Optimisation of PointNet++ for Tree Species Classification from Drone LiDAR Data

Nada Hamdani[1,4], Imane Abouhat[2], Kenza Ait El Kadi[1,2], Saloua Bensiali[1,3] & Imane Sebari[1,2]

[1]Research Unit of Geospatial Technologies for a Smart Decision, Hassan II Institute of Agronomy and Veterinary Medicine. 10101 Rabat, Morocco - n.hamdani@iav.ac.ma; abouhat.imane@gmail.com; k.aitelkadi@iav.ac.ma; i.sebari@iav.ac.ma
[2]Department of Photogrammetry and Cartography, School of Geomatics and Surveying Engineering. Hassan II Institute of Agronomy and Veterinary Medicine. 10101 Rabat, Morocco
[3]Department of Applied Statistics and Computer Science - s.bensiali@iav.ac.ma
[4]Société Topographie Informatique, 91000 Evry Courcouronnes, France - nada.hamdani@sti-topographie.fr

Keywords: Unmanned Aerial System (UAS), LiDAR, Deep Learning, PointNet++, Tree Species Classification, FOR-species20K.

Abstract

Trees play a key role in our planet. They regulate climate, preserve biodiversity, and contribute to human well-being. Each species has different contributions to our globe and a specific carbon storage potential. Identifying tree species enables better measurement of global carbon and helps authorities better manage forests and green spaces. Unmanned Aerial System (UAS) LiDAR has become a powerful source of 3D point cloud for vegetation analysis, given its ability to capture a large area in a short time and its capacity to penetrate canopy layers. Deep learning methods extract discriminative features directly from raw point clouds and generalize well to unseen datasets. This study optimises PointNet++ deep learning architecture for tree species classification by analysing the influence of sampling configurations on the performance of model detection, by using an open-source dataset “FOR-species20K”. Three-point cloud sampling configurations (4 096, 8 192, and 16 384 points per tree) were tested with three random seeds (0,42 and 123) to assess their impact on classification accuracy and ensure stability of prediction. Results on a separate test set of 508 trees show a consistent improvement in performance of PointNet++ with a sampling configuration of 8 192 points per tree, reaching a macro-average F1-score of 89.65%, surpassing the 74.9 % reported by (Puliti et al., 2025a) for evaluating the same architecture. Dominant species such as *Fagus sylvatica*, *Picea abies*, and *Pinus sylvestris* achieve F1-scores exceeding 90%, indicating high model robustness. This study approves that the performance of PointNet++ could be improved by raising the number of points from 4 096 to 8 192, but further increasing to 16 384 points introduces interspecific confusion and requires extensive computational time for model training. This research aligns and complements the global initiative led by Federal Institute of Technology Zurich in Switzerland (ETH Zurich), which is interested in identifying tree species using deep learning (“ETH Zurich”, 2025).

1. Introduction

Trees have several benefits for the environment. They improve air quality, provide oxygen, protect soil, regulate humidity, reduce heat islands, and provide citizens with a pleasant living space (Haase & Dushkova, 2024). However, not all species provide the same ecological and social benefits (Kongwen and Baoxin, 2012; Locosselli and Buckeridge, 2023).

The UAS LiDAR point clouds capture the 3D structure of vegetation, including its height, density, and canopy form. These characteristics are crucial for identifying species, going beyond what 2D images offer. Although they are pertinent, several challenges remain: variability between species, heterogeneity of point clouds and difficulty in adapting deep learning models to large datasets. Recent studies have shown that deep learning approaches, particularly point-based architectures, surpass traditional methods for tree species classification (Taher et al., 2026). To progress in this field, open databases play a key role by providing necessary data to train, test, and evaluate models. The FOR-species20K, developed by (Puliti et al., 2025b) is one of these datasets, it regroups a large tree point cloud annotated, offering a base to develop and compare a model. Previous work (Hamdani et al., 2026) explored several deep learning point-based architectures for urban tree classification using LiDAR data. However, despite these advances the optimization of PointNet++ and the influence of point cloud sampling size remain largely unexplored (Qian et al., 2022).

The main purpose of this work is to explore PointNet++ at the benchmark FOR-species20K, by evaluating the influence of size sampling. Particularly, we evaluate three-point cloud sampling configuration (4 096, 8 192 and 16 384 points per tree), in order to characterize geometric richness captured, time to calculate, and performance of classification. We reported the comparative result, discussed the model sensibility at sampling, and identified the input size that gives the optimal trade-off.

The main contribution of this article:
1-Evaluation of PointNet++ performance on drone LiDAR data from the FOR-species20K benchmark for tree species classification;
2-Analysis of the impact of point cloud sample size on species classification accuracy (4 096, 8 192, and 16 384 points per tree);
3-Systematic comparison of the three configurations to identify the optimal trade-off between modeling complexity and model performance.

Organization of the article:
The section 2 presents related works (traditional methods vs machine learning and deep learning techniques to identify tree species). Section three details the dataset FOR-species20K. Section 4 exposes materials and methods. The results and their analysis are presented in section 5. Section 6 discusses the implications, limitations, and perspectives, and section 7 conclusion.

2. Related Works

Tree classification is primarily carried out through field inventories. However, this approach is time-consuming, costly

and it relies heavily on experts. Furthermore, it cannot be used in dense environments or large forested areas.

Remote sensing, particularly UAS LiDAR, allows for automated inventory, reducing both time and costs. Combining airborne LiDAR with artificial intelligence produces promising results. However, machine learning methods require the manual extraction of geometric and/or radiometric features, as well as the use of traditional classifiers (SVM, Random Forests, etc) (Guo et al., 2020). These approaches are limited by noise and irregularities in point clouds, which reduce their capacity for generalization (Svoikin et al., 2023) (Rust and Stoinski, 2024). Deep learning methods overcome these limitations by processing the point cloud directly, without any prior extraction (Guo et al., 2020) (Kumar Vinodkumar et al., 2023). PointNet as a deep learning architecture (Qi et al., 2017a), was a significant advance because it can generate a global representation independently of the order of points. But it remains inadequate to capture local structure. PointNet++ incorporates a multi-scale hierarchy and combines the farthest point sampling with local feature aggregation. This mechanism captures global patterns while reducing the influence of redundant points or slight local noise (Qi et al., 2017b).
Another deep learning architecture appeared to analyse point cloud. KPConv, for Kernel Point Convolution, uses point kernels to capture local structures (Thomas et al., 2019). DGCNN, for Dynamic Graph of Convolutional Neural Networks, uses dynamic graphs to model relationships between neighboring point clouds (Wang et al., 2019). Rand-LA-Net, for Random sampling Local Aggregation Network, is able to reduce random points and process large point clouds (Hu et al., 2020). These architectures are often compared to PointNet++, and reveal that PointNet ++ performs well in vegetation classification (Briechle et al., 2020).

The objective of this study is to optimize PointNet++ for tree species classification based on drone LiDAR point clouds. This optimization will be conducted by testing three sampling configurations. The first with a subsampling of 4 096 points per tree, the second with 8 192 points per tree, and the third with 16 384 points per tree. This allows for the capture of increasingly detailed geometric information.

The classification model is based on an implementation of PointNet++, a hierarchical deep learning architecture adapted for point clouds. The network uses three Set Abstraction (SA) modules to capture local and global geometric patterns. Then, a MultiLayer Perceptron (MLP) performs the final classification. Figure 1 below illustrates this model architecture, which can be detailed through the following steps.

(i) Local Feature Extraction (SA1)

The first hierarchical module learns the finest geometric structures. It uses a neighborhood radius of 0.2 units and selects a maximum of two neighbors. The local MLP has an architecture of [3 + F, 64, 64, 128] (F = 0), where the first three entries correspond to the X, Y, and Z coordinates of a point, and F represents any additional characteristics (such as color or intensity). In this case, F = 0, so only the spatial coordinates are used. The MLP then has three hidden layers with 64, 64, and 128 neurons, respectively, allowing for the transformation and extraction of richer representations of the input points.

(ii) Intermediate Feature Extraction (SA2)

The second abstraction layer expands the receptive field to capture intermediate-scale patterns. Around each centroid, it groups up to 8 neighboring points within a radius of 0.25 units. The local MLP takes as input a vector of dimension 128 (features from SA1) + 3 (relative coordinates), with an architecture of [128+3 → 128 → 128 → 256].

(iii) Global Feature Aggregation (SA3)

The third and the final hierarchical module called Global Summary of Point Cloud (GlobalSAModule), treats all remaining points as a single group in order to extract a single global representation of the tree. The parameters of its global MLP are [256 +3, 256, 512, 1,024].

(iv) Final Classification MLP

The global 1,024-dimensional of SA3 is sent to a final MLP composed of four successive linear layers: 1,024 → 512 → 256 → 128 → C (where C is the number of species to be predicted, which is 7 in our case).

(v) Multi-Scale Grouping (MSG)

Multi-Scale Grouping is a strategy in PointNet++ that aggregates local features at multiple scales, allowing the network to capture both fine and coarse geometric patterns and handle variable point densities in 3D point clouds.

(vi) Activation functions

ReLU activation function is applied after each layer of the MLP, except the last one. Each layer of the final MLP is followed by batch normalization to stabilize learning. A dropout of 0.5 is applied to the final MLP to limit overfitting.

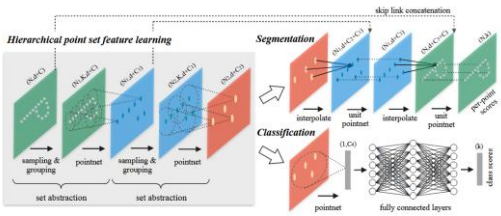

Figure 1: Illustration of PointNet++ learning architecture (Qi et al., 2017b).

3. Data used: FOR-species20K:

The FOR-species20K dataset was created through direct collaboration with researchers, public calls, and academic networks. It contains 20 158 individual trees, manually segmented, captured by three platforms: Terrestrial Laser Scanning, Unmanned Laser Scanning, and Mobile Laser Scanning (Puliti et al., 2025a).

Identifying species is complicated by the variability of tree shapes within a single species and the similarity between different species. The imbalance between species, where some are widely represented and others are rare, also reduces model performance. The density and quality of point clouds significantly impact the results, and generalizing models across different sites or sensors remains challenging (Puliti et al., 2025a).

This contribution has been peer-reviewed. The double-blind peer-review was conducted on the basis of the full paper.
https://doi.org/10.5194/isprs-annals-XI-1-2026-287-2026 | © Author(s) 2026. CC BY 4.0 License.

In addition, each platform acquisition (terrestrial, mobile, or ULS) produces a specific type of occlusion. Some trees were scanned with only one angle, creating blind spots in the canopy. Others were scanned from several directions, which provided a homogeneous point cloud. For this research, we chose to work with drone data in order to address the occlusion specific to this type of data and train a robust model, capable of generalizing to airborne datasets, where occlusion is more pronounced, as illustrated in Figure 2.

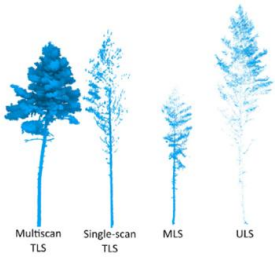

Figure 2 : Example variation of the point cloud quality across platforms, and different scanning protocols (TLS single- or multiscan, MLS and ULS) (Puliti et al., 2025a).

For this article, we are interested in evaluating a variation of point sampling in the performance of PointNet++ for the drone dataset. Therefore, we filtered FOR-species20K, we kept only trees captured by the UAS platform about 16 species: (*Abies alba; Acer campestre; Acer pseudoplatanus; Betula pendula; Carpinus betulus; Fagus sylvatica; Fraxinus excelsior; Larix decidua; Picea abies; Pinus radiata; Pinus sylvestris; Prunus avium; Pseudotsuga menziesii; Quercus petraea; Quercus robur; Quercus rubra).* However, we quickly realized that the number of trees in each species was highly variable, creating an imbalance in the classes. This issue constitutes a significant obstacle in classification, as dominant classes bias model training and reduce accuracy for smaller classes. Some species, such as *Picea abies* or *Fagus sylvatica*, have up to 649 trees in training and 139 in testing, while other species, such as *Quercus robur* or *Acer campestre*, have only one to seven trees in testing. To mitigate this issue, we applied a second filtering step by retaining only classes with at least 100 trees per species. This resulted in seven final species, whose distribution is illustrated in Figure 3: *Betula pendula* (class 0); *Carpinus betulus* (class 1); *Fagus sylvatica* (class 2); *Picea abies* (class 3); *Pinus sylvestris* (class 4); *Pseudotsuga menziesii* (class 5); *Quercus petraea* (class 6). Figure 4 provides a visual representation of these seven tree species used in this article.

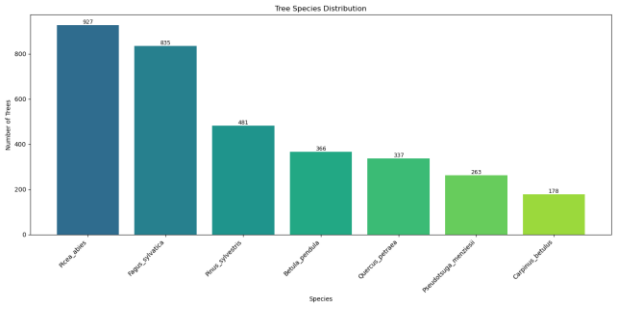

Figure 3: Distribution of tree species used in this study.

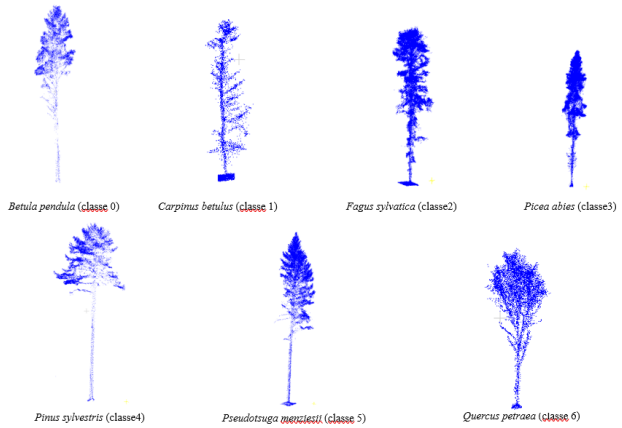

Figure 4: Visualization of seven Tree Species acquired by drone LiDAR used in this experimentation.

In order to ensure robust training, we divide the filtered data into three categories: training, validation, and testing. The training dataset is used to adjust the model's weights, the validation data is used to prevent overfitting and regulate training, and the test data is used to evaluate the model's final performance. In the end, we kept a total of 3 387 trees, with 2 879 for training and validation and 508 for testing. The distribution of tree species across the three subsets is illustrated in Figure 5.

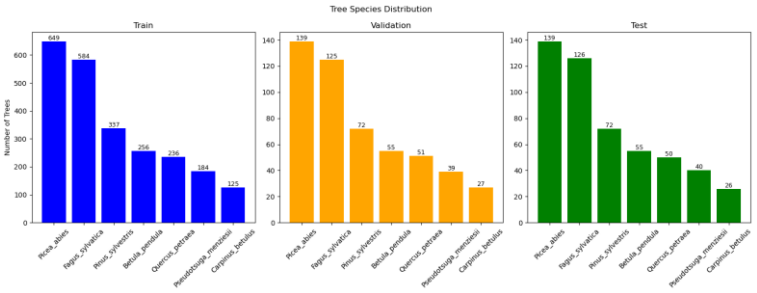

Figure 5: Repartition of tree species for each category: train, validation and test dataset.

4. Materials and Methods

This section outlines the materials and methods that we used in our study. First, we selected the “For-species20k” dataset. We then performed preprocessing and sampled three configurations, training PointNet++ for each one. Finally, we evaluated the performance of PointNet++ for each configuration. Figure 6 below illustrates our workflow:

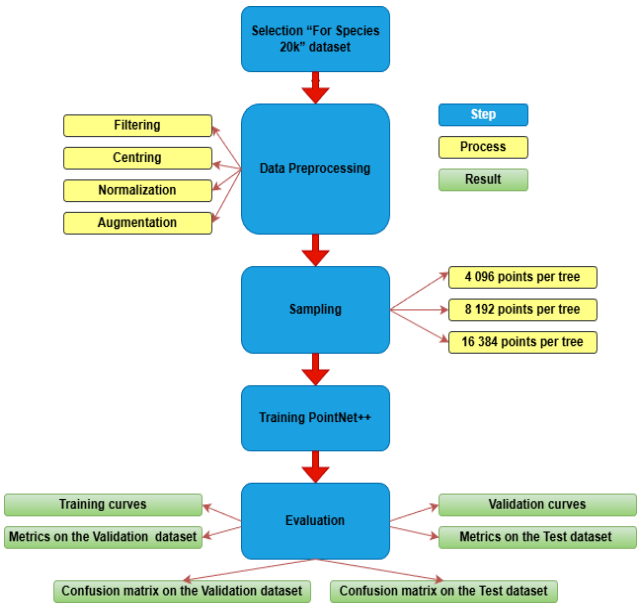

Figure 6: Workflow for our approach.

4.1 Data Preprocessing:

Data preprocessing is an important step in deep learning. It includes several operations to prepare the data for the model’s training. These operations are as follows:

(i)Filtering: FOR-species20K dataset has been filtered to retain only trees acquired by drone LiDAR. Additionally, we keep only species that present at least 100 trees, about seven species.

(ii)Centering: Each point cloud has been centered around the origin of the coordinate system (0, 0, 0). This operation ensures that the model learns geometric shapes rather than positions.

(iii)Normalization: Points are normalized in order to adjust the scale of each tree. This step ensures that all trees have approximately the same average size.

(iv)Augmentation (random rotations): Each tree is duplicated six times, with rotation angles chosen from six fixed values (0°, 60°, 120°, 180°, 240°, and 300°). This preprocessing enriches the dataset and improves model generalization.

4.2 Sampling:

The number of points per tree is an essential parameter to preserve geometric information. In order to assess the effect of input point density on the classification performance of PointNet++, we tested three configurations:

(i)4 096 points per tree: each tree point cloud was uniformly sampled and limited to a maximum of 4 096 points, representing a low point density input configuration;

(ii) 8 192 points per tree: the number of input points was doubled compared to the low-density configuration, resulting in 8 192 points per tree (medium-density configuration);

(iii)16 384 points per tree: the input size was further doubled relative to the medium density configuration, leading to 16 384 points per tree, representing a high-density input configuration.

All experiments relied on the same random point sampling strategy, ensuring consistency across configurations.

4.3 Training and validation:

The training pipeline is strictly implemented to ensure the robustness and reproducibility of the results. We used 7 trees per batch, a learning rate of 5×10^{-5}, and 105 epochs. At each epoch, the following metrics are recorded: average loss (training and validation), accuracy, recall, F1-score, and confusion matrix. The model automatically saves the best model as soon as an improvement is detected in the weighted average of the F1-scores per class. The final evaluation is performed on an independent test set to measure generalization ability.

(i)Implementation: the pipeline is implemented with Python libraries. The model architecture is based on PyTorch. PyTorch Geometric is used as an extension for processing 3D point clouds, providing the essential modules for implementing PointNet++.

(ii)Loss function: The loss function used is NLL Loss (Negative Log-Likelihood Loss). It is adapted to the LogSoftmax output and weighted according to the frequency of classes in the training data in order to mitigate the imbalance between species classes.

(iii)Optimizer used is Adam, an adaptive gradient descent optimization algorithm.

(iv)Hardware: PyTorch offers native compatibility with GPUs, allowing efficient model training on CUDA hardware, which is essential for processing large 3D point clouds.

5. Results

The results obtained for each classification model configuration are detailed below, based on overall accuracy (OA), the F1-score per class, and the analysis of the confusion matrix.

5.1 Training & Validation curves

For each sampling configuration (4 096, 8 192 and 16 384), the model was trained using three different random seeds (0,42 and 123) in order to assess the stability of the training process and ensure the reproducibility of the results. The shaded area around the curves illustrates the variability observed across the different seeds.

5.1.1 Results for the configuration with 4 096 points per tree

The training behavior of the PointNet++ model using 4 096 points per tree is illustrated in Figure 7, which shows the training loss (orange), validation loss (blue), and validation accuracy (black). The training loss decreases gradually and continuously from epochs, from around 1.6 at the start of training to close to 0.05 after 100 epochs, which indicates good model convergence. The validation loss decreases rapidly during the first epochs, then stabilizes around values between 0.55 and 0.65. From around epoch 30, we constant a small gap between training and validation curves, suggesting a slight overfitting. However, the validation curve remains stable and validation accuracy is still increasing, indicating that the model still keeps good generalization ability.

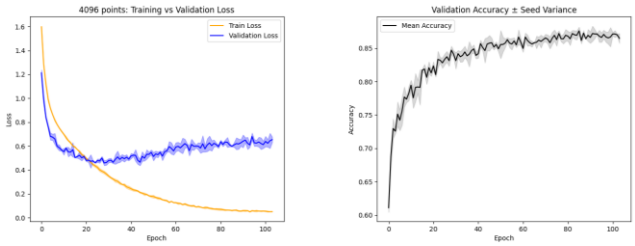

Figure 7: Training loss (orange), validation loss (blue) and validation accuracy (black) obtained with PointNet++ architecture using 4 096 points per tree.

5.1.2 Results for the configuration with 8 192 points per tree

The evolution of the training process for the PointNet++ model with 8 192 points per tree is shown in Figure 8, which reports the training loss (orange), validation loss (blue), and validation accuracy (black). The training loss decreases steadily throughout epochs, from around 1.6 at the start of training to nearly 0.05 after 100 epochs, reflecting the model’s effective convergence. Validation loss decreases during the first iterations before stabilizing around 0.55-0.65. From epoch 30 onwards, a slight gap appears between the two curves, suggesting overfitting. Nevertheless, this divergence remains moderate and overall performance continues to improve. In fact, validation accuracy gradually increases to reach an average value of around 0.87, with low variance between the different initialization seeds, confirming the stability and generalization capacity of the model.

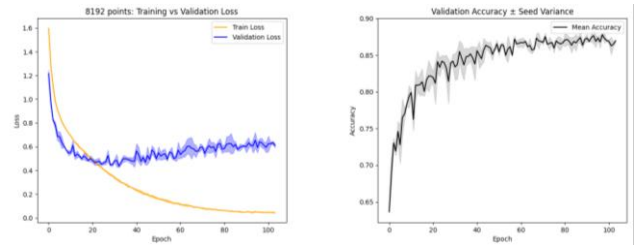

Figure 8: Training loss (orange), validation loss (blue) and validation accuracy (black) obtained with PointNet++ architecture using 8 192 points per tree.

5.1.3 Results for the configuration with 16 384 points per tree:

The training dynamics of the PointNet++ model using 16 384 points per tree are illustrated in Figure 9, which shows the training loss (orange), validation loss (blue), and validation accuracy (black). The training loss exhibits a rapid decrease during the initial epochs, indicating that the model is effectively learning the characteristics of the point clouds. The validation loss also decreases at the beginning, reaching a minimum around the first few tens of epochs, before stabilizing and then increasing slightly. This gradual divergence between the two curves suggests the onset of slight overfitting as training continues. Despite this, convergence remains stable, indicating that the model correctly captures the discriminating structures of the different tree species.

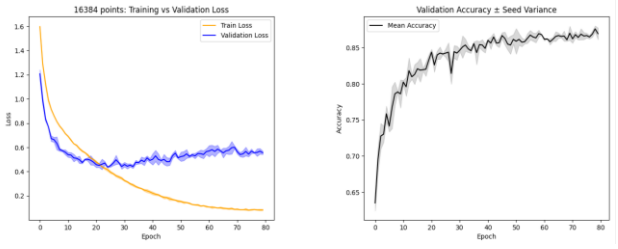

Figure 9: Training loss (orange), validation loss (blue) and validation accuracy (black) obtained with PointNet++ architecture using 16 384 points per tree.

5.2 Test accuracy and main metrics:

All results for the validation set are reported as the mean and standard deviation over three runs with different random seeds (0,42, and 123). For the test set, we used the best seed to predict the species. For all three configurations, seed 42 produced the best result, so we used it to predict species.

5.2.1 Results for the configuration with 4 096 points per tree

- **On the validation set (3 seeds):**

Table 1 reports the main evaluation metrics for the configuration using 4 096 points per tree on the validation set. The results correspond to average over three random seeds (0, 42 and 123), reported as mean ± standard deviation.

Result for configuration with 4 096 points per tree on the validation set (optimal model at the epoch: 104)				
Overall accuracy (%)	Mean loss	Mean (Weighted) F1-score	Weighted average accuracy	Weighted average recall
88.25±0.40	0.654 ± 0.022	0.846±0.010	0.882±0.004	0.882±0.004

Table 1 : Main metrics for the configuration with 4 096 points per tree for the validation dataset (mean ± standard deviation across three random seeds: 0, 42, and 123).

- **On the test set:**

Table 2 summarizes the main evaluation metrics for the configuration using 4 096 points per tree on the test set. The model correctly classified 463 out of 508 trees, resulting in 45 misclassifications.

Result for configuration with 4 096 points per tree on the test set	
Overall accuracy (%)	Macro-average F1-score (%)
91.14	88.49

Table 2 : Main metrics for the configuration with 4 096 points per tree for the test dataset.

This contribution has been peer-reviewed. The double-blind peer-review was conducted on the basis of the full paper.
https://doi.org/10.5194/isprs-annals-XI-1-2026-287-2026 | © Author(s) 2026. CC BY 4.0 License.

5.2.2 Results for the configuration with 8 192 points per tree

❖ **On the validation set (3 seeds):**

Table 3 reports the main evaluation metrics for the configuration using 8 192 points per tree on the validation set. The results are computed over three random seeds (0,42 and 123) and are presented as mean ± standard deviation.

Result for configuration with 8 192 points per tree on the validation set (optimal model at the epoch: 104)				
Overall accuracy (%)	Mean loss	Mean (Weighted) F1-score	Weighted average accuracy	Weighted average recall
88.58±0.43	0.610± 0.019	0.883±0.005	0.886±0.004	0.886±0.004

Table 3 : Main metrics for the configuration with 8 192 points per tree for the validation dataset (mean ± standard deviation across three random seeds: 0, 42, and 123).

❖ **On the test set**

Table 4 summarizes the main evaluation metrics for the configuration using 8 192 points per tree on the test set. The model correctly classified 469 out of 508 trees, resulting in 39 misclassifications.

Result for configuration with 8 192 points per tree on the test set	
Overall accuracy (%)	Macro-average F1-score (%)
92.32	89.65

Table 4 : Main metrics for the configuration with 8 192 points per tree for the test dataset.

5.2.3 Results for the configuration with 16 384 points per tree:

❖ **On the validation set (3 seeds):**

Table 5 reports the main evaluation metrics for the configuration using 16 384 points per tree on the validation set. The results are computed over three random seeds (0,42 and 123) and are presented as mean ± standard deviation.

Result for configuration with 16 384 points per tree on the validation set (optimal model at the epoch: 80)				
Overall accuracy (%)	Mean loss	Mean (Weighted) F1-score	Weighted average accuracy	Weighted average recall
87.99±0.16	0.556±0.014	0.880±0.000	0.880±0.002	0.880± 0.002

Table 5 : Main metrics for the configuration with 16 384 points per tree for the validation dataset (mean ± standard deviation across three random seeds: 0, 42, and 123).

❖ **On the test set:**

Table 6 summarizes the main evaluation metrics for the configuration using 16 384 points per tree on the test set. The model correctly classified 464 out of 508 trees, resulting in 44 misclassifications.

Result for configuration with 16 384 points per tree on the test set	
Overall accuracy (%)	Macro-average F1-score (%)
91.34	89.43

Table 6 : Main metrics for the configuration with 16 384 points per tree for the test dataset.

5.3 Confusion matrix:

5.3.1 Results the configuration with 4 096 points per tree

❖ **On the validation set:**

The average confusion matrix for the configuration of 4 096 points per tree is shown in Figure 10. It exhibits a strong diagonal structure, indicating that most tree species are correctly classified. The classes *Picea abies*, *Fagus sylvatica* and *Pinus sylvestris* have a large number of correctly classified samples. However, some confusion appears between certain species with similar structures, particularly between *Carpinus betulus* and *Fagus sylvatica*, as well as between *Betula pendula* and *Pinus sylvestris*.

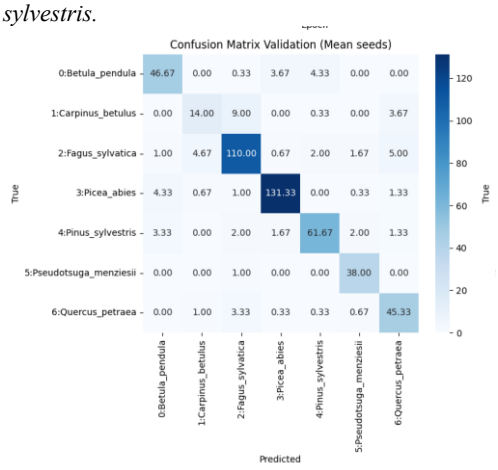

Figure 10: Average Confusion Matrix on the validation dataset for PointNet++ (4 096 points per tree) across three random seeds (0,42, 123).

❖ **On the test set**

The confusion matrix on the test set for the configuration using 4 096 points per tree is presented in Figure 11, based on the best-performing model obtained with seed42. The results highlight a strong generalization capability on unseen data, with most predictions concentrated along the diagonal, indicating correct classification for the majority of samples. For example, Picea abies is correctly classified in 111 out of 115 cases (111/115), while Fagus sylvatica has 109 correct predictions out of 115 (109/115). Similarly, Pinus sylvestris obtains 110 correct classifications out of 115 (110/115). The species Pseudotsuga menziesii is perfectly recognized with 115 correct predictions out of 115 (115/115). Some confusion remains, mainly between Betula pendula and Pinus sylvestris, as well as between Carpinus betulus and Fagus sylvatica, which can be explained by similarities in the structure of their crowns observed in the LiDAR data.

This contribution has been peer-reviewed. The double-blind peer-review was conducted on the basis of the full paper.
https://doi.org/10.5194/isprs-annals-XI-1-2026-287-2026 | © Author(s) 2026. CC BY 4.0 License.

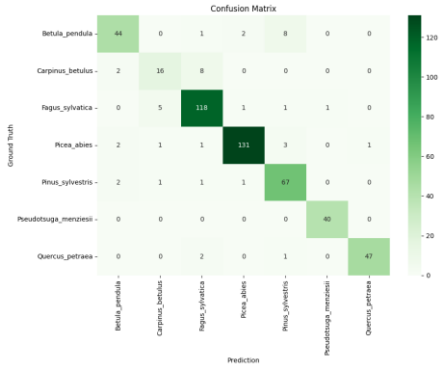

Figure 11: Confusion matrix on test dataset for the configuration with 4 096 points per tree (with the best model obtains in seed 42).

5.3.2Results for the configuration with 8 192 points per tree

❖ **On the validation set**

The average confusion matrix for the configuration using 8 192 points per tree is shown in Figure 12. It exhibits a dominant diagonal structure, indicating a high overall classification performance. The species *Picea abies*, *Fagus sylvatica*, *Pseudotsuga menziesii* and *Pinus sylvestris* have a highest rate, with limited errors. However, some confusion persists for *Betula pendula* and *Carpinus betulus,* mainly with *Pinus sylvestris* and *Fagus sylvatica*, respectively.

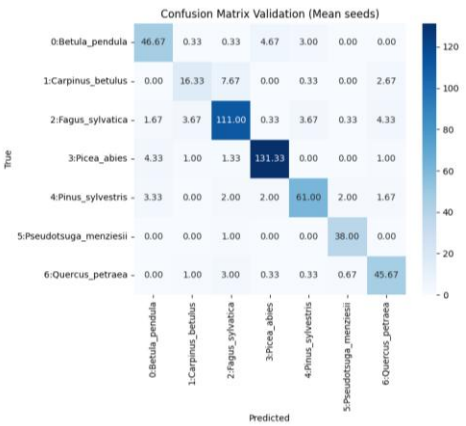

Figure 12: Average Confusion Matrix on the validation dataset for PointNet++ (8 192 points per tree) across three random seeds (0,42, 123).

❖ **On the test set**

The confusion matrix on the test set for the configuration using 8 192 points per tree is presented in Figure 13, based on the best-performing model obtained with seed 42. The matrix exhibits a strong diagonal structure, indicating robust generalization and effective discrimination between species. The best classification performance is observed for *Picea abies* (135/139), *Fagus sylvatica* (119/126) and *Pseudotsuga menziesii* (40/40), which achieve very high classification rates. *Pinus sylvestris* (68/72), and *Quercus petraea* (47/50) are also correctly identified in most cases. The main confusions concern closely related deciduous species: *Betula pendula* (44/55) is particularly confused with *Pinus sylvestris*, and *Carpinus betulus* (16/26), which is sometimes confused with *Fagus sylvatica*.

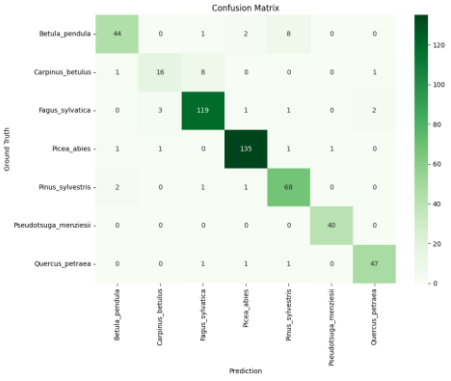

Figure 13: Confusion matrix on test dataset for the configuration with 8 192 points per tree (with the best model obtains in seed 42).

5.3.3Results for the configuration with 16 384 points per tree

❖ **On the validation set**

The average confusion matrix for the PointNet++ configuration using 16 384 points per tree on the validation set is shown in Figure 14. It shows that the majority of predictions lie on the main diagonal, indicating that the model is good at distinguishing between species. The *Picea abies* and *Fagus sylvatica* classes have the best classification rates, with a high number of correct predictions. However, some confusion appears between species with similar structural characteristics in the LiDAR data, particularly between *Carpinus betulus* and *Fagus sylvatica*, and between *Betula pendula* and *Pinus sylvestris*. These errors may be related to the similarity of crown structures or an imbalance in the training data.

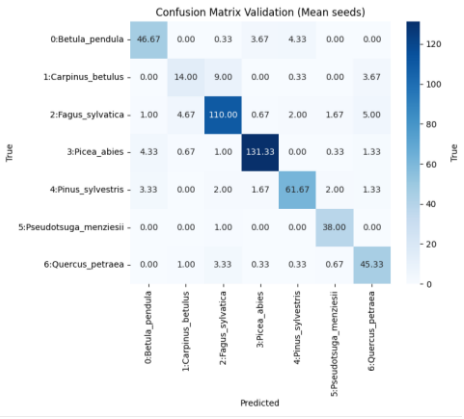

Figure 14: Average Confusion Matrix on the validation dataset for PointNet++ (16 384 points per tree) across three random seeds (0,42, 123).

❖ **On the test set**

The confusion matrix on the test set for the configuration using 16 384 points per tree is presented in Figure 15, based on the best-performing model obtained with seed 42. The results

indicate good overall performance of the model, with a majority of correct predictions concentrated on the main diagonal. Several species are particularly well recognized, notably *Picea abies*, *Fagus sylvatica*, and *Pinus sylvestris*. More specifically, the proportion of correctly identified trees is *Betula pendula* (41/55), *Carpinus betulus* (20/26), *Fagus sylvatica* (118/126), *Picea abies* (130/139), *Pinus sylvestris* (70/72), *Pseudotsuga menziesii* (39/40), and *Quercus petraea* (46/49). However, some confusion remains between certain species with similar crown structures in the LiDAR data, particularly between *Carpinus betulus* and *Fagus sylvatica*, and between *Betula pendula* and *Pinus sylvestris*.

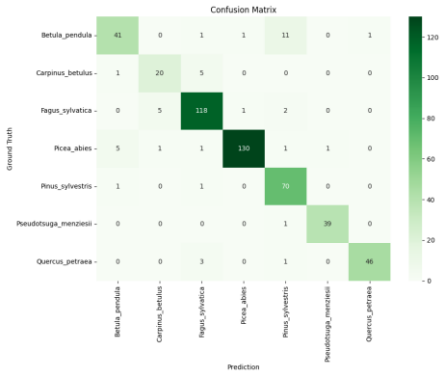

Figure 15: Confusion matrix on test dataset for the configuration with 16 384 points per tree (with the best model obtains in seed 42).

5.3 Model training time:

All models were trained on a machine equipped with an NVIDIA Quadro P5000 GPU (16GB VRAM, CUDA 12.4). It is observed that the training time increases with the number of points per tree. Figure 16 presents the average training time for each configuration (4 096, 8 192, and 16 384 points per tree), computed over the three random seeds (0, 42 and 123).

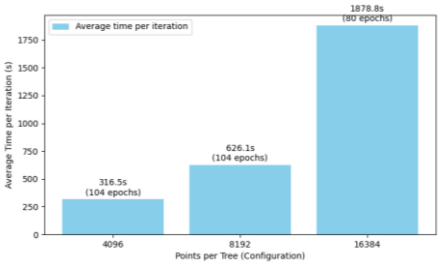

Figure 16: Average model training time per configurations (4 096, 8 192, and 16 384 points per tree), calculated using three seeds (0, 42 and 123).

6. Discussion

The results obtained for the three configurations (4 096, 8 192, and 16 384 points per tree) are summarized in Table 7 and illustrated in Figure 17, in order to compare their performance in terms of F1 score.

Species	F1-score per species		
	4 096 points per tree	8 192 points per tree	16 384 points per tree
Betula pendula	0.84	0.85	0.80
Carpinus betulus	0.65	0.70	0.77
Fagus sylvatica	0.92	0.93	0.93
Picea abies	0.96	0.97	0.96
Pinus sylvestris	0.88	0.90	0.89
Pseudotsuga menziesii	0.99	0.99	0.97
Quercus petraea	0.96	0.94	0.95

Table 7: Comparison of F1 scores obtained for the three sampling configurations (4 096, 8 192, and 16 384 points per tree).

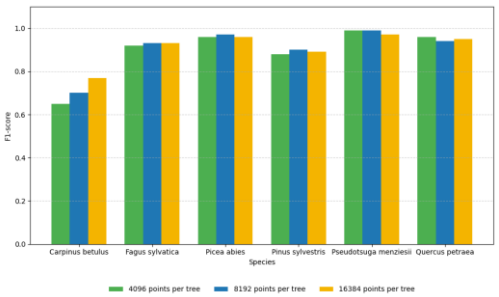

Figure 17: Evolution of F1 scores for test data according to the three sampling configurations (4 096, 8 192, and 16 384 points per tree).

The analysis highlights the influence of the number of points per tree on classification performance. With 4 096 points, the model already achieves robust results for most species, particularly *Pseudotsuga menziesii* (0.99), *Picea abies* (0.96), and *Quercus petraea* (0.96), while *Betula pendula* (0.84) and *Pinus sylvestris* (0.88) show more modest but acceptable performance. The increase to 8 192 points improves the results for *Betula pendula* (0.85) and *Pinus sylvestris* (0.90), and maintains high scores for the other species. This configuration appears to be the most balanced, as it combines a high level of overall performance with good stability between species. On the other hand, the increase to 16 384 points does not improve the result: while the scores remain acceptable for most species, *Betula pendula* (0.80) and *Pseudotsuga menziesii* (0.97) show a slight deterioration.

Overall, the 8 192-point configuration appears to be the best compromise, offering an optimal balance between the fineness of the tree representation, the robustness of the classification and reasonable processing time. However, *Carpinus betulus* remains the most difficult species to distinguish, regardless of point density, highlighting the need for complementary approaches (spectral characteristics, contextual characteristics, or adaptive density strategies).

7. Conclusion

Tree species classification is crucial for environmental science, biodiversity and urban forest management. It also helps organisms to understand the needs of each tree species in terms of irrigation and cutting to avoid long branches. They also provide a good habitat environment.

This contribution has been peer-reviewed. The double-blind peer-review was conducted on the basis of the full paper.
https://doi.org/10.5194/isprs-annals-XI-1-2026-287-2026 | © Author(s) 2026. CC BY 4.0 License.

Traditional methods of identifying tree species require the involvement of experts in the field. This method is time-consuming and costly, and it may produce false results in complex areas. In this article, we used drone LiDAR data to optimize the PointNet++ deep learning architecture for tree species classification, by evaluating the impact of point cloud sampling size on the model’s performance. The results show that the 4 096-point configuration produces robust results for most species, the 8 192-point configuration strikes the best balance between geometric representation and strong classification stability across species. However, increasing to 16 384 points reduces performance for some species and demands considerable computational resources for model training.

Carpinus betulus is a species that is difficult to identify in all configurations, suggesting significant interspecific confusion, primarily with *Fagus sylvatica*. This emphasizes the importance of future works to incorporate additional data sources, such as spectral information, and test alternative deep learning architectures, such as RandLA-Net, DGCNN and KPConv, to enhance tree species classification.

Overall, our findings demonstrate the impact of point sampling size on improving the performance of the PointNet++ model, providing a more reliable framework for LiDAR tree species classification. Our results reinforce also the potential of using drone data and deep learning methods to automate tree species classification.

Acknowledgements

The authors acknowledge Puliti et al. (2025) for making the open-source dataset “FOR-species20K” publicly available at (https://zenodo.org/records/13255198), which was fundamental to the completion of this work.

The authors would like to express their sincere gratitude to the “Communauté d’Agglomération Grand Paris Sud (CA GPS)”, particularly the GIS division of the Department of Foresight, Territorial Observation and GIS (Direction de la Prospective, de l’Observation Territoriale et du SIG), for their valuable support and active involvement throughout this research.

This article is based upon work from COST Action CA21138 (CLEANFOREST) “Joint effects of CLimate Extremes and Atmospheric depositioN on European FORESTs”, supported by COST (European Cooperation in Science and Technology).

References

Briechle, S., Krzystek, P., Vosselman, G., 2020. classification of tree species and standing dead trees by fusing uav-based lidar data and multispectral imagery in the 3d deep neural network pointnet++. ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences V-2–2020, 203–210. https://doi.org/10.5194/isprs-annals-V-2-2020-203-2020

Guo, Y., Wang, H., Hu, Q., Liu, H., Liu, L., Bennamoun, M., 2020. Deep Learning for 3D Point Clouds: A Survey. https://doi.org/10.48550/arXiv.1912.12033

Haase, D., Dushkova, D., 2024. Embracing ambivalence as the key to promoting tree diversities as nature-based solutions in European cities. Urban Ecosyst 27, 1837–1846. https://doi.org/10.1007/s11252-024-01555-9

Hamdani, N., Abouhat, I., Ait El Kadi, K., Bensiali, S., Sebari, I., 2026. Urban Tree Classification from Multispectral Airborne LiDAR Using PointNet, DGCNN & RandLA-Net. Int. Arch. Photogramm. Remote Sens. Spatial Inf. Sci. XLVIII-4/W19-2025, 55–61. https://doi.org/10.5194/isprs-archives-XLVIII-4-W19-2025-55-2026

Hu, Q., Yang, B., Xie, L., Rosa, S., Guo, Y., Wang, Z., Trigoni, N., Markham, A., 2020. RandLA-Net: Efficient Semantic Segmentation of Large-Scale Point Clouds. https://doi.org/10.48550/arXiv.1911.11236

Kongwen, Z., Baoxin, H., 2012. Individual Urban Tree Species Classification Using Very High Spatial Resolution Airborne Multi-Spectral Imagery Using Longitudinal Profiles. Remote Sensing 4, 1741–1757. https://doi.org/10.3390/RS4061741

Kumar Vinodkumar, P., Karabulut, D., Avots, E., Ozcinar, C., Gholamreza, 2023. A Survey on Deep Learning Based Segmentation, Detection and Classification for 3D Point Clouds. Entropy 25, 635–635. https://doi.org/10.3390/e25040635

Locosselli, G.M., Buckeridge, M.S., 2023. The science of urban trees to promote well-being. Trees 37, 1–7. https://doi.org/10.1007/s00468-023-02389-2

Puliti, S., Lines, E.R., Müllerová, J., Frey, J., Schindler, Z., Straker, A., Allen, M.J., Winiwarter, L., Rehush, N., Hristova, H., Murray, B., Calders, K., Coops, N., Höfle, B., Irwin, L., Junttila, S., Krůček, M., Krok, G., Král, K., Levick, S.R., Luck, L., Missarov, A., Mokroš, M., Owen, H.J.F., Stereńczak, K., Pitkänen, T.P., Puletti, N., Saarinen, N., Hopkinson, C., Terryn, L., Torresan, C., Tomelleri, E., Weiser, H., Astrup, R., 2025a. Benchmarking tree species classification from proximally sensed laser scanning data: Introducing the FOR-species20K dataset. Methods in Ecology and Evolution 18. https://doi.org/10.1111/2041-210X.14503

Puliti, S., Lines, E.R., Müllerová, J., Frey, J., Schindler, Z., Straker, A., Allen, M.J., Winiwarter, L., Rehush, N., Hristova, H., Murray, B., Calders, K., Coops, N., Höfle, B., Irwin, L., Junttila, S., Krůček, M., Krok, G., Král, K., Levick, S.R., Luck, L., Missarov, A., Mokroš, M., Owen, H.J.F., Stereńczak, K., Pitkänen, T.P., Puletti, N., Saarinen, N., Hopkinson, C., Terryn, L., Torresan, C., Tomelleri, E., Weiser, H., Astrup, R., 2025b. Benchmarking tree species classification from proximally sensed laser scanning data: Introducing the FOR-species20K dataset. Methods in Ecology and Evolution 16, 801–818. https://doi.org/10.1111/2041-210X.14503

Qi, C.R., Su, H., Mo, K., Guibas, L.J., 2017a. PointNet: Deep Learning on Point Sets for 3D Classification and Segmentation. https://doi.org/10.48550/arXiv.1612.00593

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

Qian, G., Li, Y., Peng, H., Mai, J., Hammoud, H.A.A.K., Elhoseiny, M., Ghanem, B., 2022. PointNeXt: Revisiting PointNet++ with Improved Training and Scaling Strategies. Presented at the Advances in Neural Information Processing Systems.

Rust, S., Stoinski, B., 2024. Enhancing Tree Species Identification in Forestry and Urban Forests through Light Detection and Ranging Point Cloud Structural Features and Machine Learning. Forests. https://doi.org/10.3390/f15010188

Svoikin, F., Zhuk, K., Svoikin, V., Ugryumov, S., Bacherikov, I., Iniesta, D.V., Ryapukhin, A., 2023. Classification of Tree Species in the Process of Timber-Harvesting Operations Using Machine-Learning Methods. Inventions 8, 57. https://doi.org/10.3390/inventions8020057

Taher, J., Hyyppä, E., Hyyppä, M., Salolahti, K., Yu, X., Matikainen, L., Kukko, A., Lehtomäki, M., Kaartinen, H., Thurachen, S., Litkey, P., Luoma, V., Holopainen, M., Kong, G., Fan, H., Rönnholm, P., Vaaja, M., Polvivaara, A., Junttila, S., Vastaranta, M., Puliti, S., Astrup, R., Kostensalo, J., Myllymäki, M., Kulicki, M., Stereńczak, K., Pires, R. de P., Valbuena, R., Carbonell-Rivera, J.P., Torralba, J., Chen, Y.-C., Winiwarter, L., Hollaus, M., Mandlburger, G., Takhtkeshha, N., Remondino, F., Lisiewicz, M., Kraszewski, B., Liang, X., Chen, J., Ahokas, E., Karila, K., Vezeteu, E., Manninen, P., Näsi, R., Hyyti, H., Pyykkönen, S., Hu, P., Hyyppä, J., 2026. Multispectral airborne laser scanning for tree species classification: A benchmark of machine learning and deep learning algorithms. ISPRS Journal of Photogrammetry and Remote Sensing 233, 278–309. https://doi.org/10.1016/j.isprsjprs.2026.01.031

Thomas, H., Qi, C.R., Deschaud, J.-E., Marcotegui, B., Goulette, F., Guibas, L.J., 2019. KPConv: Flexible and Deformable Convolution for Point Clouds. https://doi.org/10.48550/arXiv.1904.08889

Wang, Y., Sun, Y., Liu, Z., Sarma, S.E., Bronstein, M.M., Solomon, J.M., 2019. Dynamic Graph CNN for Learning on Point Clouds. https://doi.org/10.48550/arXiv.1801.07829