UNS Geo: LiDAR Dataset for point cloud classification in urban areas

Miro Govedarica [1], Gordana Jakovljevic [2] , Igor Ruskoviski [1], Vladimir Pajic [1]
[1] Faculty of Technical Science, University of Novi Sad, Dr. Zorana Đinđiča 1, 21000 Novi Sad, Serbia – miro@uns.ac.rs
[2] Dept. of Geodesy, Faculty of Architecture, Civil Engineering and Geodesy, University of Banja Luka, Bulevar vojvode Petra Bojevica 1, 78000 Banja Luka, Bosnia and Herzegovina. – gordana.jakovljevic@aggf.unibl.org

Keywords: benchmark dataset, ALS point cloud, semantic classification, urban areas.

Abstract

The classification of the urban point cloud is an essential task for numerous applications, including mapping, 3D urban modelling, etc.. Although in the last few years, different methodologies and algorithms have been proposed, precise and detailed point cloud labelling is still challenging. Publicly available annotated benchmark datasets have become the standard for the evaluation of algorithms' performance; however, most focus on data acquired from mobile or terrestrial laser scanners. In this paper, we introduce UNS Geo, a dense Aerial Laser Scanning (ALS) point cloud dataset consisting of 5.4 million manually annotated points across 8 semantic classes. To validate the performance of our dataset, the labelled point cloud is used for training the state-of-the-art networks (i.e. PointNet, PointNet++). Moreover, since UNS Geo includes the RGB per point information, the influence of spectral information on classification results is evaluated. The results demonstrate that UNS Geo effectively supports the training of deep learning models, highlighting its potential for advancing research in urban point cloud classification. The dataset is publicly available at: https://github.com/mirogovedarica/UNS-Geo.

1. Introduction

Automatic and reliable 3D point cloud classification is a crucial yet challenging task with applications across various domains, including urban planning, 3D modelling (Biljecki, et. al., 2015), autonomous driving (Zhao, Peng, & Azumi, 2024) and the development of smart cities and digital twins (Xue, et. al., 2020), (Wu & Zhou, 2022). Airborne LiDAR (Light Detection and Ranging) has emerged as an efficient and effective tool for conducting large-scale 3D surveys of urban areas, offering high spatial resolution and accurate data collection. The point cloud serves as a primary data source for the geometrical reconstruction and modelling of the urban environment, and classification of raw point cloud is the first step in the processing workflow.

In addition to the increased need to develop fully automated point cloud classification algorithms that will enable data processing in real-time or near-real time, detailed classification is also needed. A decade ago, the algorithms were focused on distinguishing a small number of broad classes such as buildings, terrain, and vegetation; however, featured classes such as facades, pedestrian lanes, parking lots, poles, etc, are particularly interesting nowadays. The detailed classification has unlocked new applications and increased requirements for data quality, processing power, labeled training data, and advanced algorithms. Over the years, numerous algorithms and methodologies have been proposed for point cloud classification. Despite advancements in machine learning and deep learning, this task remains a significant challenge in the geospatial community.

One of the primary challenges lies in the availability of sufficient labeled data for training classification algorithms. The creation of publicly accessible, large-scale datasets is essential for developing and benchmarking new methods. The benchmark point cloud datasets can be categorized by the type of sensors used, the number of classes and the inclusion of color information (Table 1). There have been several data sets released for semantic segmentation from ground-based laser scanners such as Terrestrial Laser Scanners (TLS) or Mobile Laser Scanners (MLS). However, there is a significant difference between Arial Laser Scanners (ALS) and ground-based methods in resolution, point locations, and areas of occlusion. Due to that nadir orientation on the sensor, object appearance can significantly differ compared to lateral orientation in a ground-based system. This difference in perspective affects how features such as walls, facades, and vegetation are captured. Moreover, due to its lower resolution, ALS data are suited for different applications, and the types of classes that can be reliably detected may differ from those detectable using higher-resolution MLS or TLS data.

Several ALS databases have been introduced, such as ISPRS Vaihingen (Niemeyer, et. al., 2014), LASDU (Ye, et al., 2020), OpenGF (Qin, et. al., 2021) or AHN3 (AHN, n.d.). However, they have some limitations. For instance, while LASDU and AHN3 datasets are valuable resources for point cloud classification, they lack the comprehensive diversity of urban-specific classes, limiting their utility in capturing the complexity of dense urban environments. OpenGF has large coverage, but it is classified into two categories. The ISPRS benchmark dataset is most commonly used in resources in this field. It provides a point cloud classified into nine classes, along with features such as x, y, z, intensity, return number, and the number of returns. However, this dataset also presents some challenges, including its highly unbalanced class distribution and the relatively small number of points available, particularly for training deep learning methods. The overview of publicly available ALS benchmark datasets is presented in Table 1.

To perform point cloud classification, usually supervised classification that consists of two main steps: feature creation and point cloud classification has been used. The feature extraction process uses the local context of each point (defined by either a fixed radius or a fixed number of nearest neighbor points) and mathematical expressions to create a meaningful representation based on spectral or spatial attributes. Created features, that describe object types, are integrated into a feature vector and fed into the algorithm. The classification is performed based on those features. In the classification step, machine learning algorithms such as Support Vector Machine

(SVM) (Zhao, et. al., 2020), Random Forest (RF) (Aljumaily, et. al., 2023) is most commonly used. However, their performance mainly relies on the definition of neighborhood and selection of handcrafted features. The selection of appropriate features and neighbourhood definition demands prior knowledge of the point cloud (Winiwarter et. al., 2019)

Dataset	Number of points	Spatial size [m^2]	Number of classes	RGB
ISPRS (Niemeyer, et. al., 2014)	1.2 M	1.6 x 10^5	9	No
OpenGF (Qin, et. al, 2021)	500 M	47 x 10^6	2	No
LASDU (Ye, et al., 2020)	3.1 M	1 x 10^6	5	No
DALES (Varney, et al, 2020)	505 M	10 x 10^6	8	No
DublinCity (Iman Zolanvari, et al., 2019)	260 M	2 x 10^6	13	No

Table 1. Comparison of ALS 3D dataset.

Deep learning (DL) has been successfully applied in different types of classification, object detection, and segmentation tasks on images. However, DL on point cloud adds additional complexity due to high dimensionality, unstructured, and highly variable density nature. The third dimension increases the number of parameters in the network, requiring more computational resources. Unlike 2D images, point clouds have unstructured, irregular, and non-uniform shapes, due to which the application of the standard Convolutional Neural Network (CNN) can be difficult (Wu, Qi, & Fuxin, 2019).

Currently, deep learning methodologies for point cloud segmentation can be divided into grid-based and point-based approaches. Projection-based methods such as 2D images (Li, et. al, 2025), multi-view images (Boulch, et al., 2018), voxels (Zhao, et. al., 2023), kd-trees (Klokov & Lempitsky, 2017), transform point cloud into regular representations to which convolution filters can be directly or in a similar way applied. Similarly,

PointNet architecture represents the first network that applies DL models to raw point clouds. PointNet (Qi, et. al., 2017) uses a symmetrical function to address the unstructured point cloud and map it into high-dimensional space. Due to that, PointNet ensures that the model is invariant to input permutation. However, it can not capture the local structure of the point cloud. To solve this limitation, PointNet++ (Qi, et. al., 2017b) is introduced. PointNet++ has a U-Net-like architecture where a hierarchical set abstraction layer preforming sampling grouping, local feature extraction, as well as a feature propagation layer to upsample those features back to the original points.

In this paper, we introduced a large-scale aerial LiDAR point cloud dataset, UNS GEO, designed specifically for the classification of complex urban environments. This new dataset complements the ISPRS Vaihingen dataset and can be synchronized to provide a more comprehensive representation of urban morphologies. In addition to providing new data, we evaluate the two state-of-the-art algorithms, including PointNet and PointNet++ and test the influence of spectral information on algorithm performance.

2. UNS Geo

The UNS Geo dataset comprises over five million points, classified into eight distinct classes, and is focused on the City of Novi Sad (Serbia), which is known for its unique urban morphology. The city's layout reflects the architectural and planning styles typical of Southeastern Europe in the post-World War II era, featuring a mix of high-density residential blocks, green spaces, wide boulevards, narrow streets, and diverse building types. These characteristics ensure that the dataset captures a wide range of structural and spatial variations. The study area is in the urban area of Novi Sad, consisting of Liman, located in the southeast part of the city, and the left Danube bank with high residential blocks, spacious green areas, and boulevards. The topography of the study area is flat, with an average elevation of 77 m. The study area is shown in Figure 1.

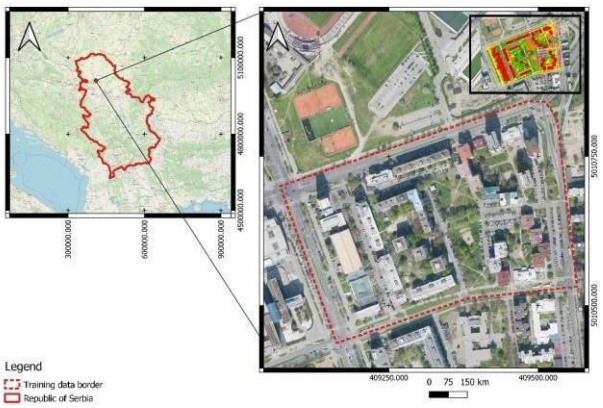

Figure 1. Study area.

2.1 Data acquisition

The ALS point cloud data were collected using a Riegl LMS-Q680i laser scanner and a digital camera DigiCam H39 onboard a Bell JetRanger 206B3. Before the survey, calibration of the onboard sensor is performed by using a double cross scheme i.e., the flight is conducted in two opposite directions on two flight heights (200 and 400 m).

2.2 Data description

The total number of annotated points is 5.4 million of points. The dataset is divided into two .las files: for training and for testing (Table 2).

	Num. of points	Point density [pts/m2]
Training	4.8 M	37
Test	0.6 M	35

Table 2. Dataset characteristics.

In the .las file, each point was assigned the following attributes:

- Position: X, Y, Z coordinates of each point in UTM 34N (EPSG:32634) projection. The original coordinates are available to enable fusion data appraoch,
- Intensity: Return strength of reflectance for each point. The intensity represents the surface physical properties.

- Return number: refers to the serial number of the reflected laser pulse that returns to the sensor after hitting the object of interest.
- Number of returns: the total number of returns per transmitting pulse,
- Classification: Define the type of object that has reflected the laser pulse.
- RGB: Each point in the point cloud is assigned a Red, Green, and Blue (RGB) color value based on the corresponding pixel in the aligned camera image by using Terrasolid software. Colors are added to enhance the visual interpretability and semantic understanding of the scene.

Regarding the labeling, the automatic, semi-automatic, and manual classification was used. We selected classes with a focus on different applications such as mapping, urban planning, and autonomous driving. The points are classified into eight different classes: ground, roads, parking, pedestrian lens, buildings, high vegetation, and cars (Figure 2). The unknown objects, which encompass objects such as traffic lights, benches, antennas, playgrounds, bus stops etc. are left out of the final point cloud. Each category includes, but is not limited to, the following types of objects:

- Ground class: bare earth, grass
- Roads: boulevards, collector roads, narrow streets
- Parking: angled parking, parking lots
- Pedestrian lanes: sidewalk, shortcuts
- Roof: flat roof, gable roof, hip roof (clay or concrete)
- High-vegetation: trees, shrubs, bushes
- Cars: sedans, SUVs, vans, trucks.

Information about class distribution is presented in Table 3.

Class code	Class description	Percentage of labeled points	
		Training	*Testing*
0	Ground	12.37	11.93
1	Roads	18.50	15.72
2	Parking	5.16	4.58
3	Pedestrian lens	11.51	7.54
4	Walls	5.88	7.52
5	Roof	18.78	25.46
6	High vegetation	25.15	25.68
7	Car	2.64	1.58

Table 3. Class distribution.

The training and testing datasets have a similar distribution, except for pedestrian lenses and cars. The high vegetation class points contain the largest number of points (Table 2.). This is expected since the multiple returns are characteristic of this class, and it is also a commonly occurring class in this type of city. The car class only reaches 2.64 % of all labeled points, making it one of the most challenging classes to detect. The imbalance of classes should be considered during the training or testing phase.

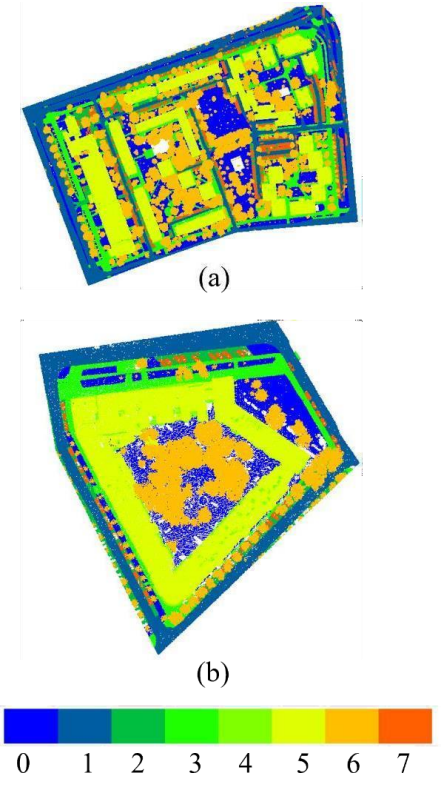

Figure 2. UNS Geo dataset (a) training data, (b) test data.

To understand the similarity in shapes between different classes, the analysis of the orientation of the surface normal per point was performed. First, the normal is computed by using least-squares plane fitting. To ensure consistency in orientation, the normal is flipped using the sensor as the reference. After that, the histogram of surface normal for each class is computed, and the correlation across classes. The results are presented in Figure 3. Each cell shows the correlation of class i and class j. The values close to 1 indicate that shapes are very similar, while 0 indicates completely different orientation.

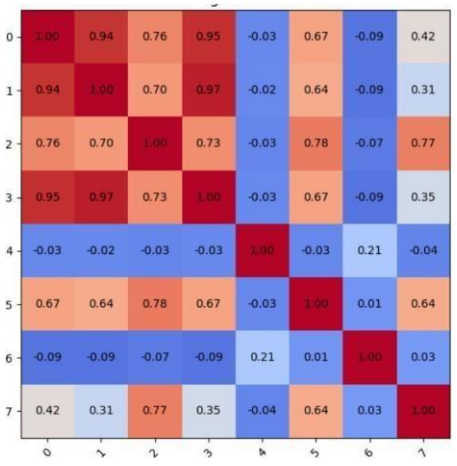

Figure 3. Surface normal correlation.

As expected, classes 0, 1, 2, and 3 have very similar surface normal distributions, indicating similar geometries, which can make distinguishing between them more challenging for classification algorithms. Classes 4 and 6 have a negative or low positive correlation, representing a structurally unique class. Class 5 shows moderate similarity, suggesting partial similar

geometry, while class 7 has high correlation with classes 2 and 5.

3. Methodology

The PointNet (Qi et al., 2017) architecture is a pioneering DL architecture developed for direct point cloud processing. The input to PointNet is a set of 3D points typically presented as a matrix (N, 3) where N is the number of points and each point is described by its coordinates. Additional features such as RGB value, intensity etc. can be used. Each point is independently passed through a series of shared Multi- Layer Perceptron (MLP) layers, which map each point into a high-dimensional feature space (from 3 to 64, 128 and 1024 dimensions). This step enables the model to learn complex spatial patterns at the point level. To aggregate the information from all individual points, PointNet uses max-pooling, producing a global feature vector (1024 dimensions), representing a global signature of the input point cloud. However, semantic segmentation demands the combination of local and global features. Due to that, the global feature vector is concatenated with each point's local feature, providing both local information and the context of the entire scene. This vector is passed through more MLP layers (512, 256, k), leading to a final softmax classifier per point. The output is (N, k), where k is the number of classes.

However, PoinNet fails to capture the local structure of the point cloud, which reduces its ability to characterize details and generalize samples with fine-grained local structure (such as sharp corners, object boundaries). PointNet++ (Qi, Yi, Su, & Guibas, 2017) resolves this problem by dividing the point cloud into a set of overlapping local regions using distance measurement in the corresponding space. Then it applies PointNet locally in a hierarchical fashion to extract local features in small regions and gradually expands the range to extract higher-level features until the global features of the entire point cloud are extracted. Finally, the classification was achieved by combining local and global features.

The core of the PointNet++ architecture is abstraction layers that consist of three sub-steps: sampling, grouping, and local feature extraction. First, multiple-resolution grouping subsampling is selected from the input point cloud (8192, 4096, 2048, and 1024 points) using iterative farthest point sampling to ensure uniform spatial coverage. For these selected points that act as centroids, 32 neighboring points are found in the respective layer using ball queries. For each group, a PointNet composed of an MLP with a setup (first layer (32, 32, 64), second layer (64, 64, 128), and third layer (128, 128, 256)) followed by a max-pooling layer, is applied to extract a local feature vector.

This vector summarizes a set of centroids with a learned local descriptor, which is essential for understanding of complex 3D scene. The learned point features are gradually interpolated and propagated back to the original dense point cloud by hierarchical propagation with distance-based interpolation and across-level skip connections (like in U-Net). The head of the network consists of a dense layer with 128 units, a dropout layer with a keep probability of 0.5, and another dense layer with 9 units. ReLU activation and max-pooling are used wherever appropriate.

Since all points include RGB and intensity value, we created 3 different data inputs for each algorithm to assess the influence of spectral information on classification accuracy. The first input is based purely on the geometrical information and it contains only point coordinates. The second input includes the intensity value, while the third input contains full geometrical and spectral information i.e. x, y, z, intensity, and RGB. . .

Accuracy assessment

In this paper, the performance of algorithms was evaluated using the precision (P), recall (R), F1-score (F1), overall accuracy, and average F1 score. The precision, recall, and F1 were used to assess the classification accuracy of individual classes, while the OA and average F1 is used to assess overall model performance.

$$P = \frac{TP}{TP + FP} \tag{1}$$

$$R = \frac{TP}{TP + FN} \tag{2}$$

$$F1 = \frac{2TP}{2TP + FP + FN} \tag{3}$$

Where TP is true positive, FN is false negative, and FP is false positive. The average F1 is calculated as the average summation of F1 for all classes, and OA (4) is calculated as follows:

$$OA = \frac{TP + TN}{TP + TN + FP + FN} \tag{4}$$

Where TN is true negative.

3.1 Implementation

The model is trained using mini-batch stochastic gradient descent, where each batch contains 32 blocks of points. The data set is divided into 80 % for testing and 20 % of data for validation. Training was performed for 40 epochs with an initial learning rate of 0.001, which decays by a factor of 0.7 every 10 epochs to improve convergence. The optimizer used is Adam, which adapts learning rates individually for each parameter. Additionally, L2 weight decay (set to 1e-4) is applied to prevent overfitting by penalizing large weights. The weights are saved if the training loss decreases, while the best model is selected according to the highest accuracy on the validation set.

4. Results

The results of the accuracy assessment for UNS Geo dataset classification by using PointNet and PointNet++ architectures and three different input sets are displayed in Table 4, while visual inspection of the results is shown in Figure 4. Taking into account the value of F1, both algorithms show moderate to weak performance depending on the input dataset.

As expected, PointNet++ outperforms PointNet across all input datasets. Similar results are presented in (Iman Zolanvari, et al., 2019) and (Ye, et al., 2020). An increase of an average F1 score was 0.12, 0.16, and 0.13 for the first, second, and third input, respectively. The highest accuracy is obtained for PointNet++ with full features (XYZ+I+RGB). The first input that relies solely on spatial information results in the lowest accuracy for both models. Using only geometrical features (X, Y, Z) can be insufficient for distinguishing objects with similar shapes, such as classes 0, 1, 2, 3 (Figure 3).

	PointNet (XYZ)			**PointNet ++ (XYZ)**			**PointNet (XYZ+I)**			**PointNet++ (XYZ+I)**			**PointNet(XYZ+I+RGB)**			**PointNet++(XYZ+I+RGB)**		
Class	***P***	***R***	***F1***	***P***	***R***	***F1***	***P***	***R***	***F1***	***P***	***R***	***F1***	***P***	***R***	***F1***	***P***	***R***	***F1***
0	0.36	0.08	0.13	0.59	0.12	0.20	0.41	0.31	0.35	0.59	0.15	0.24	0.57	0.34	**0.43**	0.51	0.44	**0.48**
1	0.70	0.26	0.38	0.83	0.27	0.41	0.67	0.35	0.46	0.72	0.58	**0.64**	0.65	0.68	0.66	0.80	0.51	0.63
2	0.16	0.27	0.20	0.22	0.29	0.25	0.18	0.29	0.22	0.20	0.54	**0.29**	0.18	0.27	0.22	0.23	0.29	0.25
3	0.24	0.20	0.22	0.20	0.70	0.32	0.23	0.19	0.21	0.22	0.26	0.24	0.21	0.11	**0.14**	0.25	0.36	**0.30**
4	0.26	0.29	0.27	0.34	0.56	0.42	0.23	0.40	0.29	0.44	0.80	**0.56**	0.21	0.49	0.29	0.39	0.81	0.53
5	0.72	0.56	0.63	0.70	0.88	0.78	0.75	0.67	0.71	0.85	0.88	**0.87**	0.77	0.83	0.80	0.90	0.85	**0.87**
6	0.48	0.69	0.56	0.75	0.47	0.58	0.51	0.58	0.54	0.84	0.69	**0.76**	0.60	0.45	0.51	0.78	0.69	0.73
7	0.10	0.67	0.18	0.42	0.56	0.56	0.21	0.45	0.29	0.66	0.78	**0.71**	0.26	0.49	0.34	0.73	0.58	0.64
AF1		0.32			0.44			0.38			0.54			0.42			0.55	
OA		0.44			0.54			0.48			0.63			0.53			0.65	

Table 4. Result of the accuracy assessment

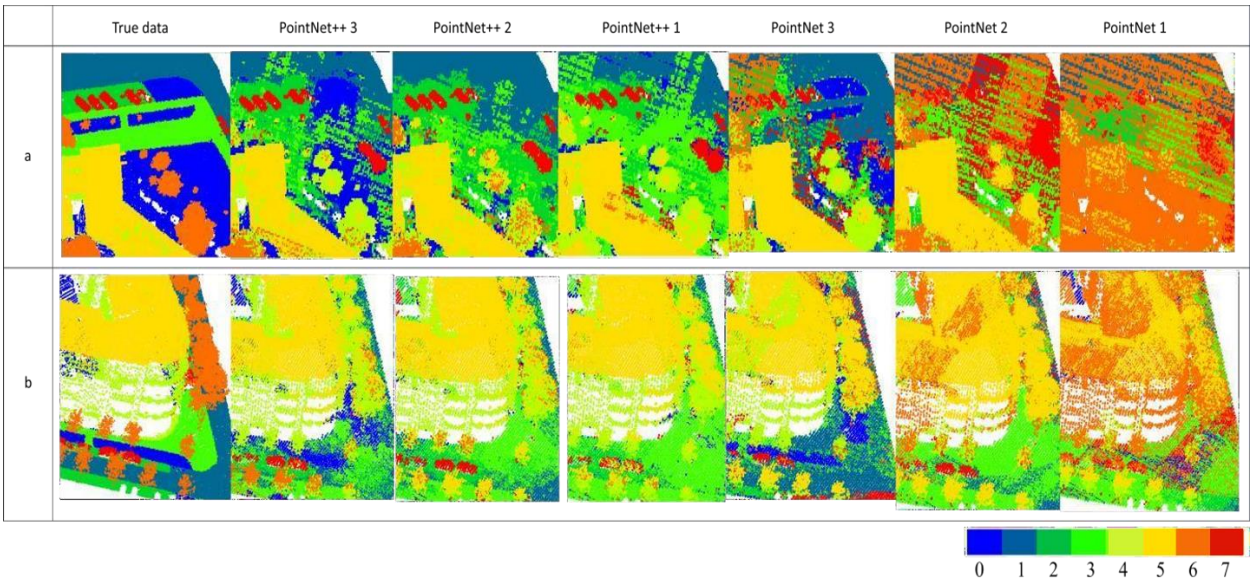

Figure 4. Visual comparison of results where 1 represents the first dataset (x, y, z), 2 represents the second dataset (x, y, z, intensity), and 3 represents the third dataset (x, y, z, intensity, r, g, b).

The inclusion of spectral features had a noticeable effect on classification performance. Adding intensity to the input significantly improved results for both PointNet and PointNet++ architectures. For PointNet, the average F1-score increased by 0.06, and an additional 0.06 gain was observed when RGB values were also included (Table 4.). For PointNet++, intensity led to an even larger improvement of 0.10 in average F1-score, while the addition of RGB further increased it by only 0.01. These results highlight the high benefits of intensity data, which captures discriminative material and surface characteristics directly from ALS measurements. Similarly, RGB values inclusion also improves algorithm performance, but it influence is more modest, particularly when using models like PointNet++ (Figure 4.). On one hand, both intensity and RGB represent the spectral reflectance characteristics of objects; however, LiDAR is less sensitive to environmental illumination and shadows compared to cameras, making it a more reliable source for accurate classification. On another hand, colour information can be highly beneficial for distinguishing classes with similar geometric features but different visual appearances. However, its effectiveness is relatively limited by factors such as lighting conditions, shadows, image quality, and the accuracy of image-to-point cloud alignment.

All models show similar behavior. The high vegetation and roof categories had strong performance over all networks and inputs. This is mostly due to the abundance of points in the dataset and unique spatial characteristics. Even though the car class is presented with only 2% in the training dataset, the PointNet ++ (XYZ+I) and PointNet ++ (XYZ+I+RGB) show strong performance in the test dataset (Figure 4.). The lowest accuracy was obtained for the parking lots class. This is expected since this class is highly imbalanced and has very similar spatial and spectral characteristics to the roads class.

5. Conclusion

In this paper, (i) a new benchmark aerial LiDAR point cloud dataset, UNS Geo, for 3D semantic segmentation of urban areas and (ii) evaluate the performance of two popular deep learning approaches, PointNet and PointNet++, and (iii) test the influence of spectral information on algorithms' performance. The dataset covers the complex and highly dense urban area, and it contains 5.4 million points manually classified into 8 classes: ground, roads, parking, pedestrian lens, walls, roofs, high vegetation, and cars. In contrast to other publicly available ALS datasets that contain x, y, z, and intensity information, the UNS Geo includes RGB information for each point, enabling easier visual interpretation and semantic understanding of the urban scene. This dataset is available at https://github.com/mirogovedarica/UNS-Geo.
The results of the state-of-the-art algorithm performance on UNS Geo dataset show that there is room for improvement in current methods, especially for the semantic segmentation of classes with similar spectral and spatial properties and imbalanced classes.

The accuracy assessment results demonstrate that both PointNet and PointNet++ show significant performance improvements with the inclusion of intensity data. In contrast, the addition of RGB information provides a more modest enhancement in classification accuracy, especially for the PointNet++ algorithm. Although color information can be highly beneficial for distinguishing between different classes, its effectiveness is relatively limited by factors such as image quality and the accuracy of image-to-point cloud alignment.
For future work, the study could be extended by labeling additional object classes and expanding the geographic coverage of the dataset. Furthermore, efforts could be directed toward the development of more advanced algorithms to achieve higher classification accuracy.

ACKNOWLEDGEMENT

This research has been supported by the Ministry of Science, Technological Development and Innovation (Contract No. 451-03-137/2025-03/200156) and the Faculty of Technical Sciences, University of Novi Sad through project “Scientific and Artistic Research Work of Researchers in Teaching and Associate Positions at the Faculty of Technical Sciences, University of Novi Sad 2025” (No. 01-50/295).

References

AHN. (n.d.). *Actueel Hoogtebestand Nederland*. Retrieved 11 4, 2024, from https://www.ahn.nl

Aljumaily, H., Laefer, D. F., Cuadra, D., & Velasco, M. (2023). Point cloud voxel classification of aerial urban LiDAR using voxel attributes and random forest approach. *International Journal of Applied Earth Observation and Geoinformation, 118*.

Biljecki, F., Stoter, J., Ledoux, H., Zlatanova, S., & Coltekin, A. (2015). Applications of 3D City Models: State of the Art Review. *ISPRS International Journal of Geo-Information*, 2842-2889.

Boulch, A., Guerry, J., Le Saux, J., & Audebert, N. (2018). SnapNet: 3D Point Cloud Semantic Labeling with 2D Deep Segmentation Networks . *Computers & Graphics, 71*, 189-198.

Iman Zolanvari, S. M., Ruano, S., Rana, A., Cummins, A., de Silva, R. E., Rahbar, M., & Smolic, A. (2019). DublinCity: Annotated lidar point cloud and. *Proceedings of the British Machine Vision.*

Klokov, R., & Lempitsky, V. (2017). Escape from Cells: Deep Kd-networks for the recognition of 3D point clouds models. *Proceedings of the IEEE International Conference on Computer Vision.*

Li, K., Zhang, T., Zhong, C., Zhang, Z., & Wang, G. (2025). Robust 3D point cloud classification based on declarative defenders. *Neural Computing and Applications, 37*, 1209-1221.

Niemeyer, J., Rottensteiner, F., & Soergel, U. (2014). Contextual classification of lidar data and building object detection in urban area. *ISPRS J. Photogramm. Remote Sens.,*, 152-165.

Qi, C. R., Su, H., Mo, K., & Guibas, L. J. (2017). PointNet: Deep Learning on Point Sets for 3D Classification and Segmentation. *Computer Vision and Pattern Recognition.* Honolulu, HI, USA.

Qi, C. R., Yi, L., Su, H., & Guibas, L. J. (2017). PointNet++: Deep Hierarchical Feature Learning on Point Sets in a Metric Space. *31st Conference on Neural Information Processing System.* Long Beach, CA, USA.

Qin, N., Tan, W., Ma, L., Zhang, D., & Li, J. (2021). An ultra-large-scale ground filtering dataset built upon open {ALS} point clouds around the world. *Proc. IEEE Conf. Comput. Vis. Pattern Recog. Workshops*, 1082-1091.

Varney, N., Asari, V. K., & Graehling, Q. (2020). DALES: A large-scale aerial lidar data set for semantic segmentation. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops.* IEEE.

Winiwarter, L., Mandlburger, G., Schmohl, S., & Pfeifer, N. (2019). Classification of ALS Point Clouds Using End-to-End Deep Learning. . *Journal of Photogrammetry, Remote Sensing and Geoinformation Science, 87*, 75-90.

Wu, W., Qi, Z., & Fuxin, L. (2019). PointConv: Deep Convolutional Networks on 3D Point Clouds . *Computer Vision and Pattern Recognition.* Long Beach, CA, USA: arXiv:1811.07246.

Wu, Y., & Zhou, Z. (2022). Intelligent City 3D Modeling Model Based on Multisource Data Point Cloud Algorithm. *Journal of Function Spaces*.

Xue, F., Weisheng, L., Zhe, C., & Webster, C. J. (2020). From LiDAR point cloud towards digital twin city: Clustering city objects based on Gestalt principles. *ISPRS Journal of Photogrammetry and Remote Sensing*, 418-431.

Ye, Z., Xu, Y., Huang, R., Tong, X., Li, X., Liu, X., Stilla, U. (2020). LASDU: A Large-Scale Aerial LiDAR Dataset for Semantic Labeling in Dense Urban Areas. *ISPRS Int. J. Geo-Inf., 9*(7).

Zhao, C., Peng, B., & Azumi, T. (2024). Point Cloud Automatic Annotation Framework for Autonomous Driving. *2024 IEEE Intelligent Vehicles Symposium (IV).* Jeju Island: IEEE.

Zhao, W., Zhang, X., Hao, X., Wang, D., & He, Y. (2023). Multi Point-Voxel COnvolution (MPVConv) for Deep Learning on Point Cloud. *Computers & Graphics*, 72-80.

Zhao, Z., Song, Y., Cui, F., Zhu, J., Song, C., Xu, Z., & Ding, K. (2020). Point Cloud Features-Based Kernel SVM for Human-Vehicle Classification in Millimeter Wave Radar. *IEEE Access, 8*, 26012-26021.