU) metrics compared to the spherical and cylindrical neighbourhood types on the DALES dataset. However, in the generalization experiment conducted on the ISPRS dataset, the spherical neighbourhood achieved the best results in these metrics, outperforming the cylindrical neighbourhood by a small margin. Notably, the kNN method was the least accurate, with a decrease in accuracy of approximately 1% in both weighted mean IoU and F1 scores. These findings suggest that the features extracted from spherical and cylindrical neighbourhood types are more generalizable compared to those from the kNN method.

1. INTRODUCTION

Airborne Laser Scanning (ALS) technique plays a crucial role in gathering data about real-world environments. The utilization of point cloud data obtained from ALS extends across many different fields, covering mapping and surveying, 3D object detection, forestry, disaster monitoring, and the preservation of cultural heritage (Nong et al., 2023). However, due to the complexity of urban scenes, assigning each 3D point in the irregularly distributed point cloud to a semantic class is a challenging task (Niemeyer et al., 2014).

Following advancements in computer systems, 3D deep neural networks (DNNs) have increasingly been employed for 3D semantic segmentation task. Research on deep learning-based semantic segmentation of point clouds is broadly classified into three categories by Lin et al. (2022), depending on the structure of the input data: projection-based, voxel-based, and point-based approaches. Guo et al., (2021), point out that converting point clouds to uniform data formats for integration with Convolutional Neural Networks (CNNs) might introduce additional computational overhead and lead to a notable loss of information. Additionally, Öngün and Temizel (2021) note that point clouds are the predominant data type in the three-dimensional perception of reality, extensively utilized across various domains such as 3D scanning, robotics, autonomous vehicles, and face recognition. Consequently, without the need to convert into any regular data formats, architectures have been developed that can directly handle 3D point clouds in their original form.

As highlighted by Luo et al. (2022), most pointwise semantic segmentation methods typically follow a process flow that includes sampling points, searching for neighbours, aggregating features, and finally, classification. In this study, we specifically concentrated on the neighbourhood selection stage. The PointNet++ architecture, introduced by Qi et al. (2017b) recognized as a pioneering point-wise method, selects neighbouring points through its set abstraction layer, which involves both sampling and grouping operations. The primary operation for identifying neighbouring points is the grouping mechanism. In this process, after representative points are sampled using Farthest Point Sampling (FPS), PointNet++ groups points by employing a spherical neighbourhood around each sampled point to effectively capture the local spatial relationships within the point cloud. According to Thomas et al. (2018), for classifying 3D points, the two predominant neighbourhood definitions employed are the spherical neighbourhood and the k-nearest neighbours (kNN). Additionally, they note that a third definition, often applied in airborne LiDAR datasets, is the cylindrical neighbourhood.

The motivation of the study is to conduct an in-depth analysis of the impact of different neighbourhood types on the feature extraction capabilities in semantic segmentation applications carried out with the PointNet++ (Qi et al., 2017b) for ALS point clouds. We have restructured the four layers of PointNet++ to construct feature vectors using different neighbourhood types. We used two datasets to investigate the effects of different neighbourhood types on the performance of PointNet++. The ISPRS 3D Semantic Labelling Benchmark dataset, a widely utilized benchmarking dataset for point cloud classification (Niemeyer et al., 2014), serves as one of the two datasets employed. The other point cloud dataset used in this study is the Dayton Annotated LiDAR Earth Scan (DALES) (Varney et al., 2020), a large-scale aerial LiDAR point cloud specifically created for semantic segmentation tasks. The evaluation of three neighbourhood types was conducted using both qualitative and

* Corresponding author: zeynepakbulut@gumushane.edu.tr

https://doi.org/10.5194/isprs-archives-XLVIII-4-W9-2024-7-2024

quantitative results, focusing on class-based accuracies as well as overall accuracies.

2. RELATED WORK

In the field of computer vision, a variety of point-based architectures has been established for the classification of 3D point cloud. These architectures employ different approaches for neighbour selection, with some utilizing kNN and others adopting spherical neighbourhood types. PointNet++ (Qi et al., 2017b) architecture employs a spherical neighbourhood to effectively gather local sets of points based on spatial proximity, facilitating the capture of local structures within the data. SpiderCNN employs the kNN approach for determining local neighbourhood characteristics, as opposed to using a radius-based method (Xu et al., 2018). In PointASNL, Yan et al. (2020) initially conduct a k-NN query to group the neighbours of sampled points, and then apply a general self-attention mechanism for the updating of group features. Zhao et al. (2019) employs kNN method to group neighbour points in their PointWeb architecture. For neighbor selection, Point Trasformer architecture (Zhao et al., 2021) utilizes kNN method to select neighbour points. They also investigate the setting of the number of neighbours, denoted as k, which is used in determining the local neighbourhood around each point. The best performance was achieved when k was set to 16. They noted that smaller neighbourhoods might not provide sufficient context for accurate predictions, while larger neighbourhoods could introduce excessive noise, lowering the accuracy of the model. Thomas et al. (2019) prefer neighbourhoods defined by radius rather than using kNN in their KPConv architecture. Thomas et al. (2018) demonstrated that manually crafted 3D point features yield superior representations when calculated using neighbourhoods defined by radius, rather than through kNN, a finding further discussed by Thomas et al. (2019). In SO-Net, Li et al. (2018) achieved systematic adjustment of the receptive field of the network by implementing a point-to-node k nearest neighbour search. Wu et al. (2019) describe that the input features are derived from the kNN method in PointConv. Turgut and Dutagaci (2024) propose a module designed to adjust the radii for each center point, in contrast to the conventional method of using a fixed radius specific to each layer for grouping.

In the domain of point-based classification of ALS data, Li et al. (2021), Chen et al. (2021) and Winiwarter et al. (2019) adapted PointNet++ for the semantic segmentation of ALS point clouds using spherical neighbourhoods. Meanwhile, Kada and Kuramin (2021) introduced an additional branch in the first of the four set abstraction layers of PointNet++, designed to compute extra features from cylindrical neighbourhoods. Then, at the end of the first set abstraction layer, they concatenated features from spherical and cylindrical neighbourhoods. They noted that the inclusion of cylindrical neighbourhoods in the first abstraction layer of PointNet++ further improved its performance, particularly in distinguishing between facades and roof points. Inspired by Kada and Kuramin (2021), we have expanded this concept by individually incorporating three types of neighbourhoods into each layer of the network, not just the first.

3. METHODOLOGY

3.1 Neighbourhood Types

Regarding a specific point P, the spherical neighbourhood consists of points located within a predetermined radius from P, while the k-nearest neighbours encompass a set number of points nearest to P. Additionally, a third definition often applied in airborne LiDAR dataset is the cylindrical neighbourhood. This includes points within a certain radius from P on a 2D projection of the cloud, commonly on the horizontal plane (Thomas et al., 2018). Figure 1 provides a visual representation of three distinct types of neighbourhoods and how each neighbourhood interacts with the point cloud.

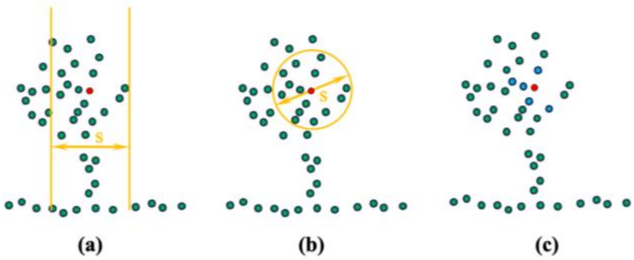

Figure 1. The types of neighbourhoods, adapted from Guo et al. (2015) and Ozdemir (2021), include: a) cylindrical neighbourhood, b) spherical neighbourhood, and c) k-nearest neighbours.

There is a range of opinions concerning the application of kNN and spherical neighbourhood search methods. Some researchers support the use of KNN for its benefits, whereas others favor the spherical neighbourhood search method, each emphasizing the strengths of these approaches. Hermosilla et al. (2018) noted that kNN method lacks resilience in settings with non-uniform sampling, as introducing additional points into a spatial region diminishes the k nearest neighbours to a limited area surrounding each point. This results in the capture of features that differ from those on which the kernel was initially trained, meaning the area of focus contracts in regions with high point density and expands in less dense areas. Hackel et al. (2016) mentioned that from a theoretical standpoint, radius search is ideally suited for point clouds with uniform point density, as it matches a constant geometric scale in object space. However, this method becomes less practical with significant variations in point density. Consequently, due to these constraints, Hackel et al. (2016) expressed a preference for the kNN search, which acts as an effective approximation to a radius that adjusts to the fluctuating densities within the point cloud. According to Thomas et al. (2018), in contrast to kNN, a spherical neighbourhood aligns with a specific, unchanging section of space. This attribute is essential for providing a more stable geometric context to the features. Nonetheless, within this consistent spatial area, the count of points may fluctuate depending on the density of the point cloud. They highlighted that in spherical neighbourhoods, a small-scale area may lack a sufficient number of points for an accurate description, whereas a larger-scale area is capable of providing the necessary information. Qi et al. (2017b) recommend using the spherical method because it ensures fixed-size local regions through its radius, which leads to the extraction of more generalizable features across different spaces.

Considering the relative advantages and disadvantages of these methods, we examined their effects on the semantic segmentation of 3D point clouds under the same conditions using PointNet++. The impact of each of the three methods on the generalization performance of the architecture has also been evaluated.

3.2 Review of PointNet++

In point-wise deep learning domain for semantic segmentation, PointNet was introduced by Qi et al. (2017a) to directly classify point clouds, but inherent to its design, PointNet does not encompass the local structures. Therefore, PointNet++, developed by Qi et al. (2017b), addresses the limitations of PointNet by iteratively applying its architecture to local regions, marking a foundational advancement in point-wise deep neural networks. Neighbourhoods, formed around center points chosen to cover the entire input point cloud, represent local regions and enable the division of the point cloud into overlapping sub-regions. PointNet++ uses the FPS method to select center points from the input point cloud, defining local regions as the neighbouring points within a specified radius around these centers, ensuring even coverage across the entire point cloud. Its hierarchical feature learning architecture is specifically designed for tasks like semantic segmentation and classification, as illustrated in Figure 2.

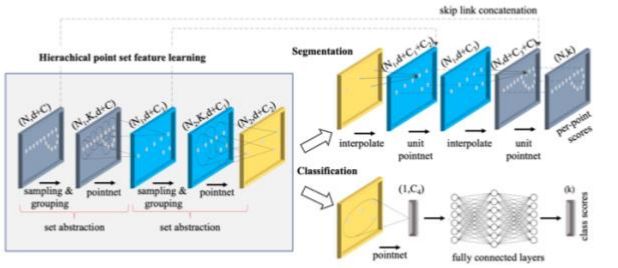

Figure 2. An example of PointNet++ architecture with two set abstraction layers, adapted from Qi et al. (2017b).

In PointNet++, a spherical neighbourhood is used to group points. The radius values for the four multi-layer perceptron (MLP) layers are set at doubled increments, starting from 0.1. Due to the unsuitability of directly applying standard radius values to ALS point clouds, we employed radius values of 1, 2, 4, and 8 meters, considering the number of points to be sampled within the radius and the point density. We randomly sampled 32 points within each neighbourhood.

4. EXPERIMENTAL RESULTS

In the study, three experiments were carried out using spherical, kNN, and cylindrical neighbourhoods only employing X, Y and Z coordinates on the PointNet++ architecture to highlight the advantages and limitations of each neighbourhood type.

4.1 Parameter Selection for PointNet++

In the study, we utilized and modified the Pytorch implementation of PointNet++ developed by Yan (2019). This implementation sets itself apart from the standard PointNet++ by using a sample rate to divide all input point clouds into fixed block sizes, enabling the architecture to observe almost all the data. We established the block sizes at 32m×32m for both spherical and cylindrical neighbourhood searches, considering the radius values and center points, to ensure as much as possible an overlapping receptive field. For both spherical and cylindrical neighbourhoods, we utilized radius values of 1, 2, 4, and 8 meters. The number of points selected from within the neighbourhood was fixed at 32. In cases where a neighbourhood has fewer points than required, the points were duplicated to achieve desired number of points. For the kNN approach, the nearest 32 points were utilized. In all applications, the number of epochs was set to 32. For testing, we choose the best model of each experiment. We configured the batch size to 8 and set the learning rate at 0.001. The number of center points was maintained as in the original architecture, starting with 1024 in the first layer and reducing by a factor of 1/4 in each subsequent layer. The dimensions of the layers were kept the same as in the original architecture.

To assess the performance of each neighbourhood type in handling sparsely sampled point clouds and their ability to generalize across different datasets, we conducted training using the DALES dataset and tested the trained model on both the DALES and ISPRS datasets. In the study, all applications were carried out on a computer configured with an Intel Core i5-11400F CPU, 24 GB of RAM, and an RTX 3060 12GB GPU, using Pytorch 2.0.

4.2 Datasets

In this study, DALES and ISPRS 3D Semantic Labelling Benchmark datasets were used. The DALES dataset is a large-scale ALS point cloud, consisting of 40-point cloud tiles, each spanning an area of 500×500 meters and has a density of approximately 50 pts/m^2 (Varney et al., 2020). The DALES dataset has nine classes: unknown, ground, vegetation, car, truck, power line, fence, pole, and building (Figure 3a). The ISPRS dataset (Niemeyer et al., 2014) is a widely utilized in point cloud classification research. The training subset of the dataset has 753,876 points with an average point density of 7.3 pts/m^2, and the test subset has 411,722 points with an average point density of 4.8 pts/m^2. ISPRS dataset has nine categories: power lines, low vegetation, impervious surfaces, cars, fences, roofs, facades, shrubs, and trees (Figure 3b).

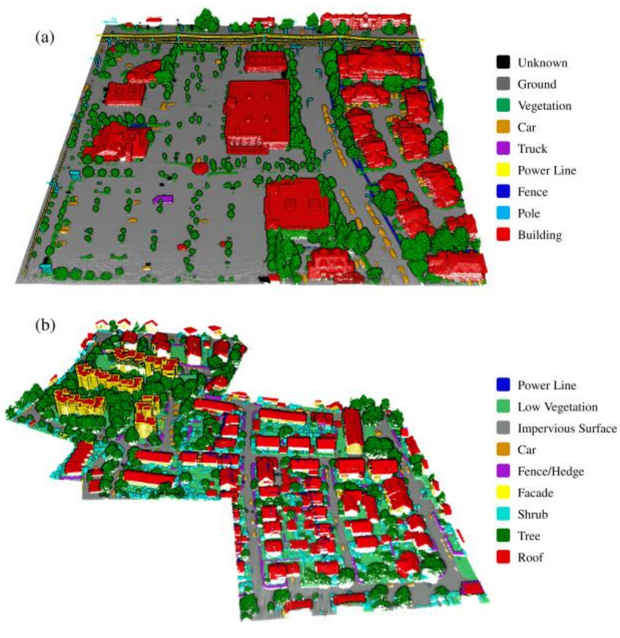

Figure 3. Datasets used in the study. A sample area of DALES dataset (a) and ISPRS Vaihingen Dataset (b).

4.3 Evaluation Metrics

In the study, the performance of semantic segmentation of 3D point clouds was evaluated using F1 score, Intersection over Union (IoU), and Overall Accuracy (OA) metrics. F1 and IoU values were calculated both per class and as a weighted mean (Equation 1).

$$\mathrm{mIoU} = \frac{\sum_{i=1}^{N} \mathrm{w_i IoU_i}}{\mathrm{N}} \tag{1}$$

Here, N represents the number of classes, and w denotes the class weights. Precision, Recall, F1-score, and overall accuracy were calculated using the following equations, from Equation 2 to Equation 5 respectively.

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{2}$$

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{3}$$

$$\text{F1} = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \tag{4}$$

$$\text{OA} = \frac{\text{TP} + \text{TN}}{(\text{TP} + \text{FN} + \text{FP} + \text{TN})} \tag{5}$$

Where TP represents the true positives, FP stands for false positives, TN denotes true negatives, and FN signifies false negatives for each class.

4.4 Results and Discussion

In the DALES dataset, all three neighbourhood methods demonstrated high performance across most classes, with the exception of trucks, fences, and pole classes. It has been observed that the kNN method is approximately 1% more accurate in terms of OA, weighted mIoU, and F1 metrics. Among these three methods, the kNN method resulted in higher accuracies across all classes, followed closely by the spherical neighbourhood method in terms of achieving the next highest accuracies. The cylindrical neighbourhood, although closely following the spherical in performance, had slight decreases in F1-score and IoU except for the poles class, as shown in Table 1.

Class Names	Spherical		Cylindrical		kNN	
	F1	IoU	F1	IoU	F1	IoU
Unknown	5.701	2.934	5.186	2.662	**19.324**	**10.695**
Ground	96.860	93.910	96.781	93.763	**97.348**	**94.833**
Vegetation	93.674	88.101	93.513	87.816	**94.237**	**89.102**
Cars	76.304	61.687	73.272	57.818	**81.352**	**68.566**
Trucks	26.793	15.469	21.820	12.246	**31.480**	**18.680**
Powerlines	91.653	84.591	91.407	84.174	**92.816**	**86.595**
Fences	57.389	40.241	54.028	37.012	**61.065**	**43.952**
Poles	60.327	43.191	60.938	43.821	**64.684**	**47.803**
Buildings	95.070	90.603	94.901	90.296	**96.117**	**92.524**
OA (%)	94.849		94.700		**95.529**	
Weighted mIoU(%)	94.677	90.776	94.512	90.522	**95.409**	**91.958**

Table 1. The results of Spherical, Cylindrical, and kNN neighbourhood types on DALES test dataset.

The kNN approach was superior in handling trucks, fences, and poles classes, but showed marginal improvement in ground and vegetation classes compared to the other two methods. The flexibility of the kNN approach in adjusting to local point density seems beneficial for distinguishing complex structures. Figure 4 illustrates an example of DALES test dataset results for three neighbourhood types.

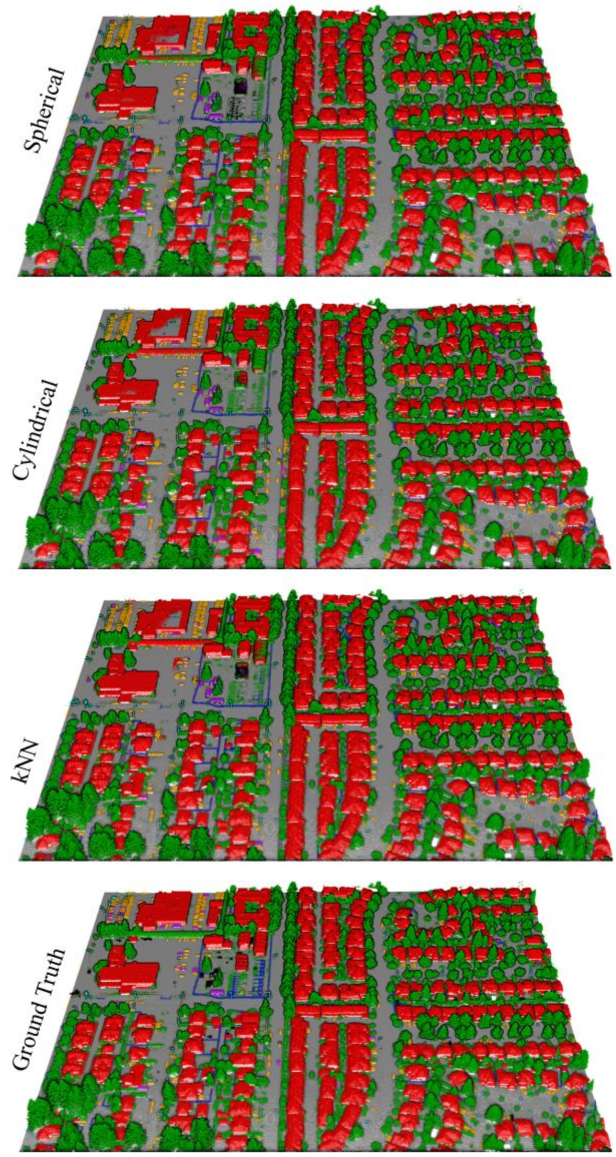

Figure 4. A visual comparison of spherical, cylindrical, and kNN methods on a sample tile from the DALES test dataset.

In Figure 4, all three neighbourhood types struggled with the segmentation of the roof of a large building. This issue may occur if the block size selected is insufficient to cover the building along with its surrounding ground information. Especially, the misclassification is more pronounced with the cylindrical neighbourhood compared to the spherical and KNN. Additionally, the spherical neighbourhood type was significantly more accurate in classifying trees near the large building at the upper left corner of the image, as opposed to the other two neighbourhood types. In general, kNN and spherical neighbourhood types has less confusion on the vegetation class compared to cylindrical. Contrary to the relatively low accuracy on the DALES dataset, the cylindrical neighbourhood type was relatively successful in classifying the ground class, whereas the spherical and kNN types confused small sections of the ground as buildings. We found that all neighbourhood types commonly confused the less represented car and truck classes with each other and with buildings. Such confusions typically occur when the two classes are in close proximity.

https://doi.org/10.5194/isprs-archives-XLVIII-4-W9-2024-7-2024

Upon examining the confusion matrix, it was observed that the cylindrical neighbourhood method resulted in the least confusion for the ground and powerline classes (Figure 5). Conversely, the kNN method showed superior performance in identifying poles and buildings. Specifically, the performance of kNN in classifying buildings was approximately 3% higher compared to both cylindrical and spherical neighbourhood methods. It was noted that the cylindrical neighbourhood method exhibited approximately 2% higher error rates compared to the other two methods in vegetation class.

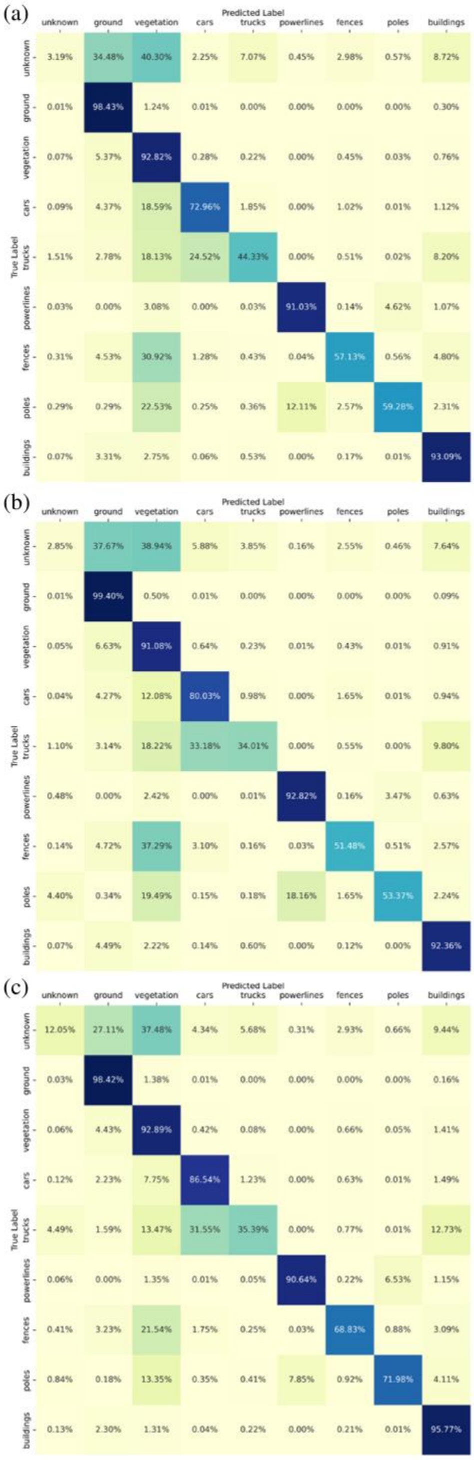

Figure 5. Confusion matrix of the neighbourhood types on all DALES test dataset. a) spherical b) cylindrical c) kNN

To assess the generalization performance of the architecture, it is essential to test it on datasets with varying characteristics. For this purpose, the ISPRS dataset, which contrasts the DALES dataset with its lower point density, presented a different challenge. After class label matching, the model trained on the DALES dataset was evaluated on the ISPRS test dataset. Upon reviewing the results, the weighted mIoU and F1 metric findings indicate that the spherical neighbourhood method performed better in generalization. The results from the ISPRS test are presented in Table 2.

Class Names	Spherical		Cylindrical		kNN	
	F1	IoU	F1	IoU	F1	IoU
Ground	94.017	88.709	**94.270**	**89.161**	93.815	88.351
Vegetation	**78.046**	**63.997**	75.475	60.610	74.251	59.048
Cars	**63.056**	**46.045**	59.312	42.159	57.983	40.828
Powerlines	39.865	24.895	**47.209**	**30.898**	46.769	30.521
Fences	**5.025**	**2.577**	1.668	0.841	4.048	2.066
Buildings	89.063	80.282	**89.613**	**81.181**	89.271	80.621
Accuracy (%)	87.370		**87.756**		87.067	
Weighted mIoU(%)	**87.541**	**79.742**	87.248	79.547	86.722	78.702

Table 2. The results of the generalization capability on ISPRS test dataset.

In Table 2, the cylindrical neighbourhood achieved results very close to those of the spherical method. The spherical neighbourhood method continued to show strong performance with the ground, vegetation, and buildings classes, yet its effectiveness was significantly lower for the fences and powerline classes. The cylindrical neighbourhood showed slight improvements in powerlines and buildings over the spherical neighbourhood but underperformed in fences and cars classes. Especially, in vegetation class, spherical neighbourhood method achieved a significant accuracy gain compared to other ones. The results suggest the cylindrical neighbourhoods may help with linear features like powerlines, buildings, and ground classes.

However, despite achieving the highest accuracies on the DALES test dataset, the kNN method was found to yield the lowest accuracies on the ISPRS dataset. The kNN method had comparable or slightly lower performance than spherical in most classes and did not offer significant gains in any specific class. This indicates that while adaptable to local point densities, did not demonstrate a significant improvement in segmentation accuracy, suggesting that in less dense environments, its advantages might be less pronounced. This situation serves as an example of Qi et al. (2017b)'s recommendation to employ the spherical method, which guarantees fixed-size local regions through its radius, thereby facilitating the extraction of more generalizable features across various spaces, as demonstrated in the case of ALS point clouds. The results for the ISPRS test dataset are displayed in Figure 6.

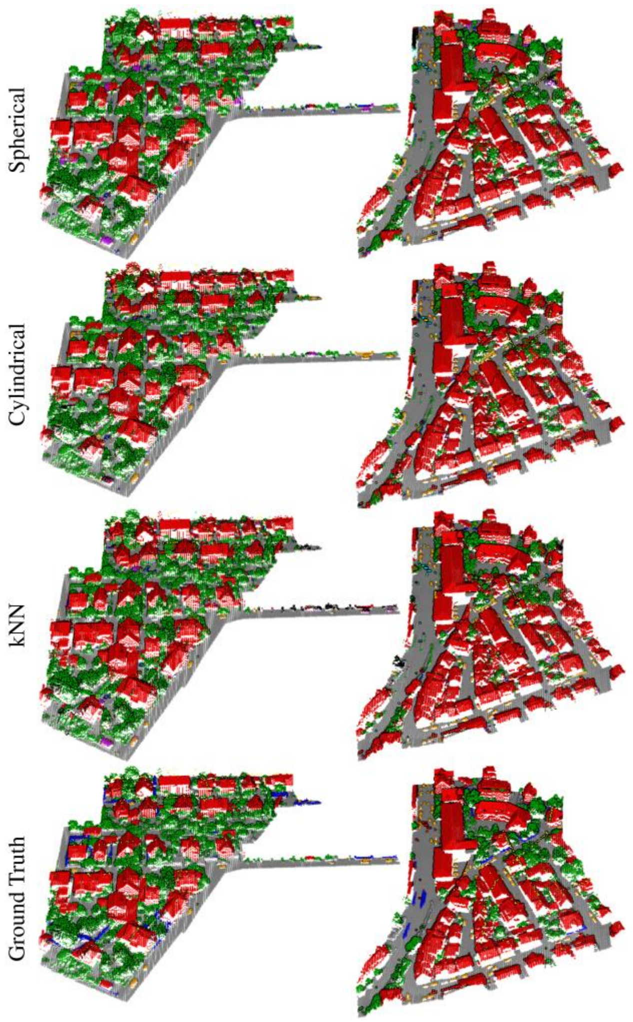

Figure 6. A visual comparison of spherical, cylindrical, and kNN methods on ISPRS test dataset.

Considering the point density differences, it is evident that the adaptability of kNN is more beneficial in the denser DALES dataset, helping it better handle the complexity of objects and irregular shapes. The consistent behaviour of spherical and cylindrical neighbourhoods across both datasets, despite the differences in point density, indicates inherent characteristics of these neighbourhood types that are somewhat independent of dataset density. Higher point density of the DALES dataset likely provides a more informative context for feature extraction, which may increase the effectiveness of the chosen neighbourhood type. However, neighbourhood parameterization is another aspect to consider, given the variations in point density. On one hand, while spherical and cylindrical neighbourhood types offer more stable neighbourhood definitions, they first need to be accurately parameterized for the selected dataset and task, which involves selecting the appropriate radius. On the other hand, kNN is easier to parameterize compared to the other two methods and offers similar, if not better, performance on a homogeneous dataset.

5. CONCLUSION

In the study, spherical, cylindrical, and kNN neighbourhood types were compared in the PointNet++ architecture for the semantic segmentation of ALS data. The research attempts to enhance the extraction of representative feature vectors by examining these neighbourhood types within the PointNet++ architecture. All three neighbourhood types achieved remarkable results in the semantic segmentation of the ALS datasets. While the kNN method showed better results on the DALES dataset, it slightly fell behind in generalization to the ISPRS dataset. Spherical and cylindrical neighbourhood types had slightly lower accuracy than kNN on the DALES dataset, but they offered higher generalization ability. The conducted study provides additional perspectives on the adaptability of PointNet++ in ALS data processing. While these neighbourhoods demonstrated varying degrees of effectiveness, their performance was highly influenced by the characteristics of the dataset, particularly the point density.

This observation suggests that dataset characteristics, along with architectural design, should be considered in the neighbourhood type selection stage. In light of these findings, future research in this area should focus on developing strategies for more advanced adaptations of neighbourhood types.

ACKNOWLEDGEMENTS

The authors wish to extend their gratitude to the publishers for providing the datasets and architecture codes.

REFERENCES

Chen, Y., Liu, G., Xu, Y., Pan, P., Xing, Y., 2021. PointNet++ Network Architecture with Individual Point Level and Global Features on Centroid for ALS Point Cloud Classification. Remote Sensing, 13(3), Article 3. https://doi.org/10.3390/rs13030472

Guo, B., Huang, X., Zhang, F., Sohn, G., 2015. Classification of Airborne Laser Scanning Data Using JointBoost. ISPRS Journal of Photogrammetry and Remote Sensing, 100, 71-83.

Guo, Y., Wang, H., Hu, Q., Liu, H., Liu, L., Bennamoun, M., 2021. Deep Learning for 3D Point Clouds: A Survey. IEEE Transactions on Pattern Analysis and Machine Intelligence, 43(12), 4338–4364. https://doi.org/10.1109/TPAMI.2020.3005434

Hackel, T., Wegner, J. D., Schindler, K., 2016. Fast semantic segmentation of 3D point clouds with strongly varying density. ISPRS annals of the photogrammetry, remote sensing and spatial information sciences, 3, 177-184.

Hermosilla, P., Ritschel, T., Vázquez, P. P., Vinacua, À., Ropinski, T., 2018. Monte carlo convolution for learning on non-uniformly sampled point clouds. ACM Transactions on Graphics (TOG), 37(6), 1-12.

Kada, M., Kuramin, D., 2021. ALS Point Cloud Classification Using Pointnet++ And Kpconv With Prior Knowledge. The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, XLVI-4-W4-2021, 91–96. https://doi.org/10.5194/isprs-archives-XLVI-4-W4-2021-91-2021

Li, J., Chen, B. M., Lee, G. H., 2018. So-net: Self-organizing network for point cloud analysis. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 9397-9406).

Li, N., Kähler, O., Pfeifer, N., 2021. A Comparison of Deep Learning Methods for Airborne Lidar Point Clouds

Classification. IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, 14, 6467–6486. https://doi.org/10.1109/JSTARS.2021.3091389

Lin, Y., Vosselman, G., Yang, M. Y., 2022. Weakly supervised semantic segmentation of airborne laser scanning point clouds. ISPRS Journal of Photogrammetry and Remote Sensing, 187, 79–100. https://doi.org/10.1016/j.isprsjprs.2022.03.001

Luo, C., Li, X., Cheng, N., Li, H., Lei, S., Li, P., 2022. MVP-Net: Multiple View Pointwise Semantic Segmentation of Large-Scale Point Clouds. arXiv preprint arXiv:2201.12769.

Niemeyer, J., Rottensteiner, F., Soergel, U., 2014. Contextual classification of lidar data and building object detection in urban areas. ISPRS journal of photogrammetry and remote sensing, 87, 152-165.

Nong, X., Bai, W., Liu, G., 2023. Airborne LiDAR point cloud classification using PointNet++ network with full neighborhood features. Plos one, 18(2), e0280346.

Ozdemir, S., 2021. Automatic Extraction and Hydro-Flattening of Water Bodies in Digital Surface and Terrain Models, Doctoral dissertation, Karadeniz Technical University, Trabzon, Turkey.

Öngün, C., Temizel, A., 2021. LPMNet: latent part modification and generation for 3D point clouds. Computers & Graphics, 96, 1-13.

Qi, C. R., Su, H., Mo, K., Guibas, L. J., 2017a. PointNet: Deep Learning on Point Sets for 3D Classification and Segmentation. 652–660. https://openaccess.thecvf.com/content_cvpr_2017/html/Qi_PointNet_Deep_Learning_CVPR_2017_paper.html

Qi, C. R., Yi, L., Su, H., Guibas, L. J., 2017b. PointNet++: Deep Hierarchical Feature Learning on Point Sets in a Metric Space (arXiv:1706.02413). arXiv. https://doi.org/10.48550/arXiv.1706.02413

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

Thomas, H., Goulette, F., Deschaud, J. E., Marcotegui, B., LeGall, Y., 2018, September. Semantic classification of 3D point clouds with multiscale spherical neighborhoods. In 2018 International conference on 3D vision (3DV) (pp. 390-398). IEEE.

Turgut, K., Dutagaci, H. 2024. Local region-learning modules for point cloud classification. Machine Vision and Applications, 35(1), 1-19.

Varney, N., Asari, V. K. Graehling, Q., 2020: DALES: A Large-Scale Aerial Lidar Data Set for Semantic Segmentation. *IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops, Haziran, Seattle, WA, USA, 717-726.*

Winiwarter, L., Mandlburger, G., Schmohl, S., Pfeifer, N., 2019. Classification of ALS Point Clouds Using End-to-End Deep Learning. PFG – Journal of Photogrammetry, Remote Sensing and Geoinformation Science, 87(3), 75–90. https://doi.org/10.1007/s41064-019-00073-0

Wu, W., Qi, Z., Fuxin, L. 2019. Pointconv: Deep convolutional networks on 3d point clouds. In Proceedings of the IEEE/CVF Conference on computer vision and pattern recognition (pp. 9621-9630).

Xu, Y., Fan, T., Xu, M., Zeng, L., Qiao, Y., 2018. Spidercnn: Deep learning on point sets with parameterized convolutional filters. In Proceedings of the European conference on computer vision (ECCV) (pp. 87-102).

Yan, X., 2019. Pointnet/Pointnet++ Pytorch. https://github.com/yanx27/Pointnet_Pointnet2_pytorch

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

Zhao, H., Jiang, L., Fu, C. W., Jia, J., 2019. Pointweb: Enhancing local neighborhood features for point cloud processing. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (pp. 5565-5573).

Zhao, H., Jiang, L., Jia, J., Torr, P. H., Koltun, V., 2021. Point transformer. In Proceedings of the IEEE/CVF international conference on computer vision (pp. 16259-16268).