Enhancing explainability of deep learning models for point cloud analysis: a focus on semantic segmentation

International Journal of Digital Earth

ISSN: (Print) (Online) Journal homepage: www.tandfonline.com/journals/tjde20

Enhancing explainability of deep learning models for point cloud analysis: a focus on semantic segmentation

Francesca Matrone, Marina Paolanti, Emanuele Frontoni & Roberto Pierdicca

To cite this article: Francesca Matrone, Marina Paolanti, Emanuele Frontoni & Roberto Pierdicca (2024) Enhancing explainability of deep learning models for point cloud analysis: a focus on semantic segmentation, International Journal of Digital Earth, 17:1, 2390457, DOI: 10.1080/17538947.2024.2390457

To link to this article: https://doi.org/10.1080/17538947.2024.2390457

INTERNATIONAL JOURNAL OF DIGITAL EARTH
2024, VOL. 17, NO. 1, 2390457
https://doi.org/10.1080/17538947.2024.2390457

Enhancing explainability of deep learning models for point cloud analysis: a focus on semantic segmentation

Francesca Matrone [a], Marina Paolanti [b], Emanuele Frontoni [b] and Roberto Pierdicca [c]

[a]Department of Environment, Land and Infrastructure Engineering (DIATI), Politecnico di Torino, Torino, Italy; [b]Department of Political Sciences, Communication and International Relations, Università di Macerata, Macerata, Italy; [c]Department of Construction, Civil Engineering and Architecture (DICEA), Università Politecnica delle Marche, Ancona, Italy

ABSTRACT

Semantic segmentation of point clouds plays a critical role in various applications, such as urban planning, infrastructure management, environmental analyses and autonomous navigation. Understanding the behaviour of deep neural networks (DNNs) in analysing point cloud data is essential for improving segmentation accuracy and developing effective network architectures and acquisition strategies. In this paper, we investigate the traits of some state-of-the-art neural networks using indoor and urban outdoor point cloud datasets. We compare PointNet, DGCNN, and BAAF-Net on specifically selected datasets, including synthetic and real-world environments. The chosen datasets are S3DIS, SynthCity, Semantic3D, and KITTI. We analyse the impact of different factors such as dataset type (synthetic vs. real), scene type (indoor vs. outdoor), and acquisition system (static vs. mobile sensors). Through detailed analyses and comparisons, we provide insights into the strengths and limitations not only of different network architectures in handling urban point clouds but also of their data structure. This study contributes to going beyond the mere and unconditional use of AI algorithms, trying to explain DNNs behaviour in point cloud analysis and paving the way for future research to enhance segmentation accuracy and develop possible guidelines both for network design and data acquisition in the geomatics field.

ARTICLE HISTORY
Received 5 January 2024
Accepted 5 August 2024

KEYWORDS
Point cloud; deep learning; explaining artificial intelligence; semantic segmentation

1. Introduction

Point clouds data have emerged as a fundamental source of information for describing the environment due to their ability to provide rich geometric and spatial information (Balado et al. 2023). Urban and indoor environments present unique challenges and complexities, making the interpretability and explainability of AI models crucial for effective decision-making, urban planning, and infrastructure management. Urban point clouds, obtained from various sensors such as LiDAR (Light Detection and Ranging) or MMSs (Mobile Mapping Systems), provide a wealth of spatial information about cities, including buildings, roads, vegetation, and other urban features. Analyzing and understanding urban point cloud data is essential for tasks such as building reconstruction,

CONTACT Francesca Matrone francesca.matrone@polito.it Department of Environment, Land and Infrastructure Engineering (DIATI), Politecnico di Torino, Torino 10129, Italy

road network analysis, urban design, environmental assessments and autonomous navigation. In recent years, Deep learning (DL) models, and in particular Deep neural networks (DNNs), have demonstrated exceptional performance in understanding and extracting meaningful insights from complex datasets, including point clouds (Pierdicca et al. 2020; Zhang et al. 2019; Zhou et al. 2023). By leveraging deep architectures composed of multiple layers of interconnected artificial neurons, DNNs have the capacity to learn hierarchical representations of data, capturing both local and global patterns (Döllner 2020). When applied to point cloud data, DNNs have the potential to uncover semantic information (e.g. object recognition, segmentation, and scene understanding). They can learn to recognize objects, infer their shapes, classify different parts of a scene, and even predict physical properties or behaviours. The strength of DNNs lies in their ability to automatically learn features from raw input data, eliminating the need for explicit feature engineering. This makes DNNs well-suited for processing point cloud data, which can be highly complex and require intricate geometric representations, facilitating tasks such as object detection, semantic segmentation, and scene reconstruction (Lin, Kong, and Lucey 2018). However, especially in the GeoAI, they are often regarded as black boxes, making users blindly trust the outputs and, therefore, challenging to understand how they achieve their predictions or decisions. This lack of transparency raises concerns about the reliability, trustworthiness, and ethical implications of these models, particularly in safety-critical domains such as autonomous vehicles or healthcare (Goodman and Flaxman 2016; Heuillet, Couthouis, and Díaz-Rodríguez 2021). Explainable Artificial Intelligence (XAI) has emerged as a vital field of research, aiming to provide transparency and comprehensibility in the decision-making processes of AI systems. XAI focuses on developing techniques and methodologies that enable humans to understand, interpret, and trust the outputs generated by AI models. Several methods have been proposed to visualize and interpret the features learned by DNN models for images (Ahsan et al. 2020; Young et al. 2019). Techniques such as occlusion analysis (Ieracitano et al. 2021), saliency maps (Hsu and Li 2023), and class activation maps (CAM) (Poppi et al. 2021) have been extensively used to highlight the regions of the input image that contribute most significantly to the model's prediction. These approaches help researchers, practitioners, and end-users comprehend how the model focuses on relevant image regions and which visual cues influence its decisions. However, XAI for 3D data remains relatively unexplored, especially for the semantic segmentation task. The lack of explainability in DNNs poses a totally new perspective and challenge, as it hinders researchers from gaining insights and engenders scepticism due to their non-self-explanatory nature. While XAI approaches have been extensively studied for 2D data, only a few studies have attempted to investigate explainability for 3D DNNs.

In our previous work (Matrone et al. 2022), we introduced BubblEX, a multimodal fusion framework specifically designed to learn 3D point features and address the challenge of explainability in the context of point cloud data. BubblEX was developed with the aim of unravelling the black-box nature of 3D point cloud feature learning, enabling users to gain insights into the decision-making processes of deep neural networks. The framework was applied and evaluated on the classification task. Through comprehensive experimentation, BubblEX demonstrated promising results, showcasing its effectiveness in learning discriminative features from 3D point clouds, thus allowing users to understand the underlying relationships and interactions between points, leading to a better understanding of the decision-making processes employed by the model.

Given the significance and potential of BubblEX in the context of explainable 3D point cloud feature learning, we have continued our investigation by expanding its application to the task of semantic segmentation, which plays a pivotal role in understanding the fine-grained structure of point cloud data (Landrieu and Simonovsky 2018). By extending the capabilities of BubblEX to incorporate semantic segmentation, we aim to further enhance the interpretability and explainability of DNNs in the context of 3D point cloud analysis. This extension allows us to explore and understand how the geometric structure of the input data and the learned features contribute to the segmentation process, providing valuable insights into the model's decision-making mechanisms. In our experiments, we trained DNNs on four state-of-the-art datasets: S3DIS (Armeni

et al. 2016), SynthCity (Griffiths and Boehm 2019), Semantic3D (Hackel et al. 2017), and KITTI (Behley et al. 2019). Besides, we perform a comparative analysis of three different networks: PointNet (Qi, Su et al. 2017), DGCNN (Dynamic Graph Convolutional Neural Network) (Wang et al. 2019), and BAAF-Net (Qiu, Anwar, and Barnes 2021). These networks, purposely selected, represent distinct architectures for processing point cloud data and have demonstrated their efficacy in various 3D tasks, including semantic segmentation.

By addressing the explainability of AI models specifically in the context of indoor and urban point cloud analysis, our work could directly support the development of new data acquisition strategies in the field of geomatics, promote intelligent urban systems, and empower stakeholders and companies to make informed decisions that benefit both the environment and the communities residing in urban areas. In addition, this paper aims to investigate two fundamental aspects in the analysis of DNNs: the influence of different datasets and the importance of network architecture. Specifically, we want to understand if DNNs behave differently when trained on real or synthetic datasets, or when presented with outdoor or indoor datasets acquired through static or mobile systems. Additionally, we aim to evaluate whether the obtained results significantly depend on the chosen network architecture or if it is a negligible factor. Analyzing and discussing these findings, we will try to understand if it is possible to provide valuable insights and possible guidelines for the initial workflow phases, namely the data acquisition.

The main contributions of this paper, compared to state-of-the-art approaches, can be summarized as follows:

- extension of the BubblEX framework, which was previously proposed for 3D point feature learning, from classification to the task of semantic segmentation. By incorporating semantic segmentation within the framework, our study enhances the explainability of DNNs for this specific task, which is a novel contribution to the field;
- a comprehensive comparative analysis of three different networks (PointNet, DGCNN, and BAAF-Net) and their behaviours. This analysis thus provides valuable insights into these networks' performance, strengths, and weaknesses in the context of semantic segmentation for 3D point cloud data. Comparing these networks helps identify the most suitable architecture for the task at hand, which can guide future research and practical applications;
- comparison of four different benchmark datasets widely used in the field of 3D point cloud analysis, including S3DIS, SynthCity, Semantic3D, and KITTI. By evaluating the networks on these datasets, the study ensures a rigorous and standardized evaluation, allowing for meaningful comparisons with state-of-the-art approaches;
- visual explanations and highlights of the regions of interest within the point clouds, influencing the segmentation results. This analysis contributes to the understanding of the key factors and features that drive the networks' decision-making processes;
- the extensive analysis contributes to the establishment of possible guidelines for data acquisition and network configuration, enabling more effective decision-making in neural network-based applications;
- the first visual demonstration and explanation of well-known features influencing the DNNs performances and classification results.

The paper is structured as follows: in Section 2, we provide an overview of existing approaches adopted for explainability in 3D point cloud data. Section 3 presents the extension of BubblEX framework specifically designed for XAI in 3D point clouds. To evaluate the effectiveness of our approach, Section 5 presents a comparative evaluation using S3DIS, SynthCity, Semantic3D, and KITTI datasets. We compare our results against state-of-the-art techniques, and provide a detailed analysis of each component of our framework, highlighting its strengths and limitations. Subsequently, in Section 5, we engage in discussions based on the obtained results. We delve into the implications, insights, and potential applications of our findings. Finally, in Section 6, we

conclude the paper by summarizing the contributions and impact of our work and outline future research directions to further advance the field of explainability in 3D point cloud analysis.

2. Related works

This section provides an overview of the current state of XAI approaches, discusses classical DNNs for point cloud data, presents existing methods for XAI on point cloud DNNs, and highlights the options available for verifying the effectiveness of XAI approaches.

2.1. Explainability approaches

Most research on explainability primarily focuses on image classification tasks. Popular methods for explaining DNNs include gradient-based approaches and local surrogate model-based approaches.

Gradient-based approaches Analyze the gradient descent process during forward passes and are specifically applicable to differentiable models such as neural networks. Saliency Maps was the pioneering method that attempted to explain DNNs by computing the partial derivatives of each pixel, attributing importance accordingly (Simonyan, Vedaldi, and Zisserman 2013). However, standard gradients encounter issues of saturation (Sundararajan, Taly, and Yan 2016) and discontinuity (Smilkov et al. 2017). Integrated Gradient (Sundararajan, Taly, and Yan 2017), Layer-wise Relevance Propagation (LRP) (Bach et al. 2015), and DeepLIFT (Shrikumar, Greenside, and Kundaje 2017) address the saturation problem by estimating the overall importance of each pixel (Kim et al. 2019). SmoothGrad (Smilkov et al. 2017) tackles the discontinuity issue by smoothing the gradient using a Gaussian kernel that randomly samples neighbouring inputs and computes their average gradients. Guided Backpropagation (Springenberg et al. 2014) produces sharper gradient maps by removing negative attributions from the prediction.

Another set of approaches that utilize gradients is activation maximization, which aims to discover the ideal input distribution for a given class (global explanation) by optimizing input gradients while keeping all network parameters fixed, instead of explaining individual instances (local explanation) (Nguyen, Yosinski, and Clune 2019).

Local surrogate model-based methods, such as LIME (Ribeiro, Singh, and Guestrin 2016) and KernelSHAP (Lundberg and Lee 2017), aim to trace the decision boundary around a specific instance by perturbing input instances and utilizing surrogate linear models that approximate the performance of the original model but are more interpretable due to their simplicity.

2.2. 3D convolutional neural networks

Recent advancements in robotics and autonomous driving have sparked interest in 3D deep learning. Efficient processing of raw point cloud data is crucial for designing systems with low energy consumption and real-time behaviour, as point clouds are the primary data format obtained directly from most 3D sensors. Point clouds possess higher structural complexity compared to 2D image data due to their unordered nature, leading to a lack of neighbourhood consistency between data structures and spatial coordinates. This inconsistency results in not reproducible outcomes when applying convolution kernels directly to raw point clouds without pre-processing.

To address this, previous works have proposed approaches that transform and reorganize point clouds into voxels, extracting features using 3D convolution kernels (Maturana and Scherer 2015; Qi et al. 2016; Wu et al. 2015). Alternatively, some studies feed neural networks with polygonal meshed spatial information as a substitute for raw point clouds (Bruna et al. 2013; Masci et al. 2015). However, these pre-processing approaches are unsuitable for real-time scenarios and may not be advantageous for semantic segmentation tasks. In contrast, other studies propose point cloud-applicable convolutional networks that concatenate local features extracted by point-wise convolutional kernels with a global feature obtained through max-pooling layers (Qi, Su et al. 2017; Qi, Yi

et al. 2017). These approaches achieve state-of-the-art accuracies on the widely used ModelNet40 dataset (Wu et al. 2015), which is currently one of the most popular point cloud classification datasets.

2.3. Explaining 3D deep neural networks for 3D data

Limited research has explored the explainability of 3D DNNs for 3D data. While a study by Zhang et al. (2020) addresses explainable point cloud classification, their approach focuses on adapting point clouds to classical classifiers through pre-processing using PointHop Units. This does not provide post-hoc explanations. The authors in Zheng et al. (2019) obtain point saliency maps by simply dropping points, which is unrelated to explainability approaches. The pioneering study by Gupta, Watson, and Yin (2020) investigates the use of explainability methods for point clouds, shedding light on the sparsity of features in 3D models. However, their work only presents sparse explanations that emphasize points at edges and corners, lacking semantic information and an evaluation criterion. Furthermore, gradient-based methods are not suitable for models without gradients, such as tree-based models. In contrast, Tan and Kotthaus (2022) use a local surrogate model-based explainability approach, stating that they are completely model-agnostic and offer a more versatile solution; however, they test it on the classification task through ModelNet40, a dataset that lies completely outside of urban geospatial applications. Another similar and recent work (Arnold, Angelov, and Atkinson 2022) proposes a new classification method, the XPCC, which incorporates several layers of human-interpretable explainability, identifying object regions that mostly contribute to the classification. Nevertheless, also in this case, the selected dataset is ModelNet40, and the authors cite as a limitation of the work that the classification relies on point clouds containing only one object, highlighting the need to focus on real-world data and other point cloud-specific objectives, exactly what our contribution is aimed at. On the other side, Verburg (2022) takes a further step and starts to move from classification to semantic segmentation. The author tests PointNet++ for segmenting point clouds of catenary arches and demonstrates that this model mainly relies on the location of the objects in order to segment them, understanding that the change of the shape of an object does not have a significant impact on the performance of the model. This approach introduces relevant insights into how the network is taking a decision, though the proposed method is not generalizable and fully applicable to the state-of-the-art DNNs since it relies on specific part substitution experiments. The recent work of Atik, Duran, and Seker (2024) is one of the first that evaluates the proposed methodology on urban photogrammetric points clouds; however, the tested models are ML classifiers as Random Forest, or XGBoost, and other types of point clouds, as those acquired by Mobile Mapping Systems or LiDAR are not taken into account yet. These latest contributions further demonstrate the novelty of our work, which, to the best of our knowledge, is the first to combine the semantic segmentation tasks, applied both to real-world and synthetic data, making an extensive comparison, drawing guidelines and ensuring an extensive applicability and generalization capability.

2.4. Explanation plausibility verification

While there is a plethora of studies on explainability methods (for 2D data), there is a lack of acknowledged quantitative assessment for these approaches (Burkart and Huber 2021), primarily due to the subjective nature of explanations as perceived by humans. The authors in Adebayo et al. (2018) argue that a feasible explanation should be sensitive to model weights and the data generation process. They propose an alternative evaluation approach by randomizing network weights and labels to examine the sensitivity of saliency maps. However, this approach tends to favour gradient-based explainability methods and validates invalidity rather than feasibility. In Cian, van Gemert, and Lengyel. (2020), the authors aimed to observe the improvement in the core performance of the network and the confidence generated by system users when processing image data. An intuitive and efficient pattern to verify explanations by flipping pixels that contribute positively,

negatively, or approximately zero to a specific class, recording the verified prediction scores has been proposed in Bach et al. (2015), Montavon (2019) and Samek et al. (2016). Nevertheless, the flipping operation in this method could be optimized when processing point cloud data.

Considering the state-of-the-art in this field, in a previous paper, we proposed a BubblEX framework specifically designed for learning 3-D point features. In fact, it adopts a multimodal fusion approach, which combines information from multiple modalities within the point cloud data to facilitate feature learning. By incorporating explainability modules within the framework, BubblEX enables to gain insights into how neighboring points contribute to the feature extraction process.

The BubblEX framework comprises two stages: the 'Visualization Module', which visualizes features learned from the network in its hidden layers, and the 'Interpretability Module', which aims to describe how neighbouring points contribute to the feature extraction process (Matrone et al. 2022). The framework was applied and evaluated on two benchmark datasets, namely Modelnet40 (Wu et al. 2015) and ScanObjectNN (Uy et al. 2019), known for their importance in 3D object classification tasks. Through comprehensive experimentation, BubblEX demonstrated promising results, showcasing its effectiveness in learning discriminative features from 3D point clouds.

In our following paper (Matrone et al. 2023), we extended the application of the BubblEX framework to a publicly available Digital Cultural Heritage Dataset (DCH) known as the ArCH (Architectural Cultural Heritage) Dataset (Matrone et al. 2020). This dataset presents a more challenging domain, characterized by the complexity inherent in architectural cultural heritage. By applying BubblEX to the ArCH dataset, we aimed to tackle the intricacies and nuances of the built heritage domain, which often involves intricate structures, intricate designs, and diverse architectural styles. This dataset provided an ideal testbed to evaluate the robustness and effectiveness of the BubblEX framework in capturing and explaining the features relevant to architectural DCH; however, since it has been tested on only one dataset, it did not allow a proper generalization.

Building upon our previous works, this paper focuses on the application of the BubblEX framework to the task of semantic segmentation, aiming to enhance the interpretability and explainability of DNNs in the context of 3D point cloud analysis. Semantic segmentation is of utmost importance as it enables a detailed understanding of the fine-grained structure and meaning within point cloud data. By extending the capabilities of BubblEX to incorporate semantic segmentation, we unlock the ability to explore and comprehend how the learned features contribute to the segmentation process. This extension provides valuable insights into the decision-making mechanisms of the DNN models, facilitating a deeper understanding of their inner workings and promoting transparency in their predictions.

3. Methods

This section outlines the methodology employed in our study to evaluate the performance and suitability of different networks for to develop an approach for XAI in the context of urban point cloud analysis.

Through this methodology (Figure 1), we aim to provide a comprehensive analysis of the selected networks and their suitability for urban point cloud analysis, enabling informed decision-making and recommendations for real-world applications. By following a structured approach and considering multiple aspects, we ensure a robust and reliable framework that provides transparent insights into the features and patterns learned by the networks.

The following subsections provide further details regarding the experimental setup, dataset selection, network architectures, and evaluation metrics.

3.1. Urban and indoor point cloud datasets

The selected datasets for testing in this study are S3DIS (Armeni et al. 2016), SynthCity (Griffiths and Boehm 2019), Semantic3D (Hackel et al. 2017), and KITTI (Behley et al. 2019). S3DIS contains 272 scenes of university rooms, grouped by areas or buildings, annotated with 14 classes and the

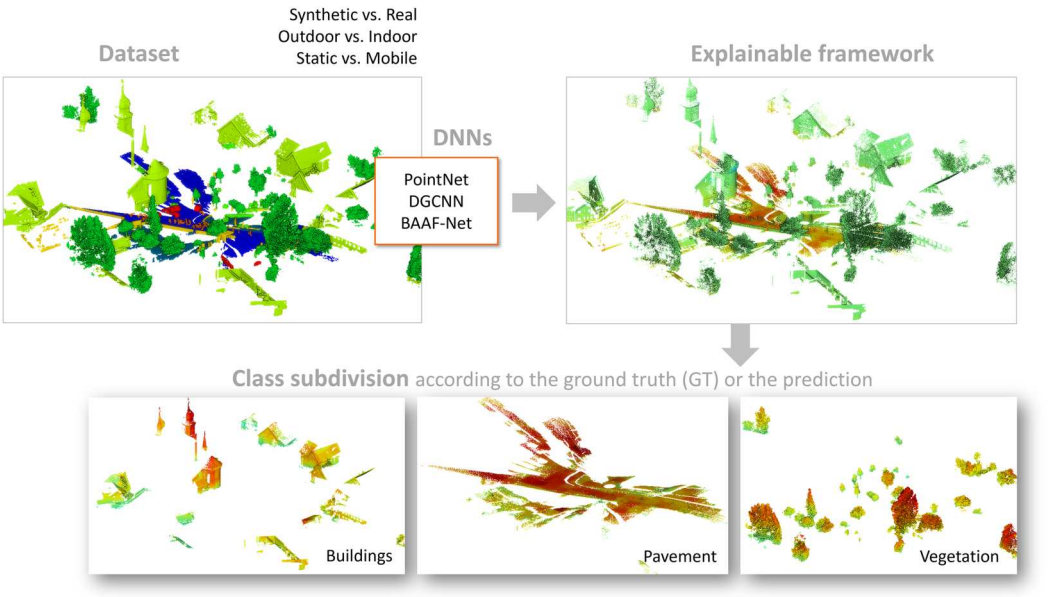

Figure 1. Overall workflow for the developed methodology.

same features as Semantic3D. Test Area 5, consisting of 3 rooms, was used for evaluation. SynthCity consists of 9 scenes of synthetic urban environments with 9 classes. The features used include 'x', 'y', 'z', 'x_noise', 'y_noise', 'z_noise', 'R', 'G', 'B', 'time', 'eol' (end of line), and 'label', with Test Area 3 chosen for evaluation. Semantic3D comprises 9 scenes of real urban environments with 8 classes and features such as 'x', 'y', 'z', 'I', 'R', 'G', 'B', and 'label'. Test Area 3 was also selected for evaluation in this dataset. The KITTI dataset, which provides Velodyne point cloud data, was processed using features 'x', 'y', 'z', 'R', 'G', and 'B'.

The selection of these four datasets enabled comparisons across different aspects (Figure 2):

- synthetic datasets (SynthCity) vs. real datasets (S3DIS and Semantic3D) in order to investigate if features such as the noisiness or the density of the point cloud data somehow affect the output of the DNNs;

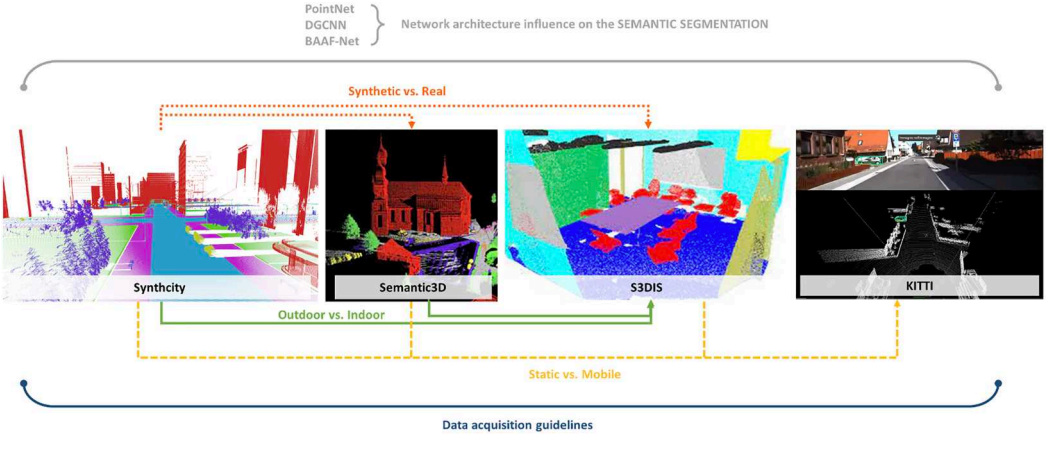

Figure 2. Dataset comparison strategies.

- indoor datasets (S3DIS) vs. outdoor datasets (Semantic3D and SynthCity), to examine whether the extension of the analysed elements and the proximity to the acquiring sensors, with implications for density, could impact the predictions;
- datasets acquired through static systems (Semantic3D) vs. dynamic systems (KITTI) to investigate the possible generalization of the method to other datasets generated with diverse sensors and with different characteristics.

To facilitate a proper comparison among the datasets, the different categories were matched and, in some cases, grouped based on the similarity of geometries, as shown in Figure 3 and in Table 1.

3.2. Deep neural networks for urban point cloud

In order to assess the performance and effectiveness of our approach, we selected three state-of-the-art networks: PointNet (Qi, Su et al. 2017), DGCNN (Wang et al. 2019), and BAAF-Net (Qiu, Anwar, and Barnes 2021). Each network was chosen for specific reasons based on their architectural characteristics and their performance. PointNet was chosen as it is a pioneering network in the semantic segmentation of point clouds. DGCNN, on the other hand, was selected due to its graph-based architecture, which differs from the previous network. To choose the third network, a comparison was made between the benchmarks of various datasets, particularly Semantic3D and S3DIS, as an extensive comparison for SynthCity was not yet available. A common and recent network that demonstrated the best results with both datasets (with code 2023b, 2023a) was thus selected, which in this case resulted in the BAAF-Net. All the DNNs employed cascade convolutional layers to extract deep features from the data. However, they differ in how the data is processed in the early layers immediately after the input, in the grouping of points to extract local and/or global features, and in the recombination of point-wise, local, and global features. PointNet aggregates point features using max pooling to obtain global features. The segmentation network concatenates

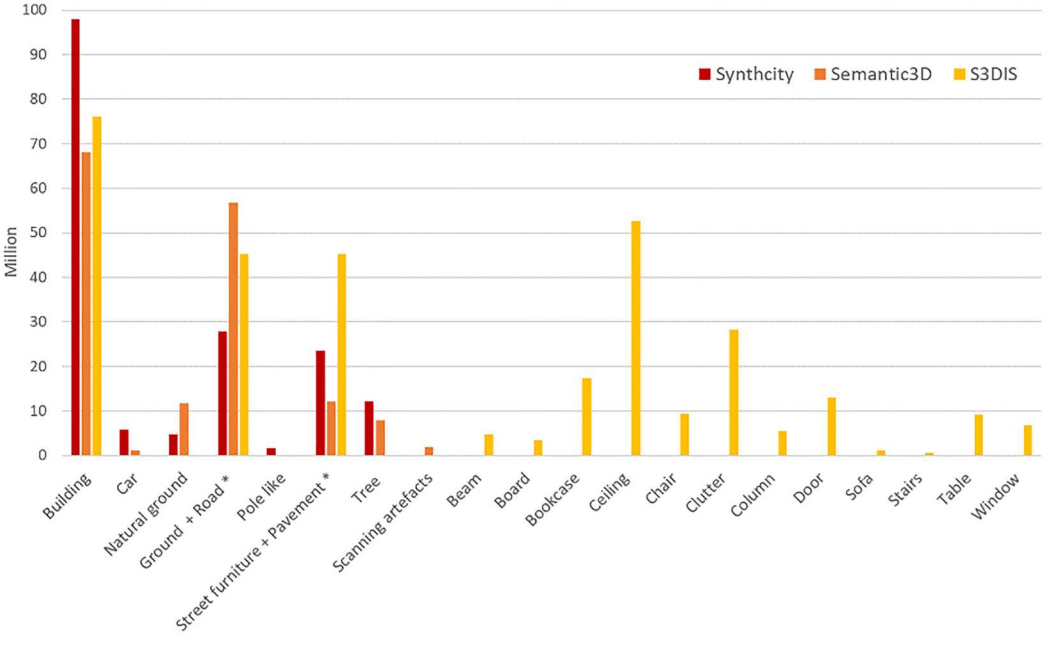

Figure 3. Number of points, divided by class, within the selected datasets. The classes with asterisks constitute the union of two Synthcity classes to allow the aggregation of as many Semantic3D classes (see Table 1). In addition, in the above classes, the number of S3DIS points was repeated as they were analysed in both Road and Pavement.

Table 1. Comparison and matching between the various classes in the datasets.

Synthcity	Semantic3D	S3DIS	KITTI
Building (0)	Building (5)	Wall (11)	Buildings (0)
			Other buildings (9 / 10)
Car (1)	Cars (8)		Cars (parked 2)
			(moving 17)
Natural ground (2)	Natural terrain (2)		Terrain (13)
Ground (3)	Man-made terrain (1)		
Pole-like (4)			Poles (14)
			Traffic sign (15)
Road (5)	Man-made terrain (1)	Floor (7)	Road (5)
			Parking (6)
			Sidewalk (7)
			Other ground (8)
Street furniture (6)	Hardscape (6)		
Tree (7)	High vegetation (3)		Vegetation (11)
	Low vegetation (4)		Trunk (12)
Pavement (9)	Hardscape (6)	Floor (7)	
	Scanning artefacts (7)		
		Beam (0)	
		Board+bookcase (1)	
		Ceiling (2)	
		Chair (3)	
		Clutter (4)	
		Column (5)	
		Door+Window (6)	
		Sofa (8)	
		Stairs (9)	
		Table (10)	
			Truck/Bus (3)
			Other Object (16)

Notes: The numbers in parentheses represent the corresponding class numbers. The classes from different datasets have been compared and matched based on their semantic similarities and geometries. The table provides an overview of the corresponding classes across the datasets, facilitating a comprehensive analysis and comparison of the class distributions and semantic labels within the urban point cloud datasets.

the global and local features before the final (point-wise) classification. PointNet applies rigid and/or affine transformations to canonicalize the input data. These spatial operations independently transform the input points. DGCNN implements a dynamic graph-based neural network using edge convolution to learn point neighbourhood features. It uses the k-nearest neighbours to select the neighbouring points. This module captures local geometric structure while maintaining permutation invariance. It also generates neighbourhood features that describe the relationship between a point and its neighbours rather than purely point-wise features. BAAF-Net is based on RandLA-Net (Hu et al. 2020) and utilizes random sampling. BAAF-Net integrates a bilateral upsampling structure to process the point cloud at different resolution levels and uses an adaptive fusion method to represent complete point-wise, local, and global features. In addition, the choice of these DNNs is also reflected in other recent works that tested PointNet and/or the DGCNN as a benchmark for XAI techniques evaluation (Arnold, Angelov, and Atkinson 2022; Feng et al. 2024; Levi and Gilboa 2024) on object classification. For example, in their latest paper, Levi et Gilboa compare both PointNet and the DGCNN, along with Robust Point-Cloud Classifier (Xu et al. 2021) and the Geometry-Disentangled Network (Ren, Pan, and Liu 2022) still for classification purposes on the ModelNet40 dataset.

4. XAI approach for urban and indoor point cloud analysis

The tests were conducted by combining the three DNNs and the four datasets. In each test, a model was trained on a point cloud dataset to solve the semantic segmentation task. Subsequently, activations and resulting gradients were extracted from the trained model for an input scene. The

results were then visualized using the implemented BubblEX method based on Grad-CAM (Gradient-weighted Class Activation Mapping) (Selvaraju et al. 2017).

For each dataset, a predefined training and test set were assigned. Specifically, the scenes used for training the model were assigned to the training set, while the scenes used for model validation and the extraction of activations and gradients were assigned to the test set. Once an architecture and dataset are selected, the model is trained. The structure of the model is based on the dataset characteristics. The input size for all datasets and models is set to 4096 points with 6 features, including 3 geometric features (representing the x, y, z coordinates) and 3 colour features (representing the R, G, B values, respectively). The output layer size is set based on the number of classes, equivalent to 4096 points for n classes. Since scenes of arbitrary length (number of points) need to be processed, a block-wise sampling (sliding windows) approach is employed. Each block is selected from a prism of a square area lying on the xy plane, unlimited in height, which slides across the entire plane. The block's metric dimensions (size) are fixed for each dataset, as well as the stride parameters along the x and y axes. Subsequently, the selected block is down-sampled to 4096 points: if the original block contains more than 4096 points, random decimation is performed to select 4096 points; if it contains fewer than 4096 points, the original points are randomly repeated to form a block of 4096 points. Blocks with less than 100 points are discarded. The obtained down-sampled blocks containing both input data (x, y, z, R, G, B, features) and ground truth labels constitute the training set and test set. The training process was initiated after setting the training hyperparameters: batch size of 8, Adam optimization algorithm with an initial learning rate of 0.001 and momentum of 0.9, minimizing the cross-entropy loss function, and 100 epochs of training.

After that, the adapted principles of Grad-CAM have been applied. In particular, Grad-CAM allows the highlighting of the regions of an input image that are most influential in the model's predictions. Grad-CAM computes the gradients of the predicted class score (output) with respect to the feature maps of the last convolutional layer. These gradients represent how much the output score would change with respect to small changes in each feature map. Then, the gradients obtained in the previous step are globally averaged across each feature map. Grad-CAM calculates a weighted sum of the feature maps, where the weights are determined by the global average-pooled gradients. This step highlights the regions of the input image that had the most significant impact on the model's decision. Finally, it generates a heatmap by applying a ReLU (Rectified Linear Unit) activation function to the weighted sum of feature maps.

Activations and gradients are, therefore, extracted from the trained model. For each architecture, the layer for analysis is selected. The deeper layers (closer to the output) were chosen as they learn more discriminative features with higher semantic content. For all three considered architectures, the last convolutional layer before the output layer was chosen as the extraction layer. To extract the activation of a model, a test scene is selected and processed (block-wise) by the model. The activation extracted from the selected activation layer for each block of the scene has dimensions of 4096 points for m extraction layer features (Figure 4). In particular, the feature dimensions are 128 for PointNet, 256 for DGCNN, and 32 for BAAF-Net. The blocks are subsequently recombined to reconstruct the overall scene.

The activation can be visualized using a point cloud where points are coloured using a JET colourmap representing the intensity of the activation. In particular, the JET colourmap, used to represent in colour a symmetrical, normalized, and bipolar range of values (from -1 to 1), associates blue with values close to -1, green with values around 0, and red with values close to 1. For this purpose, the activation itself (a matrix of 4096 points for m feature maps) is flattened along the feature dimension using the median function, resulting in a vector of dimension 4096 points, with values representing the activation intensities at each point. Analyses have been conducted to choose the median function as the flattening method for the features.

The obtained values are then normalized using a linear mapping between -1 and 1 based on the highest value among the absolute minimum and maximum values. Consequently, at least one point is mapped to either 1 or -1. To make the system more robust to outliers, points with values below

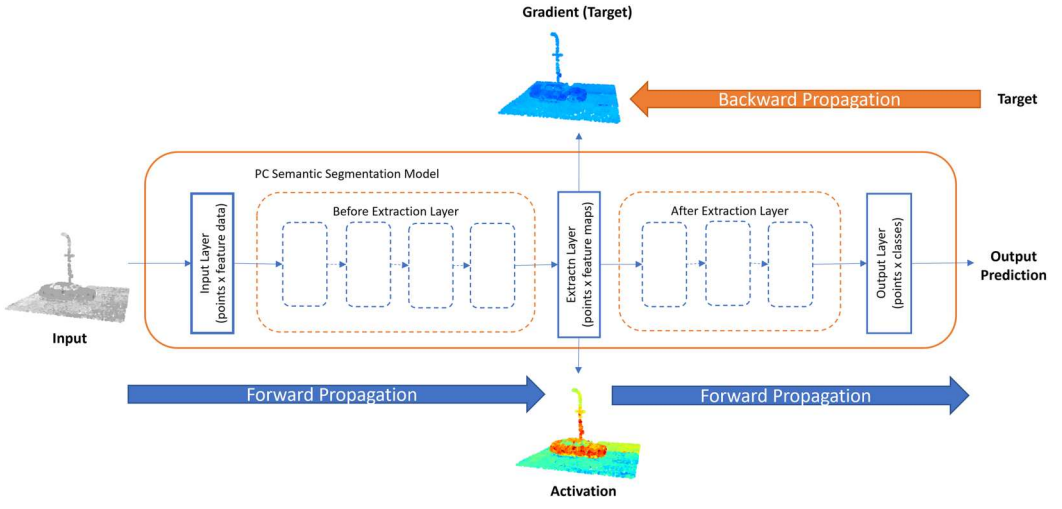

Figure 4. Workflow for the feature extraction with respect layers used for the activation and the target.

the 0.001 quantiles and above the 0.999 quantiles were removed. It is important to perform normalization only after scene recombination and not vice versa. Recombining the scene before normalization creates significant colour gradient discontinuity between blocks. However, the effects of gradient discontinuity between adjacent blocks remain visible even after global normalization across the entire scene. This is inherently due to the block-wise analysis of the point cloud. To mitigate this effect, experiments have been conducted on partially overlapping non-adjacent blocks (half overlap). This strategy improves visualization by making the colour gradients more continuous between adjacent blocks. Unlike activation, the gradient evaluated on a test scene is class-specific. To extract the gradient, a one-hot encoding signal identifying the target class is multiplied by the output vector and backpropagated to the extraction layer, similar to the error backpropagation during network training. This yields the gradient of the output associated with the target class with respect to the activation. Like the activation, the gradient is a matrix of 4096 points for m feature maps. The implemented Grad-CAM is obtained by taking the element-wise product between the activation matrix and the gradient (Figure 5, which is then flattened, normalized, and colour-coded as described for the activation.

Analyses have been conducted to choose whether to multiply the activation and gradient and then flatten or flatten the activation and gradient and then multiply. In BubblEX a statistical analysis

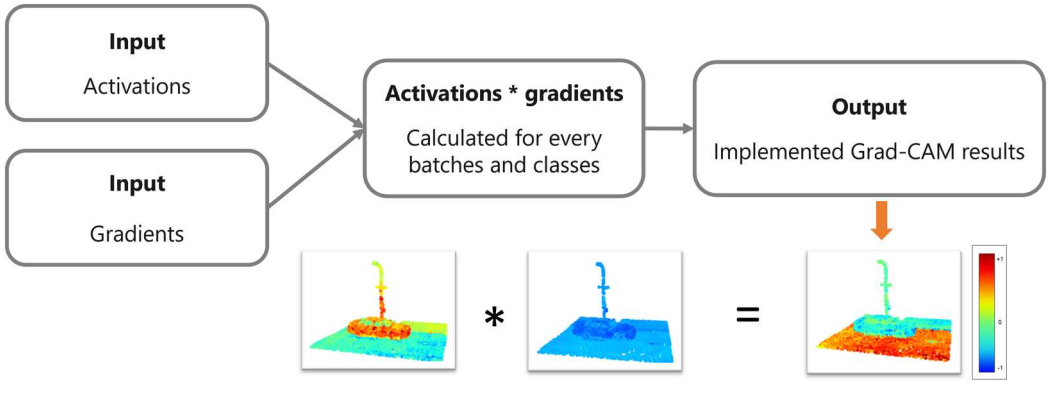

Figure 5. Generation of the final result with the salient points.

was conducted to choose the best function to flatten the size of the features. The ensemble functions tested are those frequently used in networks (pooling layers) to flatten the dimensions: minimum, maximum, mean (equivalent to the normalized sum), and median. Some samples were selected, and for these, the activation and gradient at the 'conv5' layer were extracted. For activation (matrix 1024 features x 1024 points), the distribution of values between the features for each point was evaluated. Among the functions to flatten the size of the features, the median seemed to give better results than the others, at least visually, especially in the Grad-CAM representation. It has been verified that the gradient between the features at each point has a very sparse distribution, and for this reason, the average value is very susceptible to outliers, which move it away from the centre of the distribution. While the median value is more robust. Using the median to flatten both activation and gradient before multiplication does not add enough emphasis; the nuances obtained are almost identical to those of activation alone. Differently, the use of the median to flatten the multiplication of activation and gradient creates a significant variation in nuances compared to those of activation alone.

Unlike (Matrone et al. 2022), which focused on classification tasks, the 'visualization module' was not one of the cores of this study. There was no search for misclassified objects in the 2D feature space by assessing their proximity to cluster centroids and identifying 'intruders'. Since semantic segmentation is based on point-wise classification, this analysis visualizes individual points in the scene rather than entire objects and only a few clusters have been evaluated when investing the misclassified elements. The activation is independent of the target class, while Grad-CAM modulates the activation with the target class-conditioned gradient. The intensity of the activation, regardless of its sign (positive or negative), indicates which points (local geometry) the network focuses its attention on for making the final decision. Grad-CAM also indicates how much the variation in local geometry influences the decision towards a target class, and its insights have been used for BubblEX.

Moreover, in classification networks, global-level features are learned in the final layers, resulting in the loss of spatial information. In segmentation networks, however, spatial information (x, y, z coordinates of the points) is preserved from the beginning to the end. In this approach, the features learned in the layers immediately before the final (point-wise) classification are utilized.

In the segmentation task, the spatial/geometric information of the point cloud is preserved (it is equivalent to a point-by-point classification). In the classification task, however, the information is lost in the model, which flattens and compresses the information in the deeper layers to make an overall estimate. However, in both cases, we can conduct explainability analyses with our method by selecting a layer from the model that maintains the geometric information of the point cloud till the output.

In a segmentation problem, analyzing activation alone may not be sufficient to evaluate the network's attention towards discriminative geometries, as done in a classification problem, where the network focuses on individual objects. For example, to distinguish between a tree and a car, the network's attention may be directed towards the trunk or the wheels, respectively. In the segmentation problem, there are no separate objects but rather scenes, and it is the network's task to infer the objects. Although an area with higher activation may indicate that the network is observing that particular feature, Grad-CAM analysis is necessary to verify if the network indeed focuses its attention on trees when the target is a tree and on cars when the target is a car. Additionally, it allows for further analysis of which parts of each class are more attended to, such as wheels, body, and windows for cars, and trunk and canopy for trees.

The extraction of activations and gradients does not affect the model's prediction, but their evaluations can explain the reasons why points in an input scene are predicted correctly or incorrectly. Therefore, the proposed visualization approaches can be used to intervene in the network's structure to make it more efficient. Pooling operations are necessary for flattening matrices (points by features) into point vectors during activation and gradient analysis. These operations, performed after the extraction of activations and gradients, do not affect the network's structure. All these analyses can be performed at any layer as long as the geometric information of the scene (x, y, z

coordinates) is preserved. The proposed approach can be extended to any other architecture, as it is independent of the specific type of chosen DNN. The overall accuracy in some tests (depending on the dataset and architecture) does not exceed 75%. Therefore, in the Grad-CAM plots of the scenes in the test set, the colourization may appear rather random. However, the purpose of this work is not to achieve maximum accuracy but to evaluate the model's performance using the proposed visualization methods.

5. Results and discussions

In the following section, we present the results of our analysis, organized according to the different classes under study. Additionally, we examine the performance of the DNNs chosen for the evaluation. However, it is important to note that the focus of this study is not on improving the DNN architectures but rather on understanding their behaviour, learning patterns, and how they analyze point clouds to make class predictions. Consequently, the aim is to gain insights into the decision-making process of the networks and the factors influencing their classification outcomes. Both the DGCNN and BAAF-Net architectures have shown promising results, with their performance alternating depending on the specific tests conducted, particularly in cases involving *overlap* or *no overlap* of the analysing blocks. By examining these results, we can draw meaningful conclusions and engage in insightful discussions about the findings.

5.1. Classes comparison

For the comparison between classes, the test scenes were plotted and divided according to their labels for each target class based on the ground truth points. The total number of point clouds so obtained has been 3336, divided into 1410 for the tests without overlap and 1926 for those with the overlap. They could be further categorized based on the DNN, resulting in 1344 for PointNet, 1320 for the DGCNN and 672 for the BAAF-Net, including both the overlap and no overlap settings. These numbers provide a comprehensive understanding of the scope and scale of the analysis conducted. This information is essential for assessing the data coverage and ensuring a comprehensive evaluation of the DNN performance.

From the comparison of the tested scene with and without *overlap* (Figure 6), the stripes caused by the analysis blocks are evident. It can be observed that the overlap averaging makes the result more homogeneous while still maintaining the overall trend of the Grad-CAM. For this reason, the analyses were mainly carried out with overlap.

In this contribution, only the scenes whose implemented Grad-CAM corresponds to the ground truth class are shown. However, an equal number of scenes were plotted based on the predictions as well. This allowed for further investigation in cases where the class was mostly misclassified, to determine which category it was confused with and potentially identify the reasons behind the misclassification.

5.1.1. Buildings

In this category, the points belonging to the Building class of Synthcity and Semantic3D, the Wall class of S3DIS, and the Building class of KITTI were selected and evaluated.

Considering the synthetic dataset (Figure 7), where the behaviour of the networks is more similar, the results show that the analysis of this class is mainly based on the vertical component of the scenes, with a greater emphasis on the higher points.

This trend is also observed in Semantic3D, especially with PointNet, but less so in DGCNN and BAAF-Net (Figures 8 and 9). The latter behaviour is more similar to that of the indoor environments in S3DIS (Figure 10), where the Z component is less pronounced. However, even in this case, PointNet exhibits a stronger development of saliency points upwards.

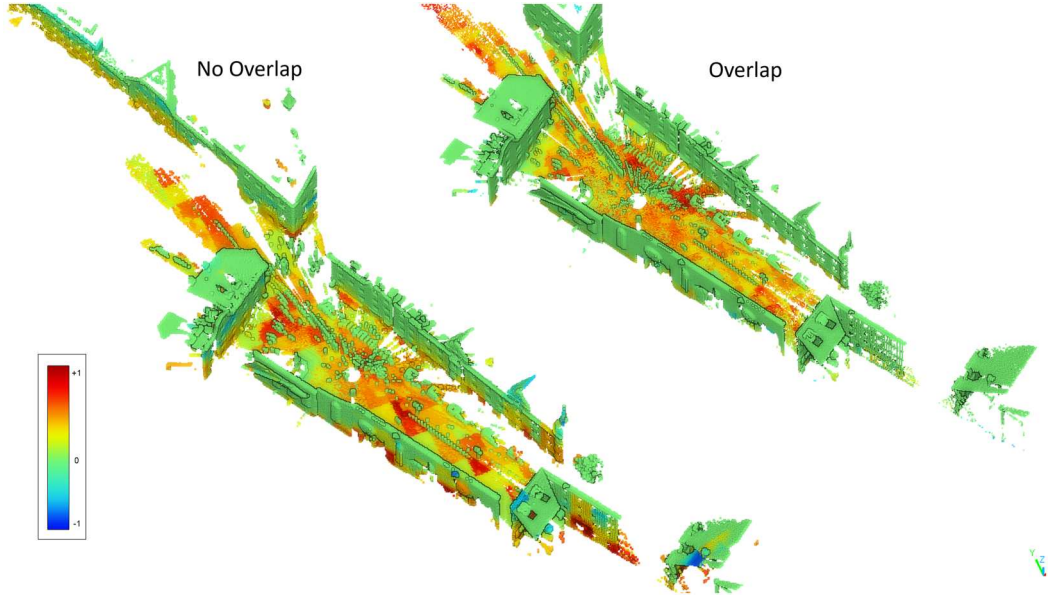

Figure 6. Comparison between the *no overlap* and the *overlap* setting for a scene of the Semantic3D dataset.

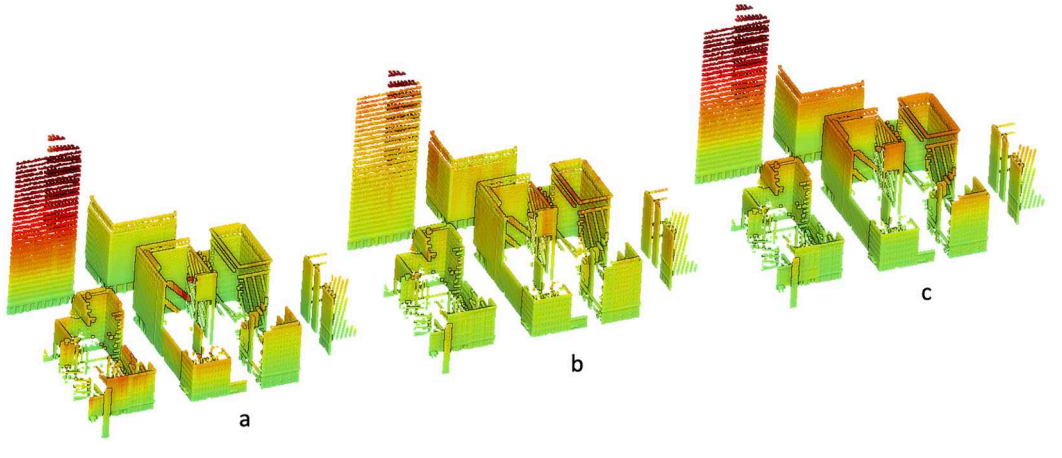

Figure 7. Synthcity: (a) PointNet (b) DGCNN (c) BAAF-Net. No Overlap.

In the KITTI dataset (Figure 11), despite the less dense point clouds and the greater influence of endless blocks analysis on the variation of values, the same tendency is observed in some parts. On the other side, when comparing the behaviour between outdoor and indoor datasets, it can be noted that the results are quite similar. In fact, Figures 9 and 10 show a correlation depending on the analyzed network: PointNet highlights a greater gradient between the lower and higher parts of the walls, which is less pronounced and almost 'blurred' in DGCNN and BAAF-Net. The KITTI dataset, being more noisy, has different results depending on the analyzed areas (Figure 11).

5.1.2. Floor – pavement

This category includes the classes 'Floor' for S3DIS, 'Ground' and 'Road' for Synthcity, 'Man-made terrain' for Semantic3D, and 'Terrain', 'Road', and 'Sidewalk' for the KITTI dataset.

Figure 8. Semantic3D: *bildstein_station3* and *domfountain_station3* scenes. No overlap, PointNet.

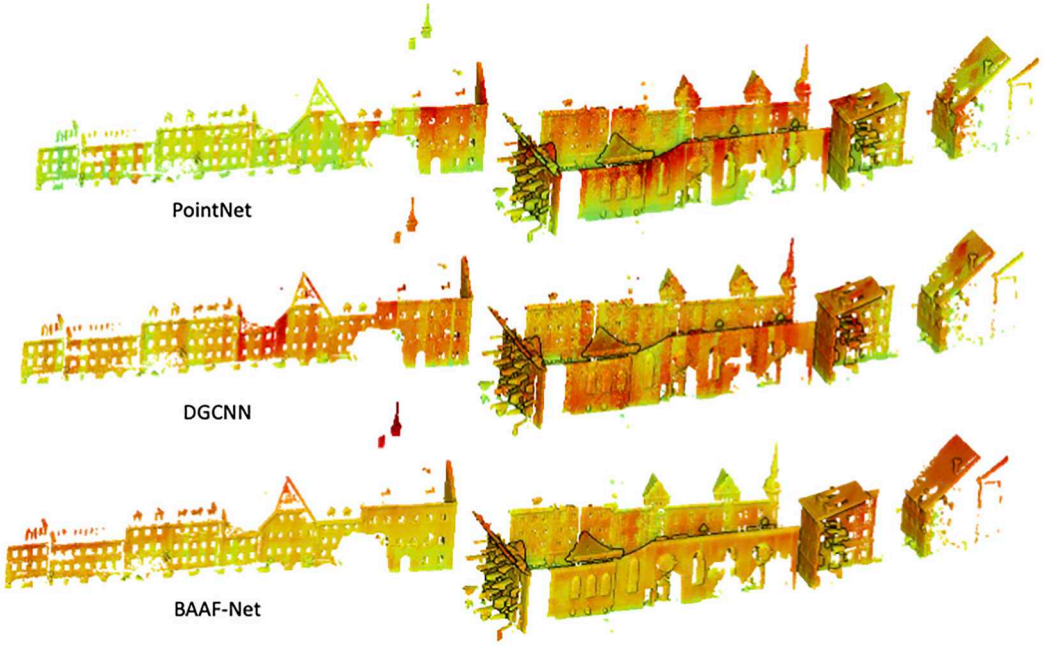

Figure 9. Semantic3D: *domfountain_station3* scenes. Overlap.

From Figure 12, it can be observed that in the case of indoor environments (S3DIS), the behaviour is similar across scenes regardless of the network. However, in outdoor environments, the results vary depending on the type of acquisition. In fact, in Semantic3D, a clear and defined pattern is not evident. However, in the KITTI dataset, where the point density varies greatly depending on the distance from the sensor, the most salient points are those with higher density. This allows to understand that density is a crucial factor in scene analysis, enabling the DNN to better and more comprehensively analyze the neighbors. However, density does not have an impact when it is made homogeneous through the sub-sampling performed by the DNN during the scene analysis.

If we compare the real datasets (Figure 12) with the synthetic ones (Figure 13), it can be noticed that the absence of noise, scene regularity, and a well-defined and unambiguous radiometric

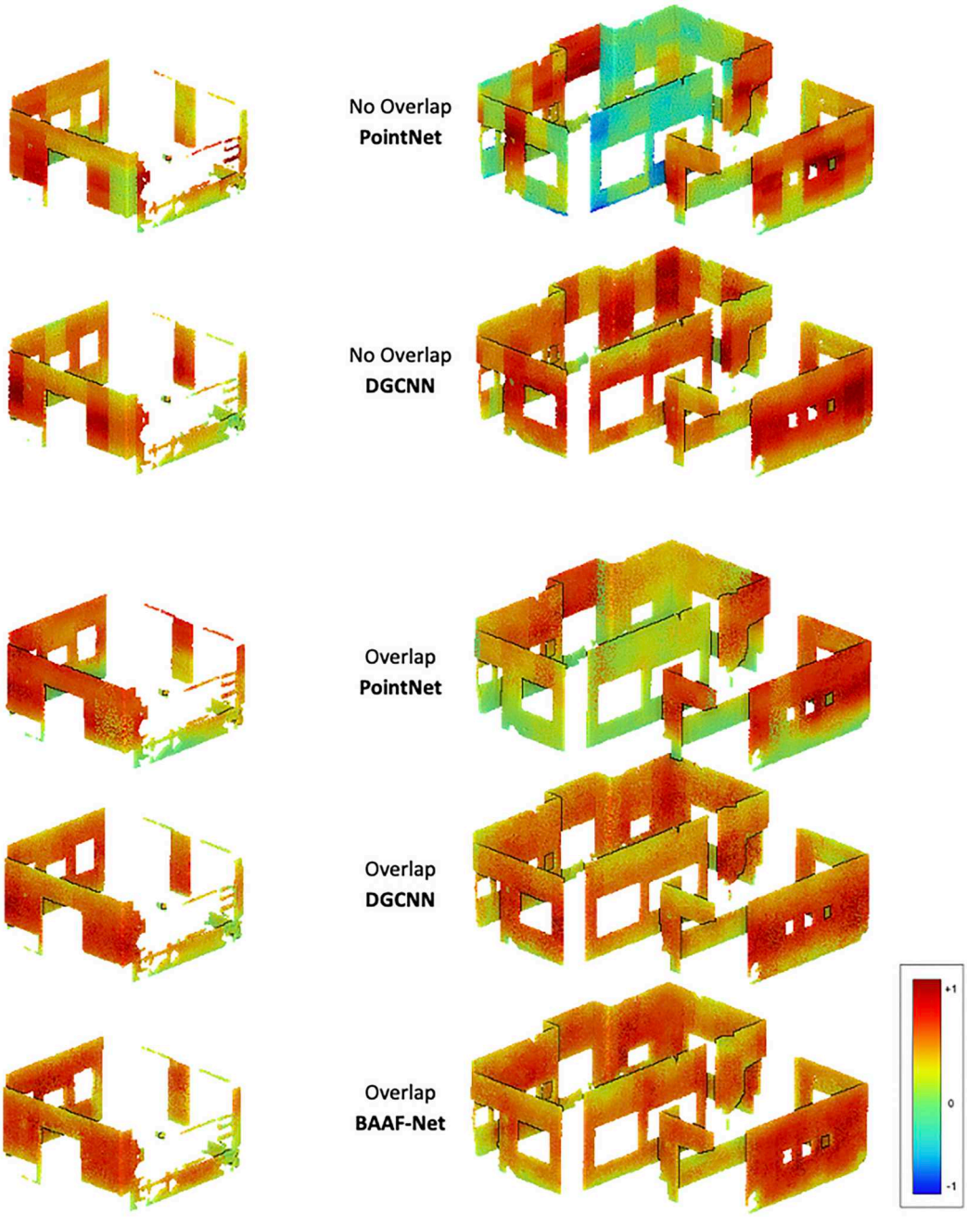

Figure 10. S3DIS. Comparison of *overlap* and *no overlap* for different networks.

component description are factors that help the network better discriminate between classes. Despite some differences in the results among the networks, the salient points are clearly identified, and urban elements such as horizontal road markings or sidewalks can be easily read. This confirms the findings of previous studies (Pierdicca et al. 2020) that highlighted the high importance of the radiometric component and emphasizes how the noise in the point cloud is a fundamental factor for accurate class recognition.

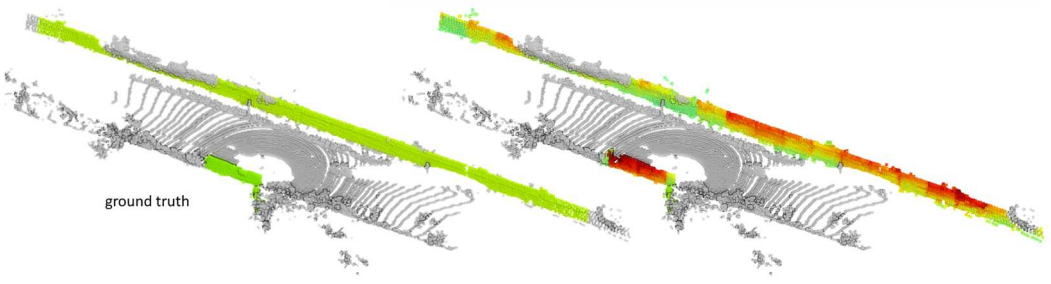

Figure 11. KITTI dataset: output of the ground truth (left) and predicted scene (right) with the DGCNN.

5.1.3. Tree – vegetation

The tree and vegetation classes fall into this category. If we analyze Semantic3D (Figure 14), we can see that the pattern of salient points is almost the same for both PointNet and DGCNN, although with some slight differences. The gradient, in fact, proceeds from bottom to top in both cases and seems more pronounced where the tree canopy is larger in size. This pattern is also found in the Synthcity dataset when processed with PointNet; however, when processed with DGCNN, it is much less visible and more homogeneous. Different discourse, however, for BAAF-Net, in which the tree trunks clearly emerge, while the canopy is more regular and has no saliency peaks. For the KITTI dataset, instead, the noise and density of the point cloud did not allow the definition of a readable pattern.

5.1.4. Car

As in the previous case, the results show that the radiometric component and the definition of the point cloud play a key role. In Figure 15, it can be seen that the wheels and the car roof, when analyzed with DGCNN, are the points of greatest interest for the network.

The developed methodology, however, makes it also possible to investigate, where there is an error in the prediction, what it is due to, whether it is due to the geometry, the features that describe it, or an incorrect annotation of the dataset. Figure 16 shows the visualization in a 2D space of the learned features and allows to hypothesize the reason for the error. In this case, the learned and descriptive features of the misclassified car, are correct, as they are positioned close to the other points in the cluster. This indicates that the features are close in feature space and, therefore, similar to each other. Thus, the error in the network could lie in the final classifier, in the incompleteness of the point cloud, or even in the geometrical proximity to other classes.

Figure 16 also shows how the salient points of the two misclassified cars do not correspond, or are at least similar, to those correctly classified.

If we then consider Figure 17, we can clearly notice that the salient points are highly net and density-dependent. In fact, analyzing only Synthcity, the car windows, rather than the car body, seem to be the parts of most attention with PointNet, while with BAAF-Net, the car roof and the back side, and with DGCNN, the roof and the wheels (Figure 15). If, on the other hand, Semantic3D is considered, the shortcomings on the point cloud make it more obvious how the pattern distinctly identified in Synthcity is no longer present here. It remains, on the other hand, constant with DGCNN, and this may be due to the inherent architecture of the network. A separate discussion should be made for the KITTI dataset, where it is possible to glimpse some patterns only where the density of points acquired allows it.

5.1.5. Hard scape

In this class, as in the previous one, the dependence of the results on network architecture and geometry is even more noticeable, since the behaviours are the opposite according to the DNN. In Figures

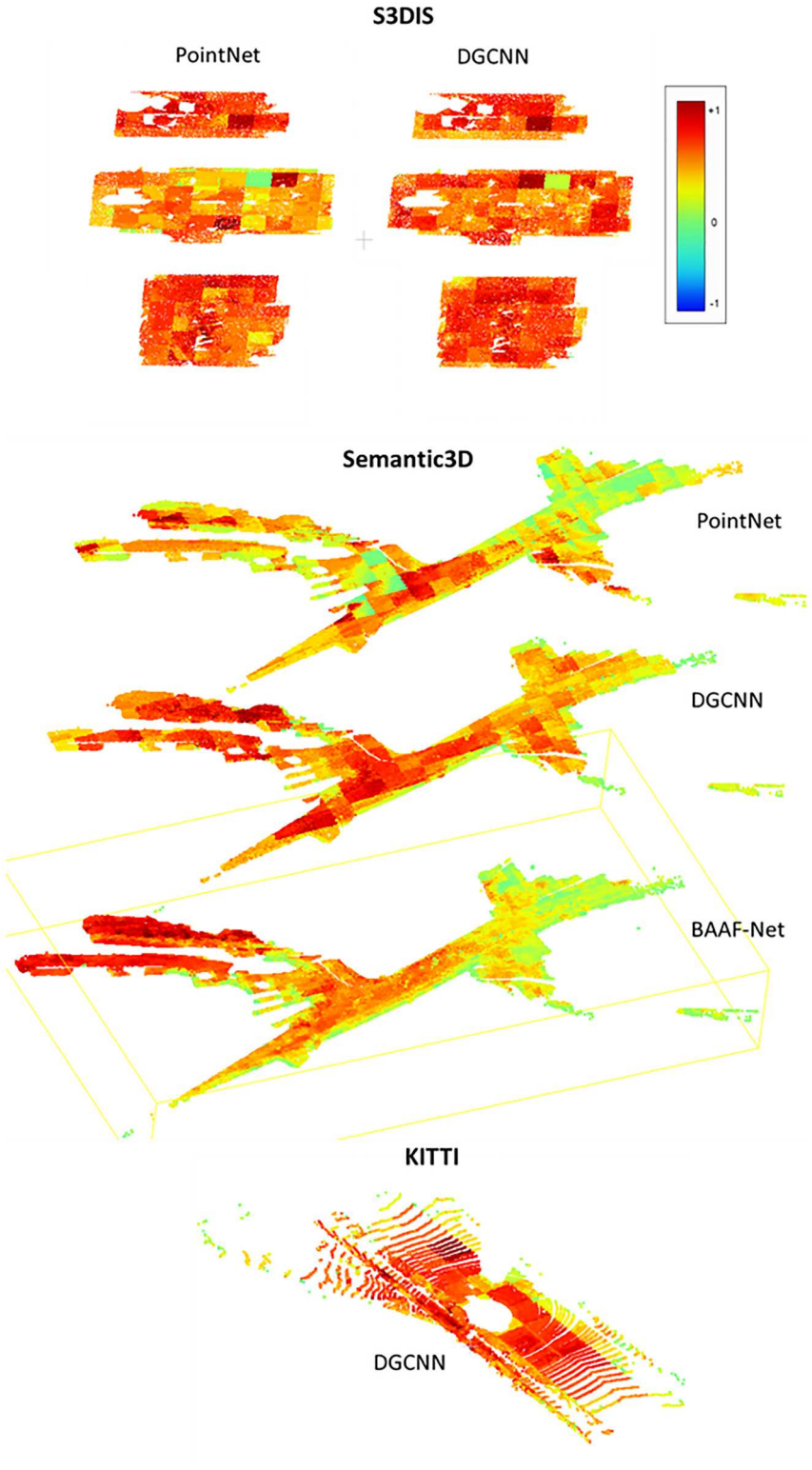

Figure 12. Comparison between the real datasets, both indoor and outdoor.

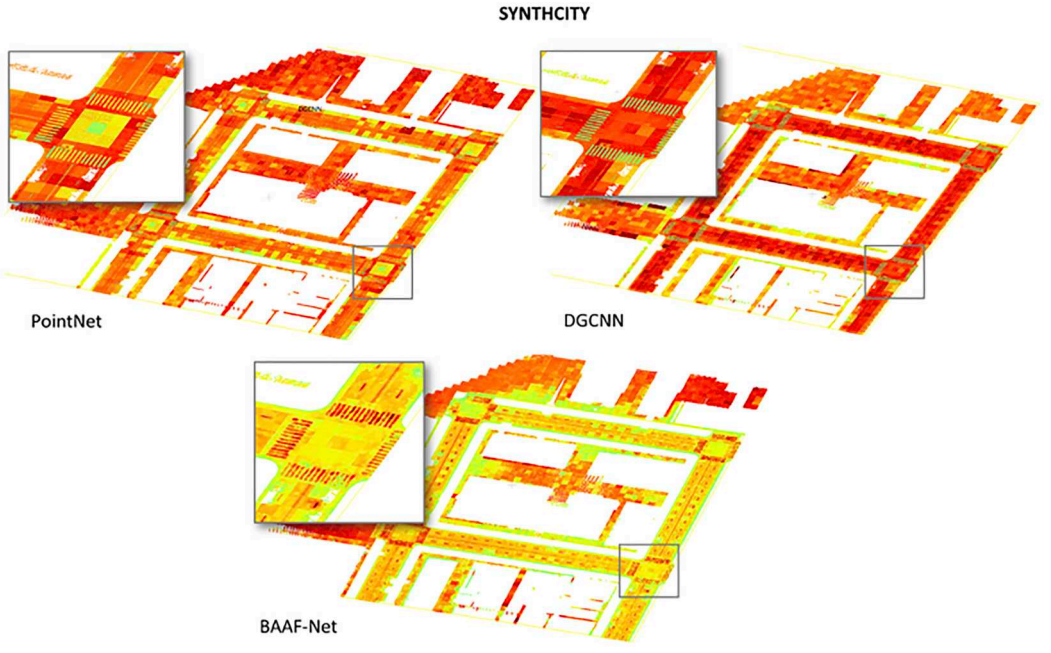

Figure 13. Comparison between the results obtained with the different DNNs within the same dataset.

18 and 19, it can be observed that for both lighting and traffic lights poles, PointNet and DGCNN pay more attention to the height dimension, starting from the ground level. At the same time, the hanging elements are ignored by the PointNet. This is due to the discontinuity of the point cloud in these particular elements. Conversely, DGCNN keeps attention on these elements.

This result indicates that the KNN edge-conv module of the network allows more robust aggregation within the neighbours of the aforementioned elements. BAAF-Net, on the contrary, pays more attention to the hanging elements, ignoring the poles. The main reason is the network's random sampling, which emphasizes elements with a higher density, ignoring the overall geometry.

5.2. Discussions

The comparison between classes has provided insights into network dependency and how different classes exhibit distinct behaviours. The performance of the networks varies depending on the class being analyzed, indicating that certain classes may be more challenging to classify accurately than others. This outcome highlights the need for careful consideration of network architectures and training strategies to achieve optimal results across different classes. In order to better evaluate the influence of the various features and to try to investigate the likely presence of common patterns, the Pearson correlation coefficient (r) was computed for each test scene of each dataset with respect to each analysed network architecture. In particular, the roughness and density have been computed with two radii (0.5 m and 1 m) on CloudCompare for each scene with the different classes as targets (e.g. dataset S3DIS-PointNet, class 0 – class 1 – class 2 …; S3DIS-DGCNN, class 0 – class 1 – class 2 …; S3DIS-BAAF-Net, class 0 – class 1 – class 2 and so on), and then correlated with the values obtained from the Explainable framework.

The measures of the two radii have been selected according to the dimension of the stride and half of it. Tests have also been carried out with lower radii, such as 0.1 and 0.2 m, but due to the structure of the point clouds, the obtained values were contained in a too narrow range, making all the values too similar. For this reason, larger radii were preferred.

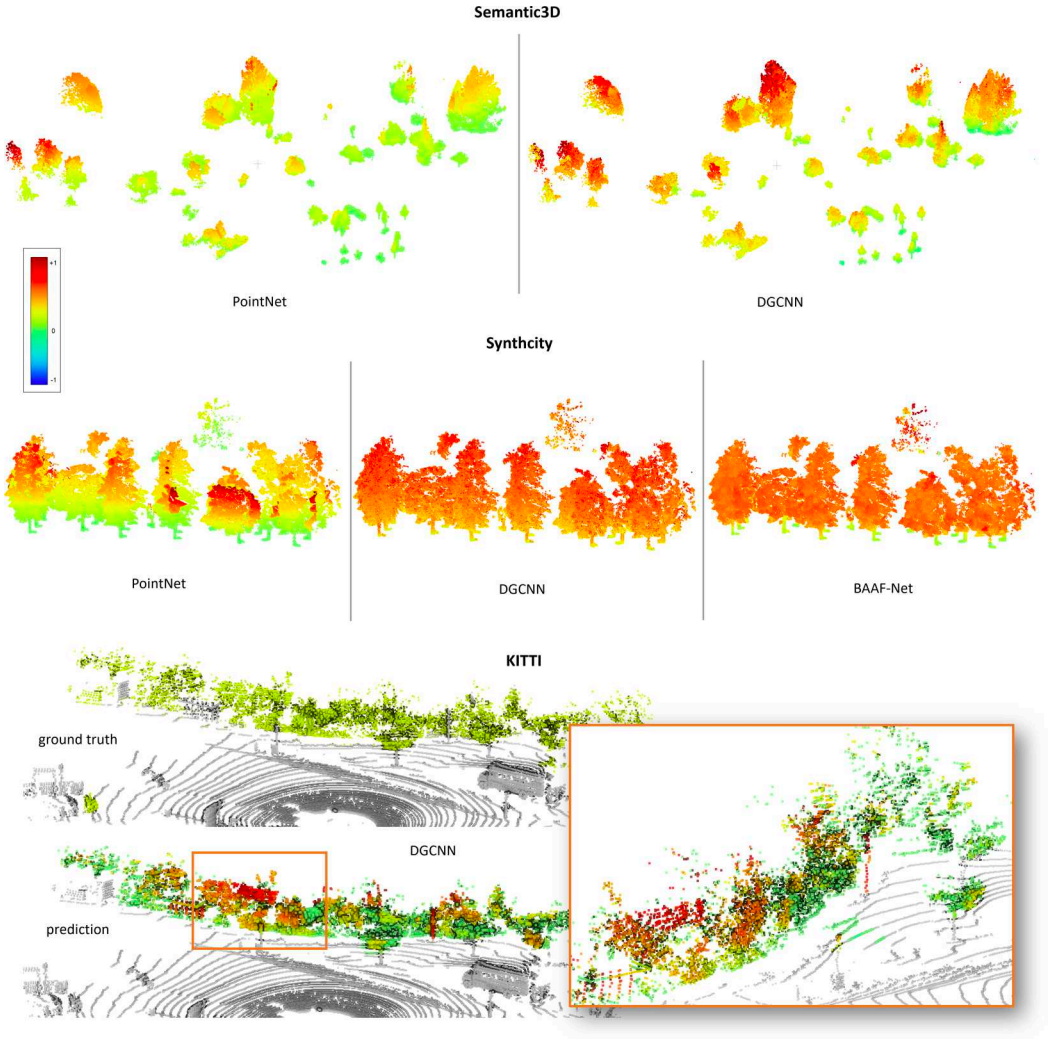

Figure 14. Results of the tree/vegetation categories and the analysed datasets.

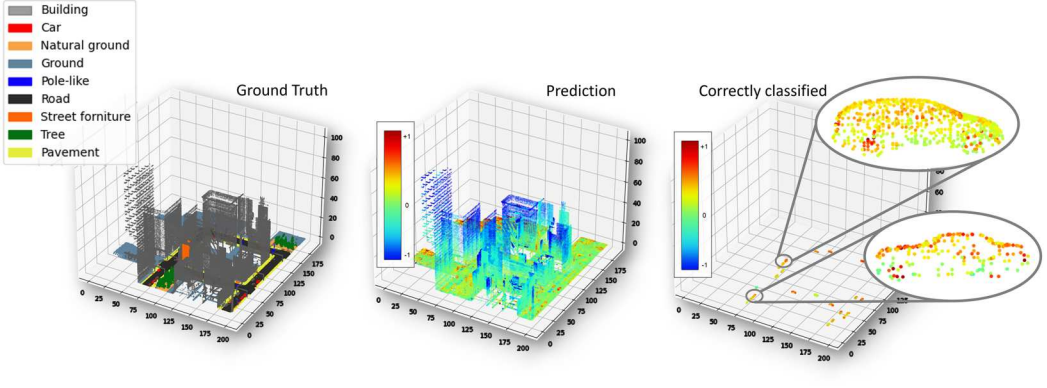

Figure 15. Highlighting of some of the correctly predicted cars in the Synthcity dataset analysed with the DGCNN. The car top and the wheels are the most relevant elements.

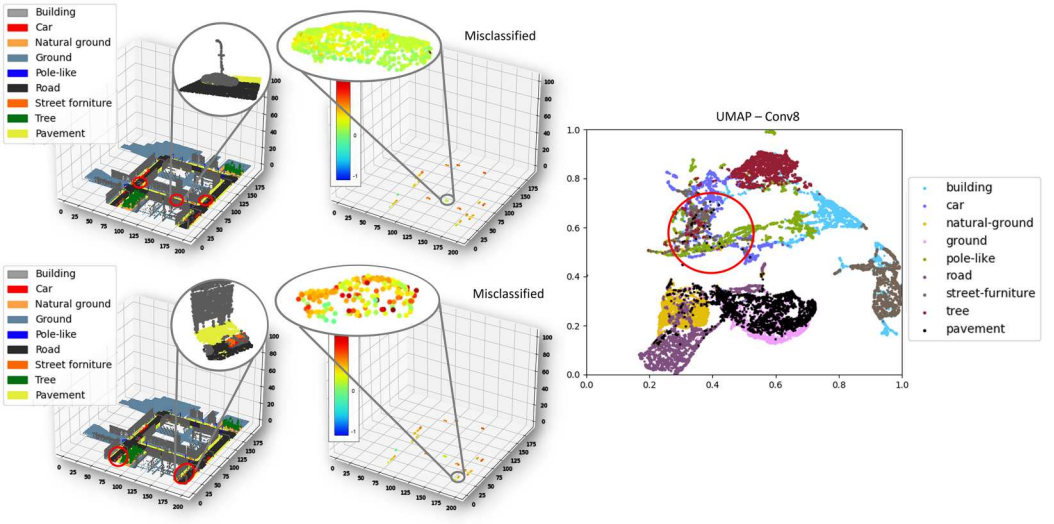

Figure 16. Highlighting of some of the misclassified cars in the Synthcity dataset and relative visualization in UMAP.

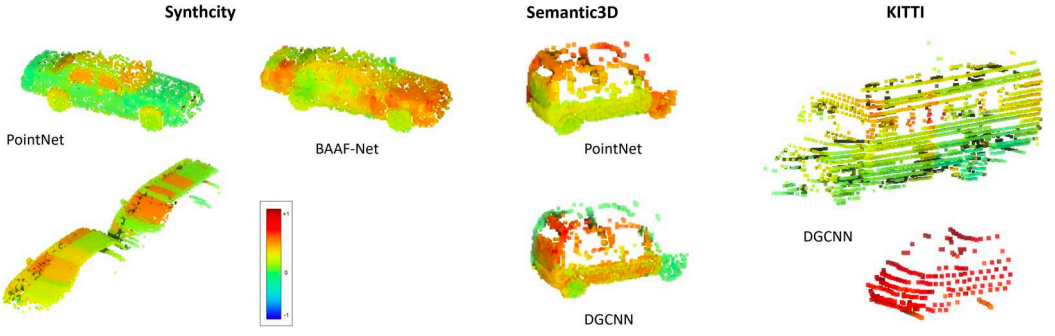

Figure 17. Comparison of the 'car' category among the various datasets.

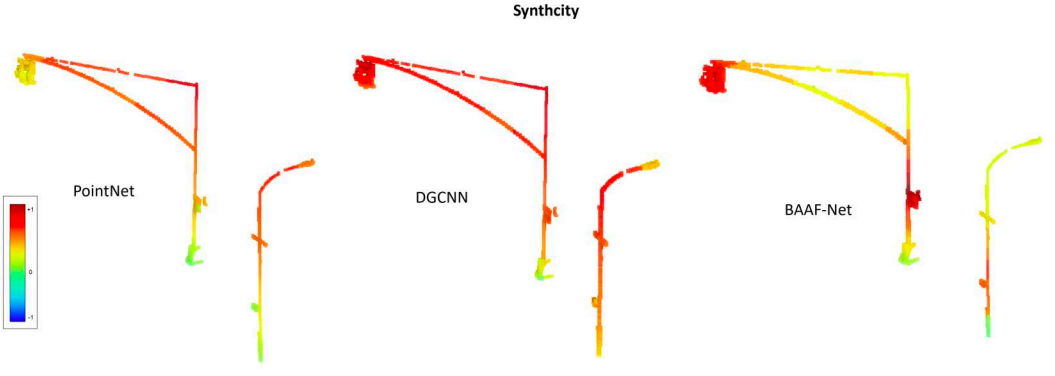

Figure 18. Example of poles and traffic lights for the Synthcity dataset.

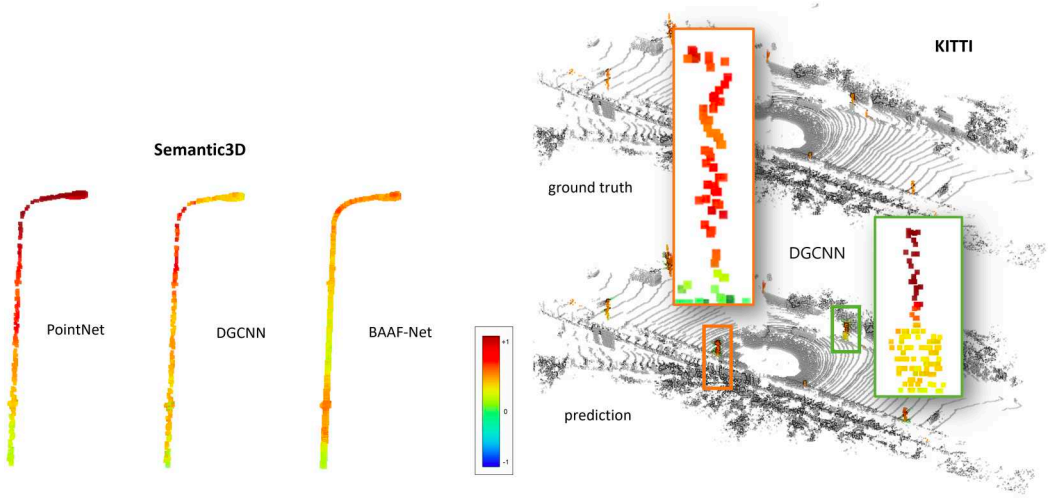

Figure 19. Example of poles in the Semantic3D and the KITTI dataset.

These tests have been divided into two groups: on one hand, the points taken into account for the r computation have been those of the whole scene, explored according to the target class; on the other hand, only the points belonging to the specific class were considered for the computation of the r coefficient, and not the whole scene.

Considering the first group of tests, an example of part of the results is reported in Tables A3, A4, A5, A6, A7, and A8 in the Appendix section). When comparing real and synthetic datasets, it becomes evident that the radiometric component and the noise level in point clouds play crucial roles in class discrimination. Synthetic datasets, with their absence of noise/roughness and well-defined radiometric descriptions, tend to yield more accurate and easily interpretable results. On the other hand, real datasets exhibit more variability, also in terms of point cloud density. However, the density factor is less relevant for most datasets since they undergo a further sub-sampling during the analysis of the DNNs themselves, except for the S3DIS dataset analysed with the DGCNN, where the point cloud density correlation is remarkably higher with respect to the other tests (see Table A4 in the Appendix. This result could be due to the initial higher density of the dataset, acquired in smaller indoor environments such as offices, combined with the DGCNN architecture that strictly relates a point with its surroundings. It is important to note that the density factor influences the network's ability to examine neighbours more comprehensively.

When comparing indoor and outdoor datasets, we observe similar behaviours among scenes regardless of the network architecture in indoor environments (e.g. S3DIS). However, in outdoor environments (e.g. Semantic3D), the results vary depending on the type of acquisition. The architectural design of the network plays a more significant role in outdoor scenarios, where the networks exhibit varying sensitivities to different scene characteristics. For example, the analysis of the floor class shows more pronounced differences in saliency patterns across networks in outdoor environments, while indoor environments exhibit more consistent patterns. The size of objects within the point cloud has a significant impact on the network's behaviour and saliency. For instance, trees exhibit distinctive characteristics that influence the network's attention and saliency patterns. The network's ability to recognize and differentiate objects depends on their size and geometric properties.

Considering the second group of tests, focused on the single classes (see Tables A9, A10, A11, A12, A13 in the Appendix section), it has been possible to highlight how *low correlation* results (in a range of ±0.3 values) are generally spread across the different classes and DNNs. This output further confirms how, as mentioned above, the density and the roughness are not effective features

in the learning process for most of the categories. However, some higher values (*medium correlation* – between ±0.31 and ±0.7) have been noticed for specific classes. An example is the relatively high correlation, if compared with the other classes, between the roughness and the *natural ground, ground and pavement classes* for the Synthcity dataset. As regards the vegetation of the Semantic3D dataset, the density seems to play a relevant role in the DGCNN, not confirmed with the BAAF-Net. Higher values, around 0.6, are reported for the *scanning artefacts* in the DGCNN, reflecting the

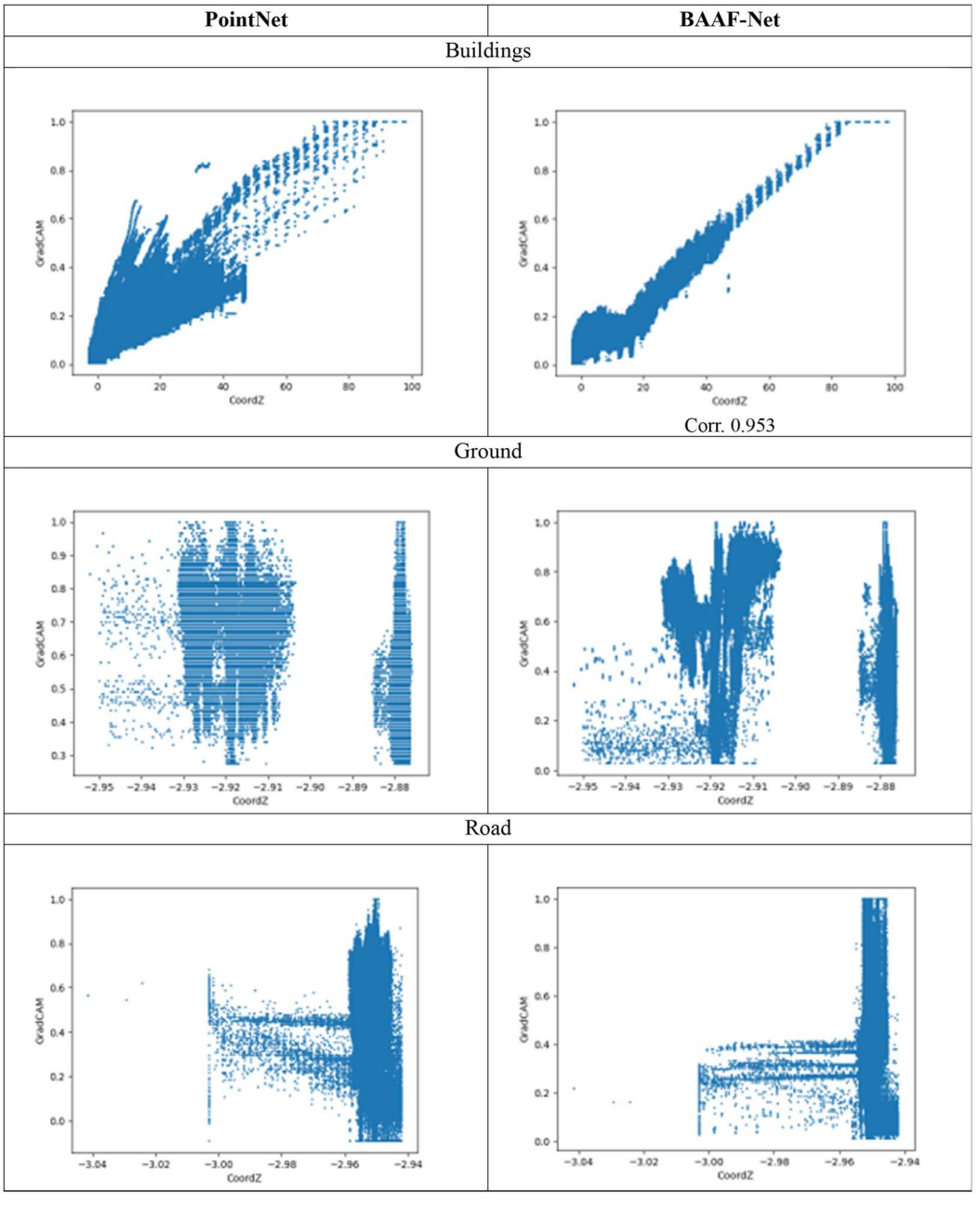

Figure 20. Examples of the scatter plots obtained for the Z-value across three different classes of the Synthcity dataset, with a comparison between PointNet and the BAAF-Net.

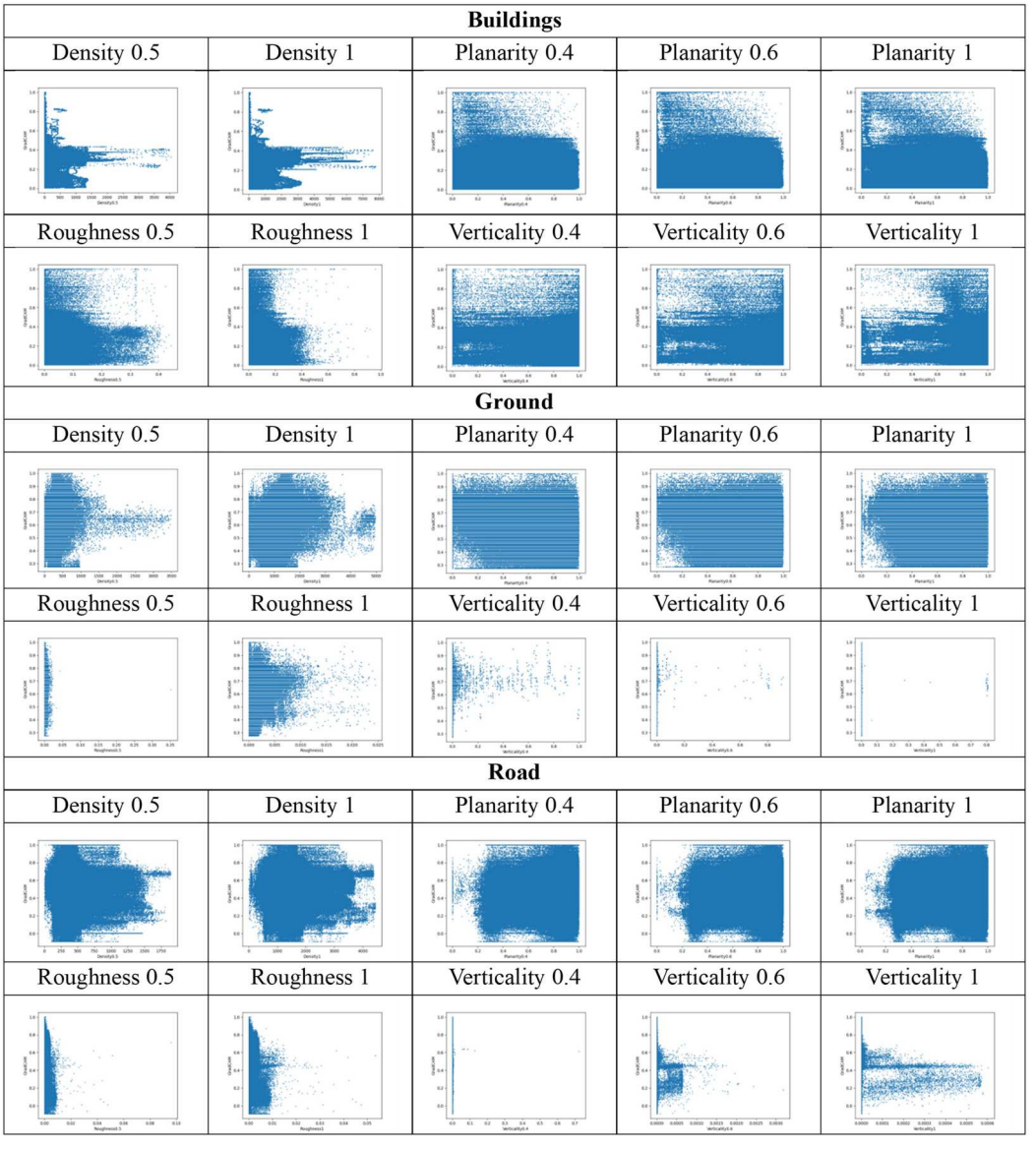

Figure 21. Example of the scatter plots from PointNet for the different features and radii.

behaviour of the vegetation with the BAAF-Net. Medium correlation, in this case, has also emerged for the roughness.

Since density and roughness have not led to additional insights, the *z-value* has been included as a further feature to be investigated, considering the visual outputs of the Explainable framework. Other 3D features, such as the verticality and planarity (computed with a radius of 0.4, 0.6 and 1 m), were also analysed, but all the coefficients resulted in a low correlation, so they have not been inserted in the overall tables. Scatter plots have also been generated to check the values visually. Figure 20 depicts an example of the scatter plots obtained for the Z-value across three different classes of the Synthcity dataset and with a comparison between PointNet and the BAAF-Net; while Figure 21 shows an example of the scatter plots obtained from PointNet for the different features and radii. Similar behaviour was obtained for the other architectures. From these figures, it is

Table 2. Comparative overview table with the relevance of the features according to the datasets. From low (*) to high (***).

Dataset/Feature	Overall	Typology of dataset		Scenes		Data acquisition sensor	
		Real	Synthetic	Indoor	Outdoor	Static	Mobile
Density	*	**	*	**	*	**	*
Roughness	**	**	**	*	**	*	**
Completeness	**	**	*	**	***	**	***
Radiometric component	**	**	***	*	**	**	**

Table 3. Comparative overview table with the relevance of the features according to the classes. From low (*) to high (***).

Class/Feature	Building	Floor/pavement	Tree/Vegetation	Car	Hard-scape
Density	*	*	**	*	**
Roughness	*	**	*	*	**
Completeness	*	*	**	***	**
Radiometric component	*	*	*	*	*
Z-component	***	*	**	*	**

possible to understand how the *z-value* is fundamental for the *building* class, and, at the same time, the other features do not have a relevant influence.

The results thus obtained were divided into 3 ranges (Table 2 and 3): ±0.3 values have been considered as a *low correlation*, between ±0.31 and ±0.7 a *medium correlation*, major of ±0.71 a *high correlation*. 'Completeness' and 'radiometric component' were instead assessed qualitatively and, therefore, moved to the end of the table.

Tables 2 and 3 provides a summary of the discussions and highlights that emerged from the conducted analyses.

Regarding the second research question, it is still challenging to define general guidelines at this stage. In fact, the results and salient points or elements are highly dependent on numerous factors, as discussed above. The interplay between network architecture, dataset characteristics (real or synthetic), noisiness, density variations, object sizes, and the distinction between static and mobile acquisition systems contribute to the complex behaviour observed in the results. Further research and analysis are thus necessary to establish more comprehensive guidelines and recommendations for effective point cloud analysis and interpretation.

6. Conclusions and future works

The burgeoning volume of 3D data necessitates the geomatics community's development of models that are more accurate, less uncertain, and physically consistent in interpreting this complex data. While neural networks for point cloud processing have garnered significant attention, there remains a lack of focus on explainability. The aim of this paper was to highlight the performance of existing frameworks by elucidating the outputs of deep learning 'black box' methods on the challenging task of semantic segmentation that, to the best of our knowledge, was never tried on 3D data. For this reason, in this study, we investigated the behaviour of DNNs in analyzing 3D point cloud data for semantic segmentation. By examining various datasets and network architectures, we aimed to understand how networks operate, learn, and analyze point clouds to classify different classes. Our analysis and discussions have provided valuable insights into the performance and characteristics of the networks. The comparison between different classes revealed that networks' behaviour could vary significantly depending on the class being considered. Some classes may exhibit more distinctive features that are easier for the networks to recognize, while others may pose more challenges in terms of classification accuracy. This consideration brings out the importance of carefully selecting network architectures and considering class-specific characteristics when designing and training networks for point cloud analysis tasks. Furthermore, the

comparison between real and synthetic datasets highlighted the impact of the radiometric component and noise level on class discrimination. Synthetic datasets, with their controlled and noise-free nature, often yield more accurate and interpretable results. On the other hand, real datasets exhibit more variability, and the noise level can influence the network's ability to accurately classify different classes. We also observed that the behaviour of networks could vary between indoor and outdoor environments. While there are similarities in behaviour within indoor environments, outdoor scenes exhibit more sensitivity to different scene characteristics, making the architectural design of the network more crucial in achieving accurate segmentation results. The size of objects within the point cloud was found to be an important factor influencing the network's traits. Larger objects, such as trees, have distinct features that influence the network's attention and saliency patterns.

This paper opens up several avenues for future research. First, further investigations can explore additional datasets and network architectures to gain a more comprehensive understanding of neural networks' behaviour in point cloud analysis. This can help identify trends, patterns, and general guidelines for network design and training in different scenarios. Generally speaking, in fact, the applicability of the proposed methodology is guaranteed provided that from the feature extraction layer, it is possible to reconstruct the geometry of the input point cloud, i.e. it is possible to visualize the intensity of the features or a combination of them, at each point of the cloud. If, for example, we consider the latest Transformer models, encoder-decoder architecture that redefined machine learning, even if particularly used in natural language processing and not in the geospatial data domain yet, the attention unit consists of 3 trained, fully connected neural network layers, so our method would be anyway applicable.

Additionally, the analysis of point cloud density and its impact on network performance warrants further exploration. Investigating the optimal sub-sampling strategies and their effects on classification accuracy and saliency patterns can provide valuable insights for improving point cloud analysis methods. Furthermore, the examination of mobile acquisition systems and their impact on network performance can offer insights into the robustness and generalizability of trained models in real-world scenarios. It would also be valuable to investigate the generalizability of network architectures across different datasets and domains. Comparing the behaviour of networks trained on specific datasets to their performance on unseen datasets can shed light on transfer learning capabilities and identify potential limitations. Therefore, we think that quantitative assessments, currently underdeveloped, will be increasingly crucial in the field of explainable deep learning, and they could cover and constitute one of the missions of the geomatics community.

Acknowledgments

The authors would like to thank: Ph.D. Andrea Felicetti for the experiments curation and data analysis support, and prof. Andrea Maria Lingua for methodological insights.

Disclosure statement

No potential conflict of interest was reported by the author(s).

Funding

This study was carried out within the FAIR – Future Artificial Intelligence Research and received funding from the European Union Next-GenerationEU (PIANO NAZIONALE DI RIPRESA E RESILIENZA (PNRR) – MISSIONE 4 COMPONENTE 2, INVESTIMENTO 1.3 – D.D. 1555 11/10/2022, PE00000013). This manuscript reflects only the authors' views and opinions, neither the European Union nor the European Commission can be considered responsible for them.

Data availability statement

The data that support the findings of this study are freely available online and published on the website of the respective datasets. Restrictions apply to the availability of these data, which were used under license for this study.

ORCID

Francesca Matrone http://orcid.org/0000-0002-9160-1674
Marina Paolanti http://orcid.org/0000-0002-5523-7174
Emanuele Frontoni http://orcid.org/0000-0002-8893-9244
Roberto Pierdicca http://orcid.org/0000-0002-9160-834X

References

Adebayo, Julius, Justin Gilmer, Michael Muelly, Ian Goodfellow, Moritz Hardt, and Been Kim. 2018. "Sanity Checks for Saliency Maps." In *32nd Conference on Neural Information Processing Systems (NeurIPS 2018), Montréal, Canada*, edited by S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, Vol. 31. Curran Associates, Inc.

Ahsan, Md Manjurul, Kishor Datta Gupta, Mohammad Maminur Islam, Sajib Sen, Md Lutfar Rahman, and Mohammad Shakhawat Hossain. 2020. "Study of Different Deep Learning Approach with Explainable AI for Screening Patients with COVID-19 Symptoms: Using CT Scan and Chest X-ray Image Dataset." Preprint arXiv:2007.12525.

Armeni, Iro, Ozan Sener, Amir R. Zamir, Helen Jiang, Ioannis Brilakis, Martin Fischer, and Silvio Savarese. 2016. "3D Semantic Parsing of Large-Scale Indoor Spaces." In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, June 27–30*, 1534–1543. IEEE. https://doi.org/10.1109/CVPR.2016.170.

Arnold, Nicholas I., Plamen Angelov, and Peter M. Atkinson. 2022. "An Improved Explainable Point Cloud Classifier (XPCC)." *IEEE Transactions on Artificial Intelligence* 4 (1): 71–80. https://doi.org/10.1109/TAI.2022.3150647.

Atik, Muhammed Enes, Zaide Duran, and Dursun Zafer Seker. 2024. "Explainable Artificial Intelligence for Machine Learning-Based Photogrammetric Point Cloud Classification." *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing* 17:5834–5846. https://doi.org/10.1109/JSTARS.2024.3370159.

Bach, Sebastian, Alexander Binder, Grégoire Montavon, Frederick Klauschen, Klaus-Robert Müller, and Wojciech Samek. 2015. "On Pixel-Wise Explanations for Non-Linear Classifier Decisions by Layer-Wise Relevance Propagation." *PloS One* 10 (7): e0130140. https://doi.org/10.1371/journal.pone.0130140.

Balado, Jesús, Elena González, Juan L. Rodríguez-Somoza, and Pedro Arias. 2023. "Multi Feature-Rich Synthetic Colour to Improve Human Visual Perception of Point Clouds." *ISPRS Journal of Photogrammetry and Remote Sensing* 196:514–527. https://doi.org/10.1016/j.isprsjprs.2023.01.019.

Behley, Jens, Martin Garbade, Andres Milioto, Jan Quenzel, Sven Behnke, Cyrill Stachniss, and Jurgen Gall. 2019. "SemanticKITTI: A Dataset for Semantic Scene Understanding of LiDAR Sequences." In *2019 IEEE/CVF International Conference on Computer Vision (ICCV), Seoul, Korea (South), October 27–November 2*, 9297–9307. IEEE. https://doi.org/10.48550/arXiv.1904.01416.

Bruna, Joan, Wojciech Zaremba, Arthur Szlam, and Yann LeCun. 2013. "Spectral Networks and Locally Connected Networks on Graphs." Preprint arXiv:1312.6203.

Burkart, Nadia, and Marco F. Huber. 2021. "A Survey on the Explainability of Supervised Machine Learning." *Journal of Artificial Intelligence Research* 70:245–317. https://doi.org/10.1613/jair.1.12228.

Cian, David, Jan van Gemert, and Attila Lengyel. 2020. "Evaluating the Performance of the LIME and Grad-CAM Explanation Methods on a LEGO Multi-Label Image Classification Task." Preprint arXiv:2008.01584.

Döllner, Jürgen. 2020. "Geospatial Artificial Intelligence: Potentials of Machine Learning for 3D Point Clouds and Geospatial Digital Twins." *PFG–Journal of Photogrammetry, Remote Sensing and Geoinformation Science* 88 (1): 15–24. https://doi.org/10.1007/s41064-020-00102-3.

Feng, Tuo, Ruijie Quan, Xiaohan Wang, Wenguan Wang, and Yi Yang. 2024. "Interpretable3D: An Ad-Hoc Interpretable Classifier for 3D Point Clouds." In *Thirty-Eighth AAAI Conference on Artificial Intelligence (AAAI-24), Vancouver, British Columbia, February 20–27*, 1761–1769. Washington, DC: AAAI Press.

Goodman, Bryce, and Seth Flaxman. 2016. "EU Regulations on Algorithmic Decision-Making and A 'Right to Explanation'." In *Proceedings of the 2016 ICML Workshop on Human Interpretability in Machine Learning (WHI 2016), New York, NY*. https://doi.org/10.48550/arXiv.1607.02531.

Griffiths, David, and Jan Boehm. 2019. "SynthCity: A Large Scale Synthetic Point Cloud." Preprint arXiv:1907.04758.

Gupta, Ananya, Simon Watson, and Hujun Yin. 2020. "3D Point Cloud Feature Explanations Using Gradient-Based Methods." In *2020 International Joint Conference on Neural Networks (IJCNN)*, 1–8. IEEE. https://arxiv.org/abs/2006.05548.

Hackel, Timo, Nikolay Savinov, Lubor Ladicky, Jan D. Wegner, Konrad Schindler, and Marc Pollefeys. 2017. "Semantic3d. Net: A New Large-Scale Point Cloud Classification Benchmark." Preprint arXiv:1704.03847.

Heuillet, Alexandre, Fabien Couthouis, and Natalia Díaz-Rodríguez. 2021. "Explainability in Deep Reinforcement Learning." *Knowledge-Based Systems* 214:106685. https://doi.org/10.1016/j.knosys.2020.106685.

Hsu, Chia-Yu, and Wenwen Li. 2023. "Explainable GeoAI: Can Saliency Maps Help Interpret Artificial Intelligence's Learning Process? An Empirical Study on Natural Feature Detection." *International Journal of Geographical Information Science* 37 (5): 963–987. https://doi.org/10.1080/13658816.2023.2191256.

Hu, Qingyong, Bo Yang, Linhai Xie, Stefano Rosa, Yulan Guo, Zhihua Wang, Niki Trigoni, and Andrew Markham. 2020. "Randla-Net: Efficient Semantic Segmentation of Large-Scale Point Clouds." In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, June 13–19*, 11108–11117. IEEE.

Ieracitano, Cosimo, Nadia Mammone, Amir Hussain, and Francesco Carlo Morabito. 2021. "A Novel Explainable Machine Learning Approach for EEG-based Brain-Computer Interface Systems." *Neural Computing and Applications* 34:11347–11360. https://doi.org/10.1007/s00521-020-05624-w.

Kim, Beomsu, Junghoon Seo, Seunghyeon Jeon, Jamyoung Koo, Jeongyeol Choe, and Taegyun Jeon. 2019. "Why Are Saliency Maps Noisy? Cause of and Solution to Noisy Saliency Maps." In *2019 IEEE/CVF International Conference on Computer Vision Workshop (ICCVW), Seoul, South Korea, October 27–28*, 4149–4157. IEEE.

Landrieu, Loic, and Martin Simonovsky. 2018. "Large-Scale Point Cloud Semantic Segmentation with Superpoint Graphs." In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, June 18–23*, 4558–4567. Institute of Electrical and Electronics Engineers (IEEE).

Levi, Meir Yossef, and Guy Gilboa. 2024. "Fast and Simple Explainability for Point Cloud Networks." Preprint arXiv:2403.07706.

Lin, Chen-Hsuan, Chen Kong, and Simon Lucey. 2018. "Learning Efficient Point Cloud Generation for Dense 3D Object Reconstruction." In *Proceedings of the AAAI Conference on Artificial Intelligence, New Orleans, Louisiana, USA, February 2–7*, Vol. 32. Association for the Advancement of Artificial Intelligence (AAAI).

Lundberg, Scott M., and Su-In Lee. 2017. "A Unified Approach to Interpreting Model Predictions." In *Advances in Neural Information Processing Systems, Long Beach, CA, USA, December 4-9*, edited by I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett Purchase, Vol. 30. Curran Associates, Inc.

Masci, Jonathan, Davide Boscaini, Michael Bronstein, and Pierre Vandergheynst. 2015. "Geodesic Convolutional Neural Networks on Riemannian Manifolds." In *Proceedings of the IEEE International Conference on Computer Vision Workshops, Santiago, Chile, December 7–13*, 37–45. NW Washington, DC: IEEE Computer Society.

Matrone, F., A. Felicetti, M. Paolanti, and R. Pierdicca. 2023. "Explaining AI: Understanding Deep Learning Models for Heritage Point Clouds." *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences* X-M-1-2023:207-214. https://doi.org/10.5194/isprs-annals-X-M-1-2023-207-2023.

Matrone, Francesca, Andrea Lingua, Roberto Pierdicca, Eva Savina Malinverni, Marina Paolanti, Eleonora Grilli, Farbio Remondino, Arnadi Murtiyoso, and Tania Landes. 2020. "A Benchmark for Large-Scale Heritage Point Cloud Semantic Segmentation." *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences* 43:1419–1426. https://doi.org/10.5194/isprs-archives-XLIII-B2-2020-1419-2020.

Matrone, Francesca, Marina Paolanti, Andrea Felicetti, Massimo Martini, and Roberto Pierdicca. 2022. "Bubblex: An Explainable Deep Learning Framework for Point-Cloud Classification." *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing* 15:6571–6587. https://doi.org/10.1109/JSTARS.2022.3195200.

Maturana, Daniel, and Sebastian Scherer. 2015. "Voxnet: A 3D Convolutional Neural Network for Real-Time Object Recognition." In *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Hamburg, Germany, September 28–October 3*, 922–928. IEEE.

Montavon, Grégoire. 2019. "Gradient-Based vs. Propagation-Based Explanations: An Axiomatic Comparison." In *Explainable AI: Interpreting, Explaining and Visualizing Deep Learning*, Lecture Notes in Computer Science (LNCS, volume 11700), edited by Wojciech Samek, Grégoire Montavon, Andrea Vedaldi, Lars Kai Hansen, and Klaus-Robert Mülle, 253–265. Springer. https://doi.org/10.1007/978-3-030-28954-6_13.

Nguyen, Anh, Jason Yosinski, and Jeff Clune. 2019. "Understanding Neural Networks Via Feature Visualization: A Survey." In *Explainable AI: Interpreting, Explaining and Visualizing Deep Learning*, Lecture Notes in Computer Science (LNCS, volume 11700), edited by Wojciech Samek, Grégoire Montavon, Andrea Vedaldi, Lars Kai Hansen, and Klaus-Robert Mülle, 55–76. Springer. https://doi.org/10.1007/978-3-030-28954-6_4.

Pierdicca, Roberto, Marina Paolanti, Francesca Matrone, Massimo Martini, Christian Morbidoni, Eva Savina Malinverni, Emanuele Frontoni, and Andrea Maria Lingua. 2020. "Point Cloud Semantic Segmentation Using a Deep Learning Framework for Cultural Heritage." *Remote Sensing* 12 (6): 1005. https://doi.org/10.3390/rs12061005.

Poppi, Samuele, Marcella Cornia, Lorenzo Baraldi, and Rita Cucchiara. 2021. "Revisiting the Evaluation of Class Activation Mapping for Explainability: A Novel Metric and Experimental Analysis." In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Nashville, TN, USA, June 20–25*, 2299–2304. IEEE Computer Society.

Qi, Charles R., Hao Su, Kaichun Mo, and Leonidas J. Guibas. 2017. "Pointnet: Deep Learning on Point Sets for 3D Classification and Segmentation." In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, July 21–26*, 652–660. IEEE Computer Society.

Qi, Charles R., Hao Su, Matthias Nießner, Angela Dai, Mengyuan Yan, and Leonidas J. Guibas. 2016. "Volumetric and Multi-View CNNs for Object Classification on 3D Data." In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, June 27–30*, 5648–5656. IEEE Computer Society.

Qi, Charles Ruizhongtai, Li Yi, Hao Su, and Leonidas J. Guibas. 2017. "Pointnet++: Deep Hierarchical Feature Learning on Point Sets in A Metric Space." In *Advances in Neural Information Processing Systems*, *Long Beach, CA, USA, December 4–9*, edited by I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, R. Garnett, Vol. 30. Curran Associates Inc.

Qiu, Shi, Saeed Anwar, and Nick Barnes. 2021. "Semantic Segmentation for Real Point Cloud Scenes via Bilateral Augmentation and Adaptive Fusion." In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Virtual Conference, June 19–25*, 1757–1767. Institute of Electrical and Electronics Engineers (IEEE).

Ren, Jiawei, Liang Pan, and Ziwei Liu. 2022. "Benchmarking and Analyzing Point Cloud Classification Under Corruptions." In *39th International Conference on Machine Learning (ICML 2022), Baltimore, Maryland, USA, July 17-23*, edited by Kamalika Chaudhuri, Stefanie Jegelka, Le Song, Csaba Szepesvari, Gang Niu, and Sivan Sabato, 18559–18575. PMLR.

Ribeiro, Marco Tulio, Sameer Singh, and Carlos Guestrin. 2016. "'Why Should I Trust You?' Explaining the Predictions of Any Classifier." In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Francisco, California, USA, August 13–17*, 1135–1144. New York, NY: Association for Computing Machinery.

Samek, Wojciech, Alexander Binder, Grégoire Montavon, Sebastian Lapuschkin, and Klaus-Robert Müller. 2016. "Evaluating the Visualization of what a Deep Neural Network Has Learned." *IEEE Transactions on Neural Networks and Learning Systems* 28 (11): 2660–2673. https://doi.org/10.1109/TNNLS.5962385.

Selvaraju, Ramprasaath R., Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. 2017. "Grad-Cam: Visual Explanations From Deep Networks Via Gradient-Based Localization." In *Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, October 22–29*, 618–626. IEEE Computer Society.

Shrikumar, Avanti, Peyton Greenside, and Anshul Kundaje. 2017. "Learning Important Features Through Propagating Activation Differences." In *International Conference on Machine Learning, International Convention Centre, Sydney, Australia, August 6–11*, 3145–3153. PMLR.

Simonyan, Karen, Andrea Vedaldi, and Andrew Zisserman. 2013. "Deep Inside Convolutional Networks: Visualising Image Classification Models and Saliency Maps." Preprint arXiv:1312.6034.

Smilkov, Daniel, Nikhil Thorat, Been Kim, Fernanda Viégas, and Martin Wattenberg. 2017. "Smoothgrad: Removing Noise by Adding Noise." Preprint arXiv:1706.03825.

Springenberg, Jost Tobias, Alexey Dosovitskiy, Thomas Brox, and Martin Riedmiller. 2014. "Striving for Simplicity: The All Convolutional Net." Preprint arXiv:1412.6806.

Sundararajan, Mukund, Ankur Taly, and Qiqi Yan. 2016. "Gradients of Counterfactuals." Preprint arXiv:1611.02639.

Sundararajan, Mukund, Ankur Taly, and Qiqi Yan. 2017. "Axiomatic Attribution for Deep Networks." In *International Conference on Machine Learning, International Convention Centre, Sydney, Australia, August 6–11*, 3319–3328. PMLR.

Tan, Hanxiao, and Helena Kotthaus. 2022. "Surrogate Model-Based Explainability Methods for Point Cloud NNS." In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, Waikoloa, HI, USA, January 3–8*, 2239–2248. IEEE Computer Society.

Uy, Mikaela Angelina, Quang-Hieu Pham, Binh-Son Hua, Thanh Nguyen, and Sai-Kit Yeung. 2019. "Revisiting Point Cloud Classification: A New Benchmark Dataset and Classification Model on Real-World Data." In *Proceedings of the IEEE/CVF International Conference on Computer Vision, Seoul, Korea (South), October 27–November 2*, 1588–1597. IEEE Computer Society.

Verburg, F. M. 2022. "Exploring Explainability and Robustness of Point Cloud Segmentation Deep Learning Model by Visualization." B.S. thesis, University of Twente.

Wang, Yue, Yongbin Sun, Ziwei Liu, Sanjay E. Sarma, Michael M. Bronstein, and Justin M. Solomon. 2019. "Dynamic Graph Cnn for Learning on Point Clouds." *ACM Transactions on Graphics (tog)* 38 (5): 1–12. https://doi.org/10.1145/3326362.

with code, Papers. 2023. "Semantic Segmentation on Semantic3D." Accessed November 20, 2022. https://paperswithcode.com/sota/semantic-segmentation-on-semantic3d.

with code, Papers. 2023a. "Semantic Segmentation on S3DIS." Accessed November 20, 2022. https://paperswithcode.com/sota/semantic-segmentation-on-s3dis.

Wu, Zhirong, Shuran Song, Aditya Khosla, Fisher Yu, Linguang Zhang, Xiaoou Tang, and Jianxiong Xiao. 2015. "3D Shapenets: A Deep Representation for Volumetric Shapes." In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, June 7–June 12*, 1912–1920. IEEE Computer Society.

Xu, Mutian, Junhao Zhang, Zhipeng Zhou, Mingye Xu, Xiaojuan Qi, and Yu Qiao. 2021. "Learning Geometry-Disentangled Representation for Complementary Understanding of 3D Object Point Cloud." In *Proceedings of*

the AAAI Conference on Artificial Intelligence, February 2–9, Vol. 35, 3056–3064. Association for the Advancement of Artificial Intelligence (AAAI).

Young, Kyle, Gareth Booth, Becks Simpson, Reuben Dutton, and Sally Shrapnel. 2019. "Deep Neural Network or Dermatologist?" In *Interpretability of Machine Intelligence in Medical Image Computing and Multimodal Learning for Clinical Decision Support: Second International Workshop, iMIMIC 2019, and 9th International Workshop, ML-CDS 2019, Held in Conjunction with MICCAI 2019*, 48–55. Shenzhen, China: Springer.

Zhang, Jiaying, Xiaoli Zhao, Zheng Chen, and Zhejun Lu. 2019. "A Review of Deep Learning-Based Semantic Segmentation for Point Cloud." *IEEE Access* 7:179118–179133. https://doi.org/10.1109/Access.6287639.

Zhang, Min, Haoxuan You, Pranav Kadam, Shan Liu, and C.-C. Jay Kuo. 2020. "Pointhop: An Explainable Machine Learning Method for Point Cloud Classification." *IEEE Transactions on Multimedia* 22 (7): 1744–1755. https://doi.org/10.1109/TMM.6046.

Zheng, Tianhang, Changyou Chen, Junsong Yuan, Bo Li, and Kui Ren. 2019. "Pointcloud Saliency Maps." In *Proceedings of the IEEE/CVF International Conference on Computer Vision, Seoul, Korea (South), October 27-November 2*, 1598–1606. IEEE Computer Society.

Zhou, Yunxiang, Ankang Ji, Limao Zhang, and Xiaolong Xue. 2023. "Sampling-Attention Deep Learning Network with Transfer Learning for Large-Scale Urban Point Cloud Semantic Segmentation." *Engineering Applications of Artificial Intelligence* 117:105554. https://doi.org/10.1016/j.engappai.2022.105554.

Appendix

In this section, have been inserted: the results of the performances of the tests with the different datasets (Tables A1 and A2), and some examples of the *r* coefficient obtained both for the whole scene analyses and the single classes. In particular, Tables A3, A4, and A5 show an example of the obtained *r* coefficients for the S3DIS dataset. Tables A6, A7, and A8 show an example of the obtained *r* coefficients for the Synthcity, Semantic3D and KITTI datasets analysed according to the deep neural network that ensured the best performances. In this case, the points taken into account have been those of the whole scene, explored according to the target class. Tables A9, A10, A11, A12, and A13 provide an example of the *r* coefficients for the Synthcity and Semantic3D datasets, computed according to the points belonging only to the analysed classes. The features have been computed with different radii (0.5 m and 1 m).

Table A1. Results with *no overlap* setting.

Class	DGCNN			PointNet			BAAF-NET			Support
	Prec	Rec	F1	Prec	Rec	F1	Prec	Rec	F1	
Synthcity										
building	0.987	0.974	0.980	0.974	0.909	0.940	0.888	0.979	0.932	1700481
car	0.900	0.804	0.850	0.936	0.654	0.770	0.883	0.627	0.733	120936
naturalGround	0.947	0.952	0.949	0.919	0.954	0.937	0.933	0.964	0.949	250431
ground	0.947	0.940	0.944	0.745	0.919	0.823	0.944	0.831	0.884	2574331
poleLike	0.980	0.873	0.924	0.912	0.884	0.898	0.553	0.010	0.019	72102
road	0.995	0.950	0.972	0.988	0.897	0.940	0.973	0.991	0.982	2933319
streetFurniture	0.743	0.847	0.791	0.401	0.730	0.518	0.773	0.299	0.431	123848
tree	0.988	0.972	0.980	0.994	0.957	0.975	0.917	0.967	0.941	314339
pavement	0.833	0.925	0.877	0.673	0.560	0.611	0.776	0.887	0.828	1642309
accuracy			0.944			0.846			0.907	9732096
S3DIS										
beam	0.000	0.000	0.000	0.000	0.008	0.001				3306
board+bookcase	0.585	0.491	0.534	0.601	0.268	0.371				951950
ceiling	0.877	0.960	0.917	0.786	0.936	0.854				2186782
chair	0.663	0.467	0.548	0.418	0.388	0.402				209680
clutter	0.400	0.320	0.356	0.293	0.213	0.247				866772
column	0.074	0.009	0.015	0.000	0.000	0.000				158603
door+window	0.550	0.221	0.316	0.421	0.215	0.285				738748
floor	0.971	0.985	0.978	0.854	0.993	0.918				910090
sofa	0.263	0.071	0.112	0.000	0.000	0.000				26187
stairs	0.000	0.000	0.000	0.000	0.000	0.000				0
table	0.624	0.621	0.622	0.633	0.306	0.413				343564
wall	0.686	0.869	0.767	0.582	0.723	0.645				2762398

(*Continued*)

Table A1. Continued.

Class	DGCNN			PointNet			BAAF-NET			Support
	Prec	Rec	F1	Prec	Rec	F1	Prec	Rec	F1	
accuracy			0.749			0.662				10158080
Semantic3D										
manMadeTerrain	0.699	0.947	0.804	0.654	0.862	0.744				820011
naturalTerrain	0.746	0.581	0.653	0.620	0.523	0.568				178791
highVegetation	0.722	0.687	0.704	0.510	0.341	0.409				816849
lowVegetation	0.586	0.266	0.366	0.326	0.100	0.153				162239
buildings	0.814	0.866	0.839	0.678	0.816	0.741				1679174
hardScape	0.372	0.261	0.307	0.258	0.166	0.202				290925
scanningArtefacts	0.337	0.035	0.064	0.036	0.003	0.005				58124
cars	0.755	0.004	0.008	0.623	0.372	0.466				89887
accuracy			0.736			0.622				4096000

Table A2. Results with *overlap* setting.

Class	DGCNN			PointNet			BAAF-NET			Support
	Prec	Rec	F1	Prec	Rec	F1	Prec	Rec	F1	
S3DIS										
beam	0.002	0.015	0.004	0.000	0.011	0.001	0.020	0.219	0.037	10586
board+bookcase	0.737	0.405	0.522	0.607	0.242	0.347	0.738	0.546	0.627	3865735
ceiling	0.907	0.955	0.931	0.818	0.919	0.865	0.913	0.935	0.924	8801531
chair	0.712	0.529	0.607	0.336	0.326	0.331	0.693	0.752	0.721	805135
clutter	0.466	0.348	0.398	0.235	0.232	0.234	0.515	0.480	0.497	3404530
column	0.109	0.019	0.033	0.009	0.001	0.001	0.157	0.012	0.023	675757
door+window	0.503	0.444	0.472	0.471	0.171	0.251	0.658	0.257	0.370	3140483
floor	0.983	0.976	0.979	0.892	0.992	0.939	0.956	0.979	0.967	7661597
sofa	0.399	0.230	0.292	0.001	0.000	0.000	0.529	0.246	0.336	111357
stairs	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0
table	0.718	0.616	0.663	0.617	0.425	0.503	0.753	0.699	0.725	1395706
wall	0.697	0.914	0.791	0.587	0.740	0.654	0.714	0.927	0.806	11218655
accuracy			0.769			0.660			0.786	41091072
Semantic3D										
manMadeTerrain	0.858	0.929	0.892	0.803	0.888	0.843	0.888	0.900	0.894	5555801
naturalTerrain	0.582	0.811	0.678	0.529	0.733	0.615	0.565	0.747	0.644	1159238
highVegetation	0.820	0.724	0.769	0.636	0.311	0.418	0.762	0.458	0.572	4699083
lowVegetation	0.727	0.314	0.439	0.314	0.221	0.260	0.372	0.375	0.374	1051331
buildings	0.830	0.900	0.864	0.690	0.841	0.758	0.718	0.912	0.803	9343245
hardScape	0.452	0.403	0.426	0.283	0.309	0.296	0.379	0.191	0.254	1573484
scanningArtefacts	0.583	0.179	0.274	0.347	0.154	0.214	0.349	0.274	0.307	407001
cars	0.467	0.377	0.418	0.789	0.349	0.484	0.734	0.402	0.520	524673
accuracy			0.787			0.661			0.722	24313856
KITTI										
building	0.732	0.668	0.698							147945
car	0.566	0.121	0.199							16600
truck	0.789	0.227	0.353							10726
road	0.789	0.973	0.871							832988
parking	0.346	0.032	0.059							12519
sidewalk	0.612	0.263	0.368							144670
otherGround	0.718	0.150	0.248							57880
otherBuilding	0.766	0.429	0.550							74967
fence	0.823	0.811	0.817							253472
vegetation	0.815	0.868	0.841							1052648
trunk	0.000	0.000	0.000							14010
terrain	0.613	0.517	0.561							268420
pool	0.663	0.321	0.432							21463
trafficSign	0.711	0.118	0.203							4269
otherObject	0.613	0.387	0.474							42193
movingCar	0.498	0.772	0.605							58955
accuracy			0.770							3013725

Table A3. Example of the *r* coefficients for the S3DIS dataset obtained with PointNet.

S3DIS	roughness 0.5	density 0.5	roughness 1	density 1
beam	−0.123	0.32	−0.229	0.099
board+bookcase	−0.155	0.141	−0.237	0.052
ceiling	−0.066	0.404	−0.155	0.196
chair	−0.028	0.025	0.052	0.312
clutter	−0.178	0.034	−0.263	−0.095
column	−0.136	0.246	−0.272	−0.042
door+window	−0.084	−0.044	−0.233	−0.15
floor	0.1	0.036	0.223	0.091
sofa	0.092	−0.111	0.276	0.265
stairs	0.115	0.002	0.196	0.165
table	−0.059	0.001	0.087	0.303
wall	−0.059	0.001	0.087	0.303

Table A4. Example of the *r* coefficients for the S3DIS dataset obtained with the DGCNN.

S3DIS	roughness 0.5	density 0.5	roughness 1	density 1
beam	−0.07	0.394	−0.153	0.194
board+bookcase	−0.022	0.347	−0.053	−0.328
ceiling	−0.117	0.454	−0.215	0.277
chair	−0.045	−0.028	0.25	0.382
clutter	−0.02	−0.386	−0.103	−0.436
column	0.041	−0.35	−0.058	−0.468
door+window	0.062	−0.487	−0.039	−0.535
floor	−0.023	0.09	0.245	0.315
sofa	0.101	−0.021	0.279	0.287
stairs	−0.006	0.113	0.152	0.354
table	0.294	−0.099	0.303	0.239
wall	0.033	−0.357	−0.063	−0.493

Table A5. Example of the *r* coefficients for the S3DIS dataset obtained with the BAAF-Net.

S3DIS	roughness 0.5	density 0.5	roughness 1	density 1
beam	−0.124	0.225	−0.213	0.02
board+bookcase	−0.013	−0.36	−0.0125	−0.39
ceiling	−0.166	0.346	−0.29	0.133
chair	0.084	0.07	0.252	0.4
clutter	−0.294	0.108	−0.42	−0.03
column	0.07	−0.31	0.067	−0.35
door+window	−0.042	−0.366	−0.06	−0.405
floor	−0.026	0.125	0.192	0.349
sofa	0.119	−0.04	0.151	0.138
stairs	−0.044	0.196	−0.004	0.242
table	0.076	0.016	0.293	0.222
wall	−0.088	−0.036	−0.245	0.29

Table A6. Example of the *r* coefficients for the Synthcity dataset obtained with the DGCNN.

Synthcity	roughness 0.5	density 0.5	roughness 1	density 1
building	0.263	−0.217	0.185	−0.422
car	0.226	−0.135	0.184	−0.273
naturalGround	−0.139	0.008	−0.064	0.021
ground	−0.293	0.127	−0.286	0.227
polelike	0.344	−0.044	0.22	−0.156
road	−0.226	0.062	−0.207	0.24
street furniture	0.377	−0.161	0.27	−0.289
pavement	0.29	−0.198	0.262	−0.334

Table A7. Example of the *r* coefficients for the Semantic3D dataset obtained with the DGCNN.

Semantic3D	roughness 0.5*	density 0.5*
manMadeTerrain	−0.355	0.003
naturalTerrain	−0.167	0.012
highVegetation	0.321	−0.298
lowVegetation	0.249	0.016
buildings	0.084	−0.206
hardScape	−0.041	0.271
scanningArtefacts	−0.132	0.059
cars	−0.066	0.292

Note: *Only 0.5 m has been analysed since with 1 m CloudCompare repeatedly aborted the computation.

Table A8. Example of the *r* coefficients for the KITTI dataset obtained with the DGCNN.

KITTI	roughness 0.5	density 0.5	roughness 1	density 1
Buildings	0.044	−0.171	0.234	−0.172
Cars (parked)	0.146	0.002	0.139	−0.001
Truck/Bus	0.152	−0.141	0.196	−0.116
People	0.077	0.007	0.098	0.053
Road	−0.393	0.04	−0.445	−0.007
Parking	−0.315	0.058	−0.351	−0.022
Sidewalk	−0.383	−0.052	0.438	−0.087
Other ground	−0.363	0.134	−0.404	0.028
Other buildings	0.226	−0.05	0.289	0.001
Other buildings	0.243	−0.046	0.222	0.031
Vegetation	0.385	−0.179	0.47	−0.094
Trunk	0.334	−0.148	0.413	−0.065
Terrain	−0.373	−0.003	−0.445	−0.062
Poles	0.13	−0.136	0.141	−0.139
Traffic sign	0.18	−0.042	0.249	−0.027
Other street objects	0.153	0.002	0.129	0.001
Cars (moving)	0.057	0.132	0.026	0.106

Table A9. Example of the *r* coefficients for the Synthcity dataset obtained with PointNet.

Synthcity	roughness 0.5	density 0.5	roughness 1	density 1
building	0.051	0.243	−0.01	0.238
car	−0.01	−0.078	0.023	0.009
naturalGround	−0.365	0.088	−0.372	0.192
ground	−0.029	0.175	0.027	0.191
polelike	−0.072	−0.195	−0.064	−0.145
road	−0.101	−0.011	−0.101	0.077
street furniture	−0.166	0.07	−0.142	0.165
pavement	0.06	−0.11	0.117	0.001

Note: Only the points belonging to the analysed classes have been selected.

Table A10. Example of the *r* coefficients for the Synthcity dataset obtained with the DGCNN.

Synthcity	roughness 0.5	density 0.5	roughness 1	density 1
building	−0.028	0.007	−0.086	0.021
car	0.079	−0.229	0.122	−0.189
naturalGround	−0.434	−0.223	−0.439	−0.118
ground	0.041	0.02	0.049	0.131
polelike	−0.059	−0.188	0.019	−0.184
road	−0.164	−0.078	−0.162	0.033
street furniture	−0.212	0.149	−0.22	0.241
pavement	0.08	−0.373	0.114	−0.216

Note: Only the points belonging to the analysed classes have been selected.

Table A11. Example of the *r* coefficients for the Synthcity dataset obtained with the BAAF-Net.

Synthcity	roughness 0.5	density 0.5	roughness 1	density 1
building	0.049	0.167	0.008	0.207
car	0.147	−0.202	0.16	−0.179
naturalGround	−0.241	−0.019	−0.257	0.041
ground	0.27	0.184	0.344	0.155
polelike	−0.002	−0.155	0.028	−0.127
road	−0.03	0.079	−0.034	0.22
pavement	0.079	−0.396	0.117	−0.249

Note: Only the points belonging to the analysed classes have been selected.

Table A12. Example of the *r* coefficients for the Semantic3D dataset obtained with the DGCNN.

Semantic3D	roughness 0.5	density 0.5	roughness 1	density 1
manMadeTerrain	−0.107	0.091	−0.114	0.198
naturalTerrain	−0.06	−0.065	−0.116	0.099
highVegetation	−0.044	−0.452	−0.079	−0.481
lowVegetation	0.121	−0.098	0.128	0.064
buildings	−0.005	−0.206	−0.058	−0.231
hardScape	−0.046	0.136	−0.057	0.178
scanningArtefacts	−0.388	−0.594	−0.374	−0.672
cars	0.15	−0.054	0.236	0.066

Note: Only the points belonging to the analysed classes have been selected.

Table A13. Example of the *r* coefficients for the Semantic3D dataset obtained with the BAAF-Net.

Semantic3D	roughness 0.5	density 0.5	roughness 1	density 1
manMadeTerrain	−0.034	0.156	−0.064	0.235
naturalTerrain	−0.138	0.09	−0.221	0.207
highVegetation	0.008	−0.241	−0.0238	−0.215
lowVegetation	0.107	−0.065	0.124	−0.014
buildings	−0.022	−0.103	−0.085	−0.142
hardScape	−0.05	−0.216	−0.083	−0.251
scanningArtefacts	−0.379	−0.165	−0.364	−0.302
cars	0.182	−0.078	0.264	0.043

Note: Only the points belonging to the analysed classes have been selected.