MOBILE POINT CLOUD SEMANTIC SEGMENTATION USING ARTIFICIAL NEURAL NETWORK AND APPROPRIATE PARAMETER DETERMINATION

Muhammed Enes Atik[1] and Zaide Duran[1]
[1]Istanbul Technical University, Faculty of Civil Engineering, Department of Geomatics Engineering, Istanbul, Turkey
Email: atikm@itu.edu.tr, duranza@itu.edu.tr

KEY WORDS: Point cloud, ANN, Semantic segmentation, LiDAR, geometric features.

ABSTRACT: The processing and information extraction of mobile point clouds has become an essential field of study in photogrammetry, remote sensing, computer vision, and robotics. Semantic segmentation is called to evaluate the singular features of the points together and collect them under meaningful clusters. This study aims to perform semantic segmentation with appropriate parameter selection using artificial neural networks. In addition, a study has been carried out to optimally define a point in the point cloud with the different feature spaces produced. Accordingly, eigen-based features are defined for each point. Eigen-based features describe the local geometry around the point and are commonly used in LiDAR processing today. Then, the most suitable parameters for semantic segmentation are determined. Multilayer Perceptron (MLP), an artificial neural network approach, was used in the study. The multilayer perceptron (MLP) is an artificial neural network to train any given non-linear input and contains several layers. Therefore, MLP is a suitable approach for solving non-linear problems. MLP has three layers: the input layer, the hidden layer, and the output layer. Paris-Carla-3D MLS dataset was used in the study. Paris-Carla-3D consists of two datasets, real (Paris) and synthetic (Carla). The dataset consists of data collected on a route 550 meters in Paris, 5.8 km in CARLA. The only real part Paris was used in this study. The highest mIoU metrics were obtained as 21.85% with the 0.4 m support radius, 30000 training samples and 200 hidden layer size.

1. INTRODUCTION

Depth perception is created for machines and 3D computer vision with three-dimensional (3D) models of objects produced by laser scanning (Duran and Aydar, 2012). Point clouds contain 3D position information, intensity, scan angle rank, and color. Extraction of information from three-dimensional (3D) data is a trending topic in photogrammetry, remote sensing, computer vision, and robotics (Atik *et al.*, 2021). Recently, different sensors such as RGB cameras, light detection and ranging (LiDAR), depth camera or Radar, have been widely used for autonomous driving. LiDARs are now an essential part of sensing systems as they provide direct point cloud generation (Biasutti *et al.*,2019). Mobile LiDAR point clouds, which are mounted on a vehicle, are used for many tasks such as object detection, object tracking, and semantic segmentation (Li *et al.*, 2021).

Artificial Intelligence (AI) has wide usage areas in photogrammetry and remote sensing applications like many other research areas (Atik and Ipbuker, 2020). Deep learning (DL), which is one of the AI approaches, is increasing in usage in different fields because it is robust and requires less operator labor (Atik and Ipbuker, 2021). The number of layers is increased to improve the performance of DL architectures for fast and automatic feature extraction from large datasets. (Atik et al., 2022). Deep learning networks are used in many applications in the fields of computer vision, photogrammetry and remote sensing. Researchers improve successful point cloud semantic segmentation approaches based on machine learning and deep learning algorithms. Different approaches dealing with the semantic segmentation problem continue to be developed in the literature. It is possible to collect these approaches under 3 main headings: Point-based, voxel-based and projection-based. Point-based methods learn the features of each point through shared MLPs (Qi *et al.*, 2017; Qi *et al.*, 2017b; Hu *et al.*, 2020; Jiang *et al.*, 2018). Point convolution methods recognize weights based on learned features with convolutions with more inputs (Li *et al.*, 2018; Xu *et al.*, 2018, Zhou *et al.*, 2021). By collecting local shape information from neighbors, graph-based algorithms build point clouds as super-graphs and send it to a graph convolution network (Wang *et al.*, 2018; Landrieu and Simonovsky, 2018). Voxel-based methods define point clouds within specific geometric shapes rather than processing them directly (Maturana and Scherer, 2015). However, 3D information loss may occur as a result of these transformations. Projection-based methods reduce the point cloud from 3D space to a 2D plane. Thus, they treat point cloud semantic segmentation as an image processing problem. Similar to voxel-based approaches, 3D information loss occurs (Atik and Duran, 2022). Some studies apply point cloud classification using machine learning and handcrafted features (Weinmann *et al.*, 2015; Duran *et al.*, 2021)

In this study, point cloud semantic segmentation was applied to the Paris part of the mobile LiDAR dataset Paris-CARLA-3D using the multilayer perceptron (MLP) method, which is an artificial neural network. The data in the point cloud is defined by feature vectors containing eigen-based features that describe the local geometries of the points. The experiment was repeated with different parameter values to determine the appropriate parameters. Thus, it is aimed to obtain the optimum result. Optimal values were determined for the radius of support area suitable for the Eigen-based calculation, the number of points for the training data set, and the size of the MLP hidden layer.

2. MATERIAL AND METHODS

2.1 Paris-CARLA-3D Dataset

The Paris-CARLA-3D (PC3D) dataset was created with a mobile mapping system including a LiDAR (Velodyne HDL32) inclined at 45° to the horizon and a 360° poly-dioptric Ladybug5 (Deschaud *et al.*, 2021). Paris-Carla-3D consists of two datasets, real (Paris) and synthetic (Carla). The dataset consists of data collected on a route 550 meters in Paris, 5.8 km in CARLA. Only real part Paris was used in this study. Although the real part does not cover a large area (it includes three streets in the center of Paris), it is captured in areas where the number and variety of urban objects, pedestrian movements and vehicles are dense, allowing for various analyzes. Paris dataset consists of six point clouds containing 10 million points (S0 to S5), a total of 60 million points. The points are labeled under 23 classes. In addition, since the mobile LiDAR system also includes a camera, the point cloud is colored (RGB) as a result of the necessary orientation processes. Example point cloud and labels from the dataset are shown in Figure 1.

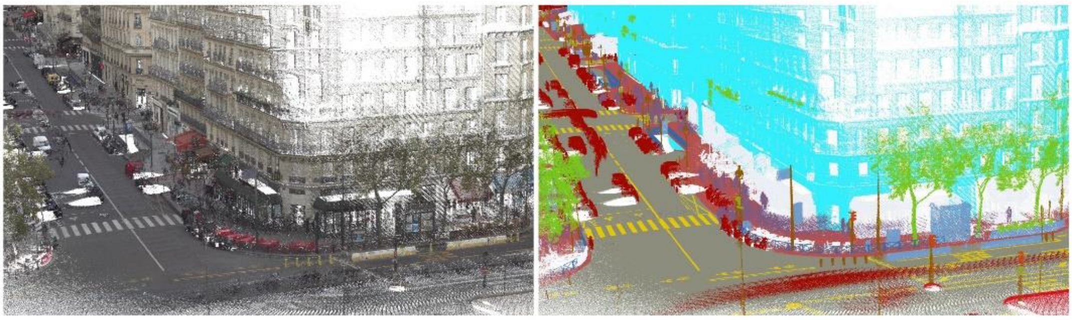

Figure 1. Sample cloud from Paris-CARLA-3D (PC3D) dataset. Colored cloud on left and labeled cloud on right (Deschaud *et al.*, 2021).

2.2 Multilayer Perceptron (MLP)

The multilayer perceptron (MLP) is an artificial neural network to train any given nonlinear input and contains several layers. A single perceptron can solve linear problems but not non-linear problems. MLP is a suitable approach for solving non-linear problems. MLP has three layers: the input layer, hidden layer, and output layer. In an MLP, data flows from the input layer to the output layer, as in feed-forward networks. The input layer takes the input value to be processed in the network. The final task such as classification or regression is applied in the output layer. Hidden layers which are nonlinear layers, are placed between the input layer and output layer. Most of the computational load in the network is on the hidden layers. In classification problems, similar to all neural networks, the number of neurons in the input layer depends on the size of the input vector, while the number of neurons in the output layer is determined by the number of classes to be learned (Meyer-Bäse *et al.*, 2004)
A two-phase backpropagation (BP) algorithm is used in the training phase of an MLP. The output of the network and the error value are determined during the forward propagation phase. In the backpropagation stage, the error value propagates backward over the network. Adjustments are applied to the network's connection weight values between the layers to minimize the error value.

Input data is taken into the network through the input layer and the “activation” propagates in the forward direction. Each hidden layer works like a detector on its own.

$$z_h = sigmoid(w_h^T x) = \frac{1}{1 + exp\left[-\left(\sum_j^d w_{hj} x_j + w_{h0}\right)\right]}, h = 1,2, \dots, H \tag{1}$$

The values calculated in the hidden layers are used as inputs to generate the yi prediction values in the output layer (Atik *et al.*, 2021).

$$y_i = v_i^T z = \sum_{h=1}^{H} v_{ih} z_h + v_{i0} \tag{2}$$

2.3 Geometric Features

Eigen-based features are those that describe the local geometry around the point and are commonly used in LiDAR processing. A sphere or other geometric shape with that point as the center can be used to calculate the neighboring points surrounding a given location. This neighborhood area is called the support area. In this study, the support area was determined with a sphere. The critical parameter when creating the support area is the radius of the sphere. The sphere radius that best describes the local geometry should be determined. The eigen-based features (Weinmann *et al.*, 2015) used in this study are presented in Table 1.

Table 1. Eigen-based features were used in this study.

Feature	Explanation
Sum of eigenvalues	$\lambda_1 + \lambda_2 + \lambda_3$
Omnivariance	$\sqrt[3]{\lambda_1 \lambda_2 \lambda_3}$
Eigenentropy	$\sum_{i=1}^{3} \lambda_i \ln \lambda_i$
Anisotropy	$(\lambda_1 - \lambda_3)/\lambda_1$
Planarity	$(\lambda_2 - \lambda_3)/\lambda_1$
Linearity	$(\lambda_1 - \lambda_2)/\lambda_1$
Surface variation	$\lambda_3/(\lambda_1 + \lambda_2 + \lambda_3)$
Sphericity	λ_3/λ_1
Verticality	$1 - \lvert\langle[0\ 0\ 1], \lambda_3\rangle\rvert$
Height value	Z_i
Roughness	Other features
Normal change rate	
Number of neighbors	
Volume density	

2.4 Experiment

In this study, point cloud semantic segmentation was performed with MLP. Each point in the point cloud is defined by its local eigen-based features. The most important parameter when determining eigen-based features is the support area of the point. In order to determine the appropriate radius, the experiment was applied with four different radius values. These radius values are defined as R_{01}, R_{02}, R_{03} and R_{04}. Numerical values represent the radius in meters. After determining the appropriate radius for the geometric features, the number of points suitable for training was calculated. Accuracy is adversely affected in machine learning approaches, as unbalanced data can lead to bias in favor of large classes. For this reason, experiments were carried out with training sets created by selecting 10000, 20000, 30000 and 40000 points from each class. Additionally, an experiment was carried out for the hidden layer dimension of the MLP algorithm. Point cloud semantic segmentation is applied for 50, 100, 150 and 200 hidden layer sizes.

Geometric features were calculated with open-source CloudCompare software. MLP experiments were developed using the sci-kit learn library in the Python environment (Pedregosa *et al.*, 2011). For the experiments, i7-11800H, 2.30 GHz processor, GTX 3070 graphics card, and 32 GB RAM hardware is used. Precision, recall, F1 score, mIoU and overall accuracy were used as evaluation metrics.

3. RESULTS AND DISCUSSION

The support radius should be determined optimally for eigen-based feature extraction. If a large radius is chosen, both the processing load will increase and there will be results that exceed the local region of the point. If a small radius is determined, the local geometric area of the point will not be adequately defined and similar properties will be calculated for all points. This will make it difficult to distinguish the points. The eigen-based features calculated at four different radius sizes were determined and semantic segmentation was carried out using the MLP algorithm. The highest evaluation metrics were obtained with the $R_{0.4}$ support radius. Only in the accuracy metric, the highest value was obtained at the $R_{0.2}$ radius with 63.99%. The lowest values in all metrics were obtained at the R0.1 radius. Eigen-based features calculated using the $R_{0.1}$ radius are insufficient to represent a point's local geometry. In general, the metrics increase as the support area gets larger. The results of the test performed for the radius of support are presented in Table 2. Predicted point clouds are shown in Figure 2.

Table 2. Evaluation results based on different support radius for eigen-based features.

Support Radius	Precision	Recall	F1 Score	mIoU	Accuracy
$R_{0.1}$	26.79	30.44	25.14	17.33	57.73
$R_{0.2}$	28.45	37.64	28.14	20.23	**63.99**
$R_{0.3}$	28.61	39.17	28.68	20.27	62.58
$R_{0.4}$	**30.34**	**42.65**	**30.00**	**20.62**	57.31

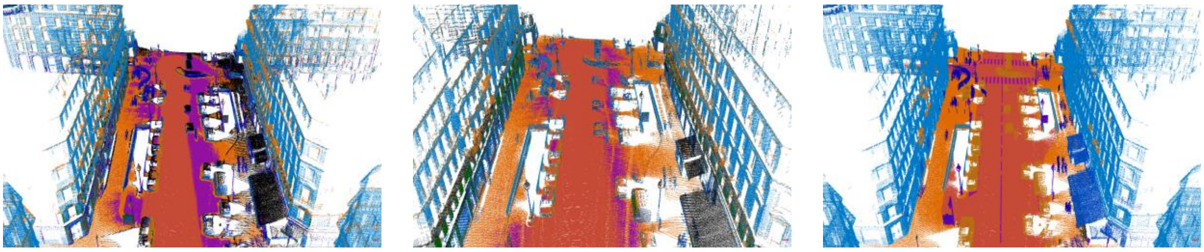

Figure 2. Predicted point clouds based on different support radius for eigen-based features (left: ground truth, middle: prediction with the highest mIoU, right: prediction with the lowest mIoU).

In the evaluation made according to the size of the training data set, it was concluded that 30000 points were the most appropriate parameter. The highest F1 score, mIoU and accuracy values were obtained with the number of 30000 points as 31.08%, 21.85% and 62.88%, respectively. The lowest metrics were obtained, except for the recall at 10000 points. From 10000 to 30000 generally metrics tended to increase but decreased at 40000 points. The reason for this is that the training set is out of balance, as there are less samples at 40000 points in some classes. The results of the experiment to determine the appropriate training set are presented in Table 3. Predicted point clouds are shown in Figure 3.

Table 3. Evaluation results based on different training sample for each class.

Number of points	Precision	Recall	F1 Score	mIoU	Accuracy
10000	30.34	**42.65**	30.00	20.62	57.31
20000	**31.06**	42.01	30.27	21.05	59.25
30000	30.82	42.58	**31.08**	**21.85**	**62.88**
40000	30.61	40.77	30.43	20.90	57.10

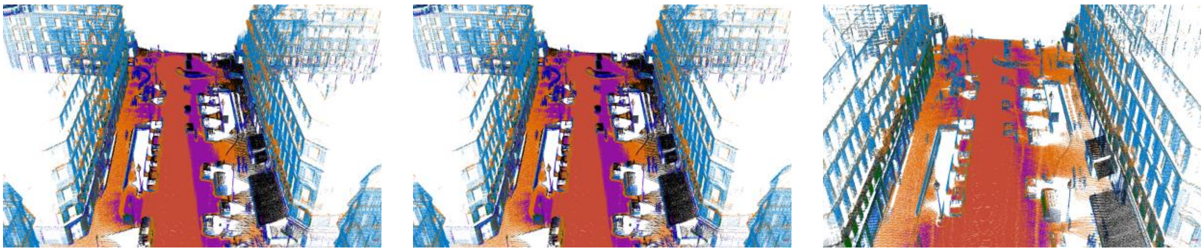

Figure 3. Predicted point clouds based on different training sample for each class (left: ground truth, middle: prediction with the highest mIoU, right: prediction with the lowest mIoU).

Considering the mIoU value, the mIoU increases as the hidden layer grows in size. The highest F1 score value was obtained at 200 with 31.08%. Depending on the dataset structure and hardware features, the hidden layer size can be selected. Especially when the accuracy is examined, 200 hidden layer sizes with 62.88% accuracy are superior to other hidden layer sizes. The results for the hidden layer size dimension are presented in Table 4. Predicted point clouds are shown in Figure 4.

Table 4. Evaluation results based on different hidden layer size of MLP.

Hidden Layer Size	Precision	Recall	F1 Score	mIoU	Accuracy
50	30.47	41.37	29.84	20.45	54.92
100	30.07	40.91	30.33	21.00	58.14
150	30.55	40.71	29.85	20.51	55.62
200	**30.82**	**42.58**	**31.08**	**21.85**	**62.88**

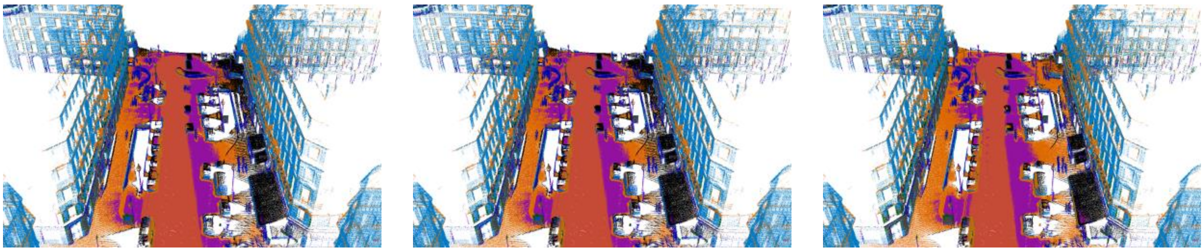

Figure 4. Predicted point clouds based on different hidden layer size of MLP (left: ground truth, middle: prediction with the highest mIoU, right: prediction with the lowest mIoU).

3. CONCLUSIONS

In this study, an experiment on point cloud semantic segmentation and appropriate parameter selection using the MLP method is presented. Considering the importance of mobile LiDAR point clouds for the environmental sensing of autonomous vehicles, there is a need for high-accuracy information extraction in such point clouds. In future studies, different deep learning and machine learning methods can be used besides MLP. In addition, filtering methods can be developed to remove moving objects causing noise in mobile LiDAR point clouds to improve accuracy. Considering the rapid rise of artificial intelligence studies in every field today, artificial intelligence approaches have an important potential for point cloud semantic segmentation.

REFERENCES

Atik, M. E., Duran, Z. 2022. An Efficient Ensemble Deep Learning Approach for Semantic Point Cloud Segmentation Based on 3D Geometric Features and Range Images. Sensors, 22(16), 6210.

Atik, M. E., Duran, Z., Seker, D. Z. 2021. Machine learning-based supervised classification of point clouds using multiscale geometric features. ISPRS International Journal of Geo-Information, 10(3), 187.

Atik, S. O., Atik, M. E., Ipbuker, C. 2022. Comparative research on different backbone architectures of DeepLabV3+ for building segmentation. Journal of Applied Remote Sensing, 16(2), 024510.

Atik, S. O., Ipbuker, C. 2020. Instance Segmentation Of Crowd Detection In The Camera Images. In Proceeding of Asian Conference on Remote Sensing.

Atik, S. O., Ipbuker, C. 2021. Integrating convolutional neural network and multiresolution segmentation for land cover and land use mapping using satellite imagery. Applied Sciences, 11(12), 5551.

Biasutti, P., Lepetit, V., Aujol, J. F., Brédif, M., Bugeau, A. 2019. Lu-net: An efficient network for 3d lidar point cloud semantic segmentation based on end-to-end-learned 3d features and u-net. In Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops (pp. 0-0).

Deschaud, J. E., Duque, D., Richa, J. P., Velasco-Forero, S., Marcotegui, B., Goulette, F. 2021. Paris-CARLA-3D: A real and synthetic outdoor point cloud dataset for challenging tasks in 3D mapping. Remote Sensing, 13(22), 4713.

Duran, Z., Aydar, U. 2012. Digital modeling of world's first known length reference unit: The Nippur cubit rod. Journal of cultural heritage, 13(3), pp. 352-356.

Duran, Z., Ozcan, K., Atik, M. E. 2021. Classification of Photogrammetric and Airborne LiDAR Point Clouds Using Machine Learning Algorithms. Drones, 5(4), 104.

Hu, Q., Yang, B., Xie, L., Rosa, S., Guo, Y., Wang, Z., Markham, A. (2020). Randla-net: Efficient semantic segmentation of large-scale point clouds. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 11108-11117.

Jiang, M., Wu, Y., Zhao, T., Zhao, Z., Lu, C. 2018. Pointsift: A sift-like network module for 3d point cloud semantic segmentation. arXiv preprint arXiv:1807.00652.

Landrieu, L., Simonovsky, M. 2018. Large-scale point cloud semantic segmentation with superpoint graphs. In Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 4558-4567.

Li, S., Liu, Y., & Gall, J. 2021. Rethinking 3-D LiDAR Point Cloud Segmentation. IEEE Transactions on Neural Networks and Learning Systems.

Li, Y., Bu, R., Sun, M., Wu, W., Di, X., Chen, B. 2018. Pointcnn: Convolution on x-transformed points. Advances in neural information processing systems, 31.

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

Meyer-Bäse, A., Meyer-Baese, A., Schmid, V. J. 2004. Pattern Recognition and Signal Analysis in Medical Imaging. Academic Press.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., ... Duchesnay, E. 2011. Scikit-learn: Machine learning in Python. the Journal of machine Learning research, 12, pp. 2825-2830.

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

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.

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), pp. 1-12.

Weinmann, M., Jutzi, B., Hinz, S., Mallet, C. 2015. Semantic point cloud interpretation based on optimal neighborhoods, relevant features and efficient classifiers. ISPRS Journal of Photogrammetry and Remote Sensing, 105, 286-304.

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.

Zhou, H., Feng, Y., Fang, M., Wei, M., Qin, J., Lu, T. 2021. Adaptive graph convolution for point cloud analysis. In Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 4965-4974.