HAL open science

Predicting Vegetation Stratum Occupancy from Airborne LiDAR Data with Deep Learning

Ekaterina Kalinicheva, Loic Landrieu, Clément Mallet, Nesrine Chehata

To cite this version:

Ekaterina Kalinicheva, Loic Landrieu, Clément Mallet, Nesrine Chehata. Predicting Vegetation Stratum Occupancy from Airborne LiDAR Data with Deep Learning. International Journal of Applied Earth Observation and Geoinformation, 2022, 112, pp.102863. 10.1016/j.jag.2022.102863 . hal-03727656

HAL Id: hal-03727656

https://hal.science/hal-03727656v1

Submitted on 19 Jul 2022

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Distributed under a Creative Commons Attribution 4.0 International License

International Journal of Applied Earth Observations and Geoinformation 112 (2022) 102863

Contents lists available at ScienceDirect

International Journal of Applied Earth Observations and Geoinformation

journal homepage: www.elsevier.com/locate/jag

Predicting Vegetation Stratum Occupancy from Airborne LiDAR Data with Deep Learning

Ekaterina Kalinicheva [a,b], Loic Landrieu [a], Clément Mallet [a], Nesrine Chehata [a,c]

[a] LASTIG, Univ Gustave Eiffel, IGN, ENSG, F-94160 Saint-Mande, France
[b] INRAE, UMR 1202 BIOGECO, Université de Bordeaux, France
[c] EA G&E Bordeaux INP, Université Bordeaux Montaigne, France

ARTICLE INFO

Keywords:
Vegetation analysis
Stratum
Airborne LiDAR
Deep neural networks
Weakly-supervised learning

ABSTRACT

We propose a new deep learning-based method for estimating the occupancy of vegetation strata from airborne 3D LiDAR point clouds. Our model predicts rasterized occupancy maps for three vegetation strata corresponding to lower, medium, and higher cover. Our weakly-supervised training scheme allows our network to only be supervised with vegetation occupancy values aggregated over cylindrical plots containing thousands of points which are typically easier to produce than pixel-wise or point-wise annotations. We propose to employ a deep neural network operating on 3D points, and whose prediction are projected onto rasters representing the different vegetation strata. Our method outperforms handcrafted, regression and deep learning baselines in terms of precision by up to 30%, while simultaneously providing visual and interpretable predictions. We provide an open-source implementation along with a dataset of 199 agricultural plots to train and evaluate weakly supervised occupancy regression algorithms.

1. Introduction

Estimating the structure of vegetation is a crucial first step for many environmental and ecological applications (Daubenmire, 1959; Bergen et al., 2009; Morsdorf et al., 2010). This is typically a time-consuming, undertaking, and often performed with *in situ* visual approximate measurements (Willem et al., 2000). Knowledge about the structure of vegetation is essential for pasture land management (Velthof et al., 2014), and helps to better model the risk of forest fire (MacLean, 1996; McKenzie and Crystal, 2011; Sandberg et al., 2001).

The progress in hardware precision and portability allows public and private actors to gather large quantities of geometric and radiometric data from airborne platforms (Chen, 2007). Such data sources are particularly well suited for vegetation analysis (Ferraz et al., 2016; Secord and Zakhor, 2007; Strîmbu and Strîmbu, 2015). Bolstered by the compelling performance (Guo et al., 2020) and increasing accessibility (Chaton et al., 2020) of deep learning for 3D point cloud analysis, we propose a deep learning approach to solve the problem of stratum occupancy prediction at the plot level. Our network can output two-dimensional occupancy maps for three vegetation strata relevant to land pasture management: lower, medium, and higher vegetation. Each of these stratum occupancy maps is presented in a form of a regular grid-a rasterized map — of circular shape with a user-defined pixel size. Our algorithm is developed for 10 m-radius plot-level data with the purpose of using the final model for large-scale mapping at the parcel level by dividing the area of interest into habitual plot-size samples. A benefit of our approach is that it can be entirely supervised with values describing the average stratum occupancy over cylindrical plots, which can contain thousands of points. Such *aggregated* values are much easier to produce than point-wise or pixel-wise annotations.

Automated Vegetation Analysis. Vegetation analysis covers multiple tasks, depending on the level of analysis (tree-based, stand-based, plot-based, etc) and the area of interest (urban VS natural environments). During the two last decades, remote sensing has shown to be the most suitable solution for automatic information extraction (Coops et al., 2021; Hildebrandt, 1990; Lechner et al., 2020; Pekkarinen et al., 2009), such as individual tree detection (Ferraz et al., 2016; Hyyppa et al., 2001; Reitberger et al., 2009; Strîmbu and Strîmbu, 2015; Vega et al., 2014; Wan Mohd Jaafar et al., 2018), tree species classification (Dechesne et al., 2017; Diedershagen et al., 2004), and structural and biophysical analysis of vegetation (Bouvier et al., 2015; Dutta et al., 2017; Lefsky et al., 1999; Latifi et al., 2015). Canopy analysis at the tree level, which entails biomass estimation, is typically conducted by combining the characteristics of individual trees. However, small trees

E-mail address: ekaterina.kalinicheva@ign.fr (E. Kalinicheva).

https://doi.org/10.1016/j.jag.2022.102863
Received 26 November 2021; Received in revised form 9 March 2022; Accepted 8 June 2022
1569-8432/© 2022 The Authors. Published by Elsevier B.V. This is an open access article under the CC BY-NC-ND license (http://creativecommons.org/licenses/by-nc-nd/4.0/).

are often missed by segmentation algorithms, leading to a less precise estimation of understory cover (Williams et al., 2020). Therefore, several works propose to focus on modeling the understory layer explicitly (Venier et al., 2019; Campbell et al., 2018; Wing et al., 2012) or as one of several vegetation layers (Latifi et al., 2015). Several works also propose to characterize the stratification of vegetation (Ferraz et al., 2009; Morsdorf et al., 2010). In the present paper, our objective is to automatically derive two-dimensional occupancy maps for different vegetation strata.

Use of 3D LiDAR Sensors in Forestry. The emergence of high-performing and compact LiDAR sensors has increased significantly the use of 3D data obtained from aerial platforms (Ferraz et al., 2016; Hyyppa et al., 2001; Strîmbu and Strîmbu, 2015). It has enabled operational forest mapping and inventory both at local and national scales (Naesset, 2007). Indeed, contrary to optical images that only capture the upper vegetation layer, LiDAR is able to penetrate the tree canopy and provide precise geometric information about the vegetation structure for different strata. Some works have combined LiDAR 3D point clouds with aerial images (Ke et al., 2010), forest-centric GIS (Diedershagen et al., 2004), or expert information on forest habitats (Latifi et al., 2015) in order to improve information extraction. Focusing on an operational and reproducible scenario, our approach operates on a common acquisition setting in which a LiDAR acquisition is combined with a simultaneous multi-spectral very high resolution optical acquisition. This leads to the generation of a 3D point cloud attributed with both geometric and radiometric information.

Traditional Approaches to Vegetation Structure Analysis. In order to exploit the rich structural information of aerial LiDAR scans, researchers have developed two main approaches. The *Area-based* approach consists in deriving handcrafted descriptors from 3D acquisitions and regressing vegetation features for a subset of acquisition (Bouvier et al., 2015; Latifi et al., 2015). Such prediction models have been adopted for operational purposes since they require lower point densities and are computationally efficient. However, they require a sufficient amount of ground-based measurements for establishing reliable models (Yu et al., 2010). The *Tree-based* method first delineates individual trees and in turn aggregates morphological indicators across the area of interest. Such methods typically start with a non-parametric detection method (Hamraz et al., 2016) and then use clustering algorithms like watershed (Chen et al., 2006), region growing (Hyyppa et al., 2001), or graph-based methods (Reitberger et al., 2009; Strîmbu and Strîmbu, 2015). A limitation of these segmentation task it tends to overlook small subdominant trees and focus on larger trees, limiting subsequent stratum analysis. As our method directly operates on a plot and does not require any pre-segmentation step, it qualifies as *Area-based*. However, we produce rasterized occupancy maps whose pixel size is typically sub-metric.

Learning-Based Vegetation Structure Analysis. In order to achieve high generality without requiring expert knowledge on the vegetation structure of the considered area, several works have explored the benefit of learning-based approaches for automated forest analysis, see the review of Liu et al. (2018). More recently, the first deep learning methods operating on 3D forestry data have been proposed. Lang et al. (2021) investigate the possibility of achieving global-scale mapping with satellite-borne LiDAR and Bayesian deep learning. Several approaches based on 2D convolutional networks have been proposed for the classification of individually segmented trees (Hamraz et al., 2018; Chen et al., 2021; Seidel et al., 2021; Zou et al., 2017). However, using networks designed for 2D to analyse 3D data not only incurs costly preprocessing steps, but generally leads to lower performance than using architectures dedicated to 3D data (Guo et al., 2020). Furthermore, these methods require databases with precisely segmented trees, which makes them less applicable to the operational setting of vegetation structure prediction. This is particularly problematic for natural forests for which trees' canopies often intersect, making segmentation a difficult and sometimes dubious process.

3D Deep Learning. The main difficulty in analysing 3D point clouds with deep learning is their irregular structure and varying sampling density. This has been alleviated by relying on images (Boulch et al., 2018; Su et al., 2015; Wei et al., 2020), 3D regular grids (Graham et al., 2018; Choy et al., 2019; Maturana and Scherer, 2015; Riegler et al., 2017), graphs (Simonovsky and Komodakis, 2017; Landrieu and Simonovsky, 2018), or continuous-space convolutions (Boulch, 2020; Thomas et al., 2019). A simpler class of algorithms considers point clouds as unordered sets of points (Ruizhongtai Qi et al., 2017; Charles et al., 2017; Zaheer et al., 2017), and does not need any of the preprocessing steps required by the aforementioned methods. Since we consider the strata occupancy prediction problem for plots individually, and with the objective of scalability and computational efficiency, we choose the straightforward PointNet network (Charles et al., 2017).

Weakly Supervised Learning. Training deep networks typically requires a large training database. However, manually producing dense annotations of LiDAR 3D point clouds of vegetation is a laborious task, often made even more complicated by visual ambiguities (Milberg et al., 2008). Furthermore, since most monitoring tasks, such as fire management (Price and Gordon, 2016; Stefanidou et al., 2020), stratum analysis (Martinuzzi et al., 2009), or national forest inventory (Hauglin et al., 2021) do not typically operate at the tree-level, such a degree of detail is unnecessary in practice. Similarly to what Tong et al. (2021) developed for images, our approach only requires parsimonious annotations to be trained: our network can be entirely supervised from a single aggregated occupancy per cylindrical plot and per stratum. Such values can be obtained by an operator estimating visually the stratum occupancy of their immediate surrounding area. While this requires an *in situ* intervention, the annotation task is less tedious than annotating individual points. As commonly encountered in weakly supervised schemes (Ratner et al., 2019; Liu et al., 2019), our method requires regularization terms for more realistic outputs.

Our method, developed for pasture land management, can be used for applications requiring multi-strata vegetation analysis such as forest fire monitoring, habitat analysis, or forest inventory. The key contributions of this paper are as follows:

- We show that a simple deep network can produce two-dimensional stratum maps using only plot-aggregated weak annotations.
- We propose regularization terms improving the realism and generality of the predicted occupancy maps without requiring expert knowledge.
- We introduce an open-access dataset of multi-spectral 3D point clouds corresponding to plots of agricultural parcels along with aggregated stratum occupancy annotations.

2. Materials and methods

2.1. Dataset Description

In this subsection, we present the new proposed dataset and give further precision on the task of automated stratum occupancy prediction.

Dataset Composition. We gathered a total of 199 aerial LiDAR scans of cylindrical plots with a 10 m radius with an average 10-pulse per square meter density. The plots have been selected by forestry experts, and correspond to typical pasture land parcels in South-Eastern France (see Fig. 1). A RGB camera sensor captures Red-Green-Blue radiometric information simultaneously with the LiDAR acquisition.

Each plot comprises between 3000 to 17000 3D points, and each point is attributed with a total of 9 features: (i) absolute 3D coordinates in Lambert-93 system, (ii) RGB reflectance values acquired by the aerial camera, (iii) Near-InfraRed reflectance values, uncalibrated laser intensity and return number as provided by the LiDAR device.

In order to align the 3D point clouds and the camera acquisition, we project the pixels' colors to the corresponding 3D points without taking

(a) LiDAR scan. (b) Aerial image. (c) *In situ* view.

Fig. 1. Examples of plot-based acquisitions. The two point clouds in (a) correspond to two distinct plots. They are colored using the aerial images in (b) (the plots are represented by the circles). A human annotator visually assesses their surroundings (c) and estimates the occupancy ratio of the lower, medium, and higher vegetation stratum. In both represented plots, the occupancy ratio of the lower stratum is 50%, while the occupancy higher strata differs.

occlusion into account. Since both 2D and 3D acquisitions are georeferenced, we simply compute a point-to-pixel mapping using the x and y coordinates and color the points according to their neighboring pixels. Note that this visibility model is naive, and ignore potential occlusion due to the canopy. However, our ablation study shows that even such simple treatment improves the precision. In addition, to the best of our knowledge, we are not aware of any efficient technique able to color the point cloud for the under-canopy layers from overhead imagery.

Since there is no theoretical framework proposing intensity calibration over forest plots, we decided to keep the raw values. We could also have added the total number of returns for each LiDAR ray, but observed that this information is in practice redundant with the return number.

Normalisation. The x and y values of the points in each plot are normalized within the unit square $[-1,1]^2$. The z-value—or height—of each point is normalized locally by subtracting the z-value of the lowest point in a 0.5m cylindrical neighborhood. This simple approach allows to compensate for irregular terrain and avoid the propagation of potential errors in the Digital Terrain Model (Mallet et al., 2011).

Annotation. Each plot has been annotated by a human expert *in situ*, as illustrated in Fig. 1. The annotation describes the occupancy of different strata of the cylindrical plot as assessed visually by the human annotator. We rely on their experience and expertise to be able to disambiguate overlap between vegetation strata. However, this measure remains, of course, subjective and sometimes ambiguous.

More precisely, we are provided with the lower stratum occupancy ratio $\widehat{o}_L$, the medium vegetation stratum occupancy ($\widehat{o}_M$), and the higher strata occupancy ($\widehat{o}_H$), see Fig. 2. The occupancy value $\widehat{o}_L$ characterizes the proportion of the ground surface occupied by grass or low vegetation, as opposed to stone, soil, or sand. $\widehat{o}_M$ characterizes the proportion of the surface of the plot occupied by the footprint of medium vegetation, *ie*, with a height between 0.5 and 1.5m.

This type of vegetation, typically bush-like, is the most accessible by pasture animals and represents an important indicator for land-use monitoring agencies. Note that the trunks of trees exceeding 1.5m do not contribute to this coverage. Finally, the canopy occupancy is defined as the ratio of the plot surface occupied by the footprint of the canopy of trees over 1.5m.

Our dataset (Kalinicheva, 2021) is publicly available at https://doi.org/10.5281/zenodo.5555758.

2.2. Weakly-Supervized Stratum Prediction

We consider a point cloud $X \in \mathbf{R}^{N\times 9}$ with N points. Each point is characterized by the 9 radiometric and geometric features described in Section 2.1. Our objective is to predict rasterized vegetation occupancy maps of the three vegetation strata (Fig. 4): lower (o_L), medium (o_M), and higher (o_H). We first predict a semantic class for each point (Section 2.2.1), then aggregate these predictions into explicit rasterized stratum occupancy maps (Section 2.2.2). To improve our weakly-supervised model, we use some regularization terms. We introduce in Section 2.2.3 an unsupervised elevation model encouraging more coherent classification. In Section 2.2.4, we present our entropy–based prior designed to produce crisper maps. All can be incorporated into a global loss function presented in Section 2.2.5.

2.2.1. Point-wise Class Prediction

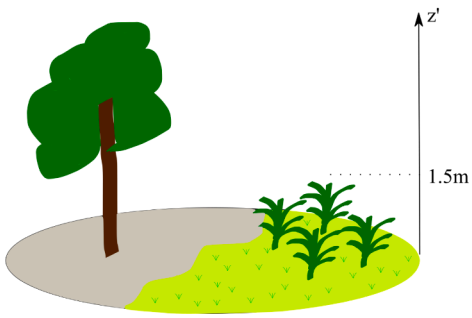
We classify each point among $C = 4$ classes: lower vegetation, bare soil, medium vegetation, and higher vegetation. For this task, we use the straightforward PointNet semantic segmentation network (Charles et al., 2017), see Appendix A. While more recent networks such as Minkowski Engine (Choy et al., 2019) or KPConv (Thomas et al., 2019) have even higher accuracy, they typically require considerably more annotations. As we only have access to plot-aggregated annotations (under 600 values in total), these complex methods may rapidly suffer from overfitting. Furthermore, these networks are used to learn intricate spatial relationship between individual points, while our problem is well captured by plot-level point distribution, which PointNet is well-suited to capture.

To handle the varying density of point clouds and to facilitate batch-training, we first sample each plot's point cloud X into the same number M of points. The sampled points are classified by a PointNet network, and in turn interpolated to the initial full size point cloud with nearest neighbor interpolation, see Fig. 3.

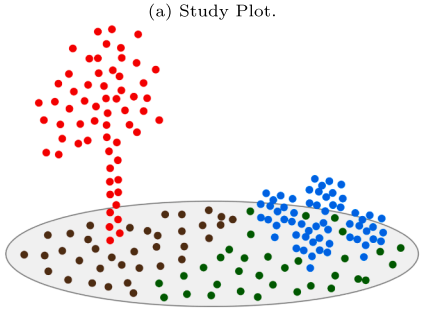
We denote the predicted probabilities for a point $n \in [1, N]$ as follows: ($Y_{n,\mathrm{BS}}$) for bare soil, ($Y_{n,L}$) for low, ($Y_{n,M}$) for medium and ($Y_{n,H}$) for high vegetation, respectively.

2.2.2. Stratum Modeling

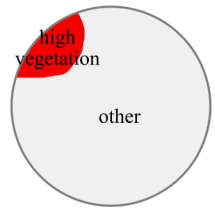
In order to obtain stratum occupancy maps and predictions, we project the point cloud onto the stratum rasters and aggregate the function into a ratio prediction which we can supervise end-to-end.



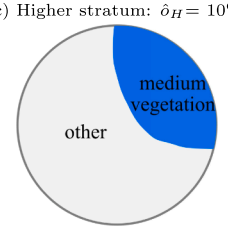
(a) Study Plot.



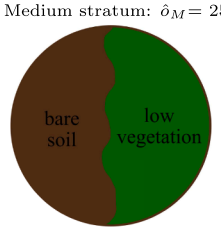
(b) Pointwise labelling:
soil, grass, medium vegetation, high vegetation.



(c) Higher stratum: $\hat{o}_H = 10\%$.



(d) Medium stratum: $\hat{o}_M = 25\%$.



(e) Lower stratum: $\hat{o}_L = 50\%$.

Fig. 2. Occupancy-annotated plot. In (a), we represent a synthetic scene whose point labels are represented in (b). The occupancy of the higher, medium, and lower strata are estimated by *in situ* observers. Our goal is to retrieve occupancy maps (c), (d) and (e) for each stratum based on these plot-aggregated annotations. Here, the lower stratum is covered in equal proportions by low vegetation and bare soil, while the medium stratum is occupied at 25% by bushes and the higher stratum is occupied at 10% by the tree crown. Note that the lower part of the tree trunk represented in red in (b) is not counted as medium vegetation despite being under 1.5m.

Point Projection. We use point-wise prediction to estimate the occupancy of pixels of each stratum. We consider a raster of $K \times K$ pixels aligned with the projection of the cylindrical plot on the horizontal axes. We associate each pixel (i,j) of the raster with the set of 3D points $\text{proj}(i,j) \subset [1,\cdots,N]$ whose vertical projections *fall* in the pixel's extent:

$$\text{proj}(i,j) = \left\{ n \in [1,N] \middle| \left\lfloor \frac{x_n}{K} \right\rfloor = i, \left\lfloor \frac{y_n}{K} \right\rfloor = j \right\}, \tag{1}$$

with x_n, y_n the x and y coordinates of a point n. Note that the cylindrical shape of the plot implies that only pixels within a disk inscribed in the raster will be associated with any point, see Fig. 5.

Stratum Aggregation. We compute an occupancy $O_{i,j}^{(\text{stratum})}$ for each pixel $i,j \in [1,K]^2$ and each stratum by taking the highest predicted probability for all points associated with this pixel. Finally, we aggregate the pixel projection stratum-wise to obtain a single prediction for each plot and stratum. For each stratum in $\{L,M,H\}$ we have:

$$O_{i,j}^{(\text{stratum})} = \max_{n \in \text{proj}(i,j)} Y_{n,\text{stratum}} \tag{2}$$

$$o_{\text{stratum}} = \frac{1}{D} \sum_{i,j=1\cdots k} O_{i,j}^{(\text{stratum})} \tag{3}$$

with D the number of pixels within the disk obtained when projecting the cylindrical scan ($D \sim \pi/4K^2$). Note that this projection raster allows us to visualize pixel-precise predicted occupancy maps for each stratum.

Occupancy Supervision. We supervise the predicted stratum occupancies with the ground truth annotated occupancies $\hat{o}_L, \hat{o}_M, \hat{o}_H$ using the distance function ϕ:

$$\mathscr{L}_{data} = \phi(o_L - \hat{o}_L) + \phi(o_M - \hat{o}_M) + \phi(o_H - \hat{o}_H). \tag{4}$$

In practice, we use $\phi(x) = \sqrt{x^2 + 10^{-4}}$ as a differentiable surrogate of the ℓ_1 norm.

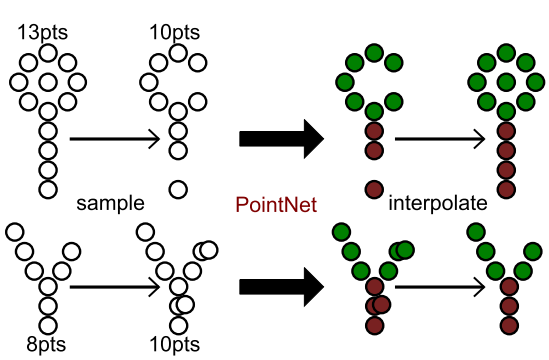


Fig. 3. Point Sampling. Each point cloud is sampled to the same size M. When $N < M$, we duplicate randomly chosen points to get the desired point count. Then, the sampled points are classified with a PointNet network. Finally, the predictions are interpolated to the full size of the initial point cloud.

2.2.3. Elevation Modeling

The model described above does not explicitly model the distribution of elevations within each stratum. In theory, points several meters above the ground can contribute to the lower vegetation stratum as long as the stratum-wise aggregated values are in agreement with the ground truth. We propose to explicitly model the elevation of points within each stratum in an unsupervised way with the goal of making the occupancy maps more realistic, and to increase the generalizability of the models.

By plotting the elevation histograms of all points as seen in Fig. 6, we observe that this empirical distribution follows a mixture model of two Gamma distributions. Moreover, we can easily interpret its components: the low elevation density peak corresponds to bare soil and low vegetation, while the long-tailed high elevation distribution corresponds to medium and high vegetation. To simplify the problem, we group the three strata into two groups: ground G and non-ground NG (medium and high vegetation).

We can estimate the parameters $\{\alpha_i, \beta_i\}_{i \in \text{G,NG}} \in \mathbf{R}^2$ of both Gamma distributions as well as the mixture parameters $\{\rho_G, \rho_{NG}\} \in [0,1]^2$ with the expectation–conditional–maximization (ECM) algorithm (Young

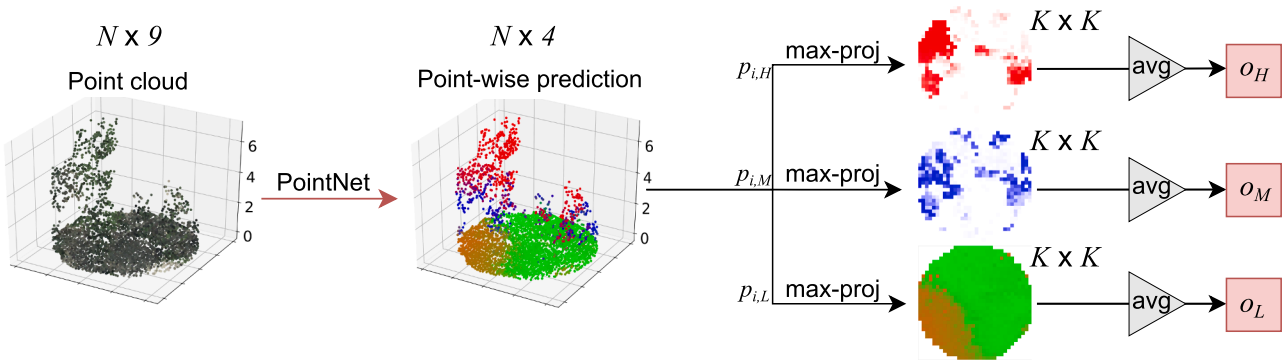


Fig. 4. Pipeline. Our network performs the semantic segmentation of a 3D point cloud within four different classes. The resulting probabilities are projected onto rasters corresponding to different strata. Finally, the occupancy maps are aggregated into the stratum vegetation ratio.

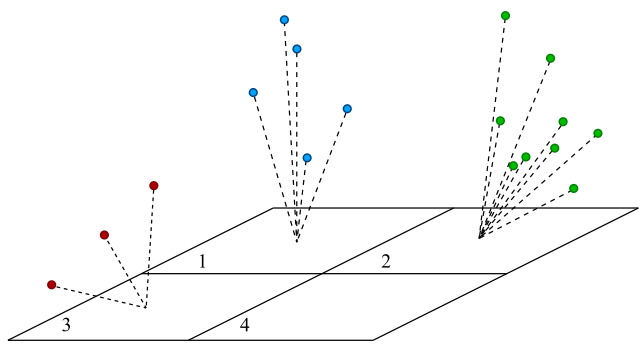


Fig. 5. Projection onto Stratum Rasters. Example of 3D data projection to the 2D space. A different number of points can be projected to one pixel, or possibly none (pixel 4) depending on point density of the area.

et al., 2019) detailed in Algorithm 1. This procedure is entirely unsupervised but requires a meaningful initialization of mixture parameters, which can be achieved by trial-and-error guided by the resulting likelihood value. The ECM algorithm and its inner Newton-Rachson optimization Cheney and Kincaid (2012, Chapter 3.2) converges in only a few iterations, and this step is only required to be performed once per dataset. In Fig. 6, we represent the mixture of Gamma distributions

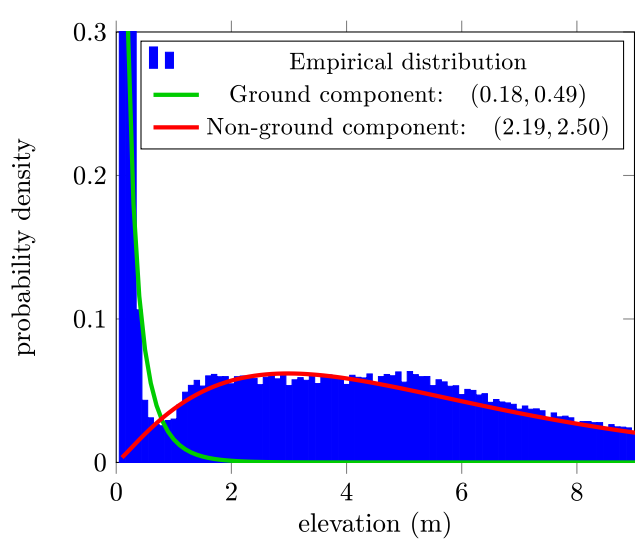


Fig. 6. Elevation Modeling. Empirical elevation distribution (blue) and two components of a fitted mixture of Gamma distributions with weight 0.55 and 0.45. The green component models the elevation of ground and low vegetation, while the red component models the medium and high vegetation.

obtained with the ECM algorithm. We denote by Γ_G and Γ_{NG} the distributions whose parameters (scale and shape) are learnt with the ECM algorithm.

Algorithm 1. ECM Algorithm for Gamma Mixture estimation.

ψ denotes the digamma function (Abramowitz and Stegun, 1948)

Require: Input: elevations $z \in \mathbf{R}_+^N$
Require: $\alpha_i, \beta_i, \rho_i \leftarrow$ manual initialization for $i \in \{G, NG\}$
  while not converged **do**
    for $n \in [1, N], i \in \{G, NG\}$ **do** ▷E-step
      $\gamma_n^i = \rho_i \Gamma(z_n; \alpha_i, \beta_i)$
      $e_n^i = \gamma_n^i / (\sum_{j \in G,NG} \gamma_n^j)$ ▷Expectation that point n is in group i
    end for
    for $i \in \{G, NG\}$ **do** ▷Conditional M-steps
      $\rho_i \leftarrow \frac{1}{N} \sum_{n=1}^N e_n^i$ ▷Mixture parameter
      $\alpha_i \leftarrow$ root of $\sum_{n=1}^N \gamma_i^n [\log(z_n) + \log(\beta_i) - \psi(\alpha_i)]$ ▷With Netwton-Rachson
      $\beta_i \leftarrow N \alpha_i \rho_i / \sum_{n=1}^N z_n e_n^i$
    end for
  end while

As described in the Appendix C, the Bayesian theorem allows us to define loss $\mathscr{L}_{\text{elevation}}$ as the negative log-likelihood of the observed elevations $Z \in \mathbf{R}_+^N$ conditionally to the observations $X \in \mathbf{R}^{N \times 9}$:

$$\mathscr{L}_{\text{elevation}} = -\sum_{n=1}^{N} \log((Y_L + Y_{BS})\Gamma_G(z_n) + (Y_M + Y_H)\Gamma_{NG}(z_n)). \tag{5}$$

This function encourages the network to classify points with low elevation as ground or low vegetation, and points with high elevation as medium or high vegetation. While this could arguably be done by setting a manual threshold or hand-picked parameters, this method is adaptive to each new dataset and converges quickly.

2.2.4. Occupancy Prior Modeling

Pixel values of the vegetation occupancy maps take continuous values from 0: no vegetation, to 1: completely covered by vegetation. This value can also be influenced by the confidence of the point classifier: an ambiguous pixel with low confidence may be predicted at 0.5 occupancy. Once aggregated plot-wise, such indecisive predictions may average to the correct prediction and lead to a low loss. However, the resulting stratum occupancy maps may become fuzzy and hard-to-interpret. We would like to reserve values such as 0.5 for the rare case of pixels which are partially covered by a given vegetation structure.

In order to discourage the network to express its uncertainty through intermediate prediction, we propose to regularize our loss with the average entropy of the pixel prediction of all maps:

$$\mathscr{L}_{\text{entropy}} = -\frac{1}{D}\sum_{\substack{\text{stratum}\\ \in\{L,M,H\}}}\sum_{(i,j)\in[1,K]^2} H\left(O_{i,j}^{(\text{stratum})}\right), \tag{6}$$

with $H(\cdot)$ the function returning the entropy of an input distribution, and D the number of pixels in the cylinder projection onto the raster.

2.2.5. Global Loss

We add the elevation loss $\mathscr{L}_{\text{elevation}}$ and entropy loss $\mathscr{L}_{\text{entropy}}$ as regularizers of the data loss $\mathscr{L}_{\text{data}}$. The resulting loss for model optimization is computed batch-wise and averaged over the batch's plots:

$$\mathscr{L} = \mathscr{L}_{\text{data}} + \lambda\mathscr{L}_{\text{elevation}} + \mu\mathscr{L}_{\text{entropy}}, \tag{7}$$

with $\lambda = 1$ and $\mu = 0.2$ the respective regularization strengths of $\mathscr{L}_{\text{elevation}}$ and $\mathscr{L}_{\text{entropy}}$.

2.3. Implementations Details

Our entire pipeline is implemented in PyTorch 1.7 and CUDA 10.2. Our network is trained with a batch size of 20 plots for 100 epochs, and we use the ADAM optimizer(Kingma and Ba, 2015) with a learning rate of 0.001 divided by 10 after 50 epochs, and all other default parameters. We add a dropout layer (Srivastava et al., 2014) with probability 0.4 before the last layer to increase the model's robustness. Our network can be trained in under 24 min on an NVIDIA GeForce RTX 3060 GPU and a Xeon W-2123 CPU with 64 GB of RAM. Our python impementation of the ECM algorithm converges in under 5 s on a standard workstation for over 500 000 points.

During both training and inference, we sample a fixed number of $M = 4096$ points for each plot, and duplicate points for plots with fewer points. This allows us to use efficient batch-parallel computing, as well as adding sampling stochasticity to decrease overfitting. The efficient organization of data into batches plays a crucial role in both inference speed and the convergence of the optimization step. The size of the stratum raster is set to $K = 32$ pixels.

The layer sizes of three MultiLayer Perceptron (MLP) blocks of our PointNet model are respectively: [32,32], [64,128] and [64,32,4], see Appendix A.

2.4. Experimental Setting

We perform 5-fold cross-validation on the dataset presented in Section 2.1, which is composed of $T = 199$ cylindrical plots. We report the mean absolute occupancy error e_L, e_M, e_H between the predicted and true occupancy for each stratum, as well as the inter-stratum macro-average e:

$$e_K = \frac{1}{T}\sum_{n=1}^{T}\left|\hat{o}_n^K - o_n^K\right| \text{ for } K \in \{L, M, H\} \tag{8}$$

$$e = \frac{1}{3}(e_L + e_M + e_H). \tag{9}$$

Our code is available at https://github.com/ekalinicheva/plot_vegetation_coverage.

3. Results and discussions

In this section, we present an experimental evaluation of the performance of our approach, compared to four baselines relying on handcrafted features, linear and Random Forest (RF) regressions, and a simple deep network.

3.1. Competing Approaches

We propose four baselines to better assess the performance of our method: a classic tree-based algorithm operating on handcrafted features, linear regression and Random Forest regressions, and a simple deep learning-based regression. All learning-based models are trained and evaluated using 5 folds cross-validation.

Handcrafted Approach. This method relies on handcrafted descriptors and is composed of several steps:

- **Lower Stratum Occupancy.** We first consider all points with a normalized elevation under 0.5m. We then select the points of all plots with 0% lower stratum vegetation occupancy and average the 6 non-geometric normalized point features to form a prototypical *bare soil* point. Likewise, we form a prototypical *low vegetation* point. We then classify all points according to their Euclidean distance with respect to the bare soil and low vegetation prototypes.
  Finally, all points with an elevation below 0.5m are projected onto the raster corresponding to the lower stratum, and the pixels are classified as bare soil or low vegetation by a majority vote. This allows us to compute a predicted occupancy for the lower stratum.
- **Medium and Higher Stratum Occupancy.** The points with an elevation between 0.5 and 1.5m are classified as medium vegetation, and the remaining points as high vegetation. The points are then projected onto the raster corresponding to their predicted stratum, and the raster's pixels are classified as vegetation if it contains the projection at least one such point.

Linear Regression Model. We present a baseline in the spirit of regression models (Latifi et al., 2015). For each plot, we divide the points into 3 sets based on their height: 0-0.5m, 0.5-1.5m and >1.5m. We fit a regression model for each layer predicting the aggregated values from 10 input features: mean z value, standard deviation of z value, mean of radiometric values (RGB, NIR), mean intensity, mean point density, mean return number, mean of ratio of return number and number of returns.

Random Forest Regression Model. The random forest regression model is often used to derive the vegetation parameters (Venier et al., 2019). Our Random Forest Regression model is built using the same techniques as the linear regression. We set the maximum tree depth parameter at 4 and 3 features per split.

Deep Learning Baseline. We also train a simple PointNet network to directly predict the three strata occupancy values directly. The network then follows the same training procedure than our proposed approach. See Appendix B for more details about this method.

3.2. Results

In Table 1, we present the quantitative performance of the different methods evaluated. Our method outperforms both regression models and the simple deep learning baseline at the cost of added computation time due to the point-wise nature of our prediction and the additional steps, such as elevation modeling for stratum-wise point projection. In practice, our method provides further improvements beyond precision: (i) the occupancy maps can be easily visualized for each stratum in raster form; (ii) the point heights are explicitly modeled, ensuring the consistency of our predictions.

In Fig. 7, we present qualitative results. Despite the absence of ground truth to evaluate the quality of the predicted occupancy maps, one can see the visual correspondence between the point clouds and their corresponding strata coverage.

In Table 2, we evaluate the quantitative impact of some of our main design choices and report quantitative results in Table 2 and qualitative results in Fig. 8.

We highlight the importance of elevation modeling and entropy penalization by presenting the performance without $\mathscr{L}_{\text{elevation}}$ (No elevation modeling), without $\mathscr{L}_{\text{entropy}}$ (No entropy penalization), and with neither (No el. modeling & ent. pen.). We observe that both penalizations have little effect on the performance. This is an expected results,

Table 1
Quantitative results. We report the accuracy of the predicted aggregated plot occupancy, along with the inference speed in number of plots per second.

Method	Absolute error, %				Inference Time
	low	medium	high	average	plots/s
Handcrafted	21.9	20.7	10.3	17.6	20
Linear Regression	19.7	**11.6**	9.8	13.7	**3500**
RF Regression	18.3	12.4	9.5	13.4	2700
PointNet Baseline	17.4	13.5	7.7	12.8	400
Ours	**15.5**	13.6	**7.5**	**12.2**	125

as $\mathscr{L}_{\text{data}}$ directly represents the error rate. However, as illustrated in Fig. 8, these regularizations play a crucial role in obtaining realistic occupancy maps. Without elevation modeling, the localization of high occupancy pixels is decorrelated with the actual position of vegetation. In the absence of entropy minimization, the maps become fuzzy and lack sharp features. However, if the regularization parameter μ of the entropy is set too high, *eg*, $\mu = 0.5$ instead of 0.2, our algorithm will predict uniform occupancy maps of either 0s or 1s. As seen in Table 2, this results in lower prediction accuracy. Regarding the weight of the elevation modeling loss, we did not observe a significant difference in prediction between $\lambda = 1$ and $\lambda = 0.5$. However, removing this loss altogether has a negative impact on the quality of the produced results.

We study the influence of the raster size K with $K = 16$ (Coarser raster) and $K = 64$ (Finer raster) instead of the chosen $K = 32$. We can see that the results are quantitatively worse with these choices of K. Lower values for K produce less informative occupancy maps, while higher values lead to many empty pixels as the number of points is not sufficient to propagate the occupancy.

Finally, we evaluate the impact of non-geometric features by removing the number of returns and LiDAR intensity (No geometric features), and by removing RGB and NIR (No radiometric features). Radiometric features prove to be useful to distinguish between low vegetation from bare soil, or leaves from tree branches. However, the benefits of these features vary from one plot to the other due to the acquisition geometry: the RGB radiometry is acquired as an optical image and superposed on the 3D point cloud with NIR band. Occlusions can prevent the precise colorization of points in lower stratum, see Fig. 9. Geometric features, on the other hand, are not affected by the plot geometry. However, they contain less discriminative information. In both cases, using only one set of features slightly decreases the overall model accuracy and often produces visually incorrect lower and sometimes medium stratum occupancy maps.

Limitations While we can assess the precision of our approach in terms of aggregated predictions, we are not able to evaluate quantitatively the pixel-wise or point-wise prediction. Indeed, we do not have access to these costly annotations. Nevertheless, some plots can be easily

Table 2
Quantitative Ablation Study. Impact of some design choice on the mean absolute error.

Method	Absolute error, %			
	low	medium	high	average
Our method	**15.5**	**13.6**	**7.5**	**12.2**
No elevation modeling	16.6	13.8	7.4	12.6
No entropy penalisation	15.6	13.9	7.3	12.3
Higher entropy penalisation, $\mu = 0.5$	18.4	23.6	11.6	17.9
No el. modeling& ent. pen.	15.8	13.4	6.5	**11.9**
Coarser raster	16.7	14.4	8.3	13.1
Finer raster	15.9	18.5	7.1	14.0
No geometric features	16.8	13.4	7.6	12.5
No radiometric features	16.5	13.4	7.7	12.5

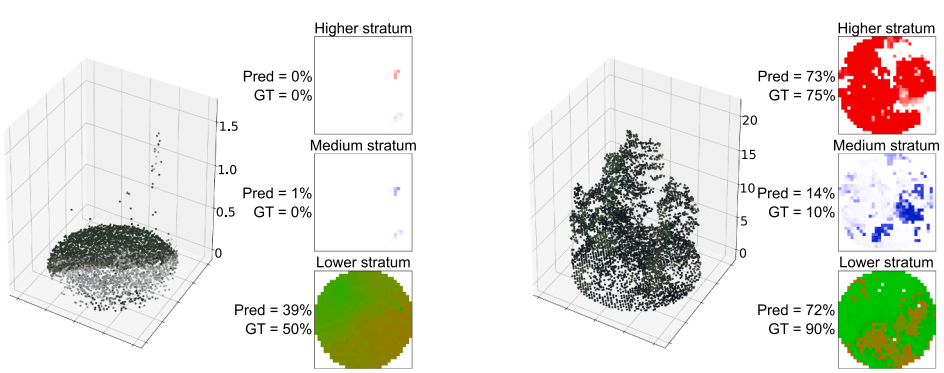
visually validated. For example, Fig. 9 provides an example of a plot without medium or higher vegetation, and a lower stratum evenly divided into lower vegetation and bare soil. Even though none of the aggregated predictions o_L are exact, we observe that our model produces an occupancy map with the best visual fidelity to the plot. Hence, our model is able to combine all available features for better occupancy map prediction.

The global loss function used to train the model is composed of three different losses, which implies some manual parameter tuning. However, as the weights of the networks are learned from data, our method does not rely on handcrafted parameters, which typically require setting numerous parameters by hand. As our data only originates from one region, we are not able to evaluate the robustness of our approach. However, since our method is very generic in its formulation, it should be able to be trained on data from another area and not require manual tuning.

Finally, the ECM algorithm for elevation modelling requires a manual initialization step. However, the parameters of the Gamma distributions are intuitive as they relate to the moment of the distribution (mean height, deviation) and can be approximated by a knowledgeable operator. Alternatively, this can be hand-tuned using the likelihood as a guide, and only needs to be approximately tuned before running the optimization.

4. Conclusion

In this paper, we presented a 3D deep learning method for predicting occupancy across three vegetation strata: lower, medium, and higher. Contrary to traditional regression models with comparable accuracy, our model is able to perform a point-wise classification using only three aggregated values per example plot. Furthermore, projecting the point prediction produces vegetation occupancy rasters with a high precision and at a small computational cost. Our code is released in open access along with the first



(a) Simple plot without vegetation. (b) Complex, multi-strate plot.

Fig. 7. Qualitative Results. Our method predicts aggregated stratum occupancy along with the corresponding rasterized occupancy maps. Here, the pixels are colored according to the value of the predicted occupancy: shades of green, blue, and red indicate pixels with high-predicted vegetation coverage for the lower, medium, and higher strata respectively, while brown corresponds to bare soil.

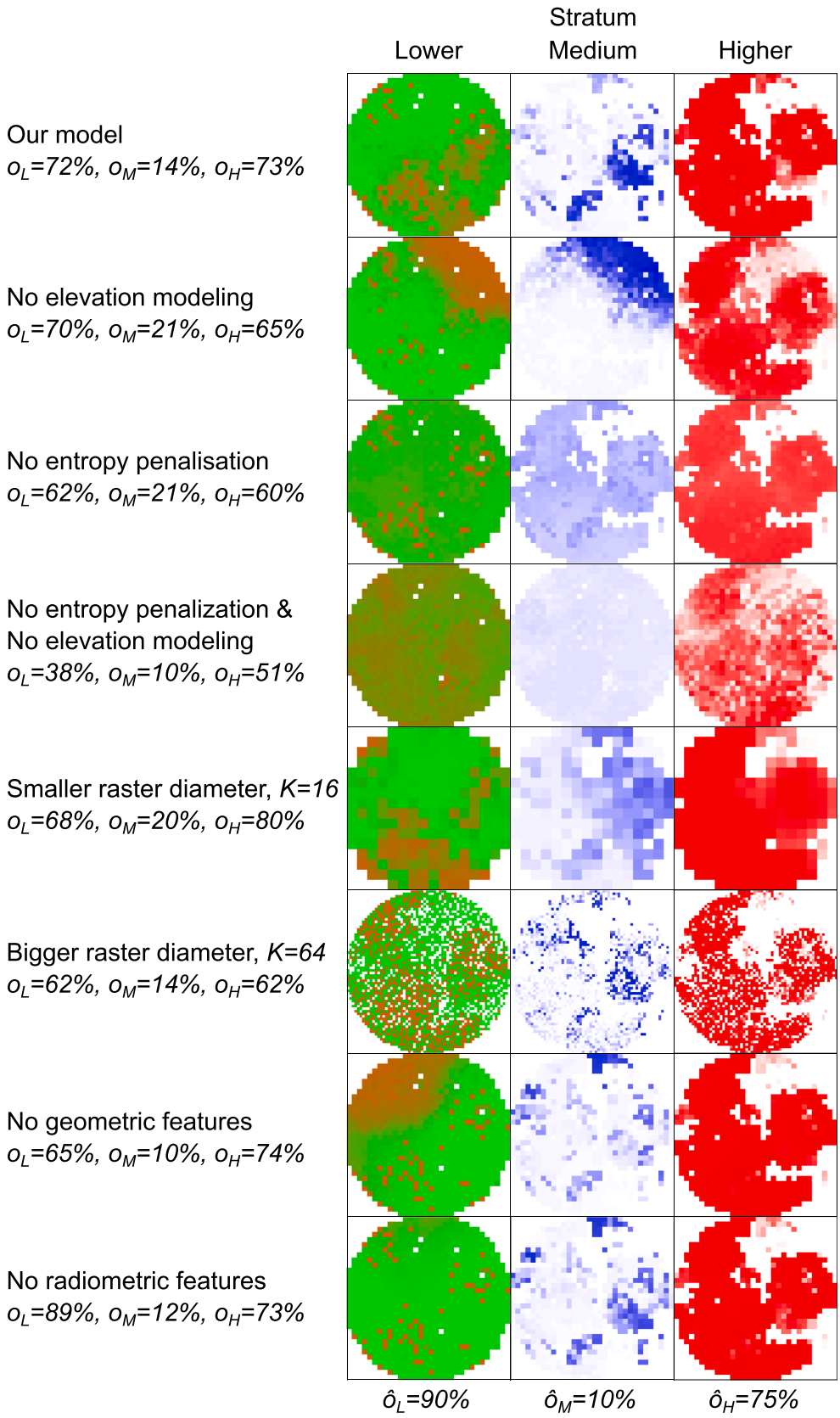


Fig. 8. Qualitative Ablation Study. Occupancy maps produced by variations of our model, for the original point cloud presented in Fig. 7b. The accuracy of lower and medium stratum occupancy maps directly depends on the model configuration. However, the higher stratum occupancy map is almost identical for most models, as the higher vegetation is easiest. to distinguish.

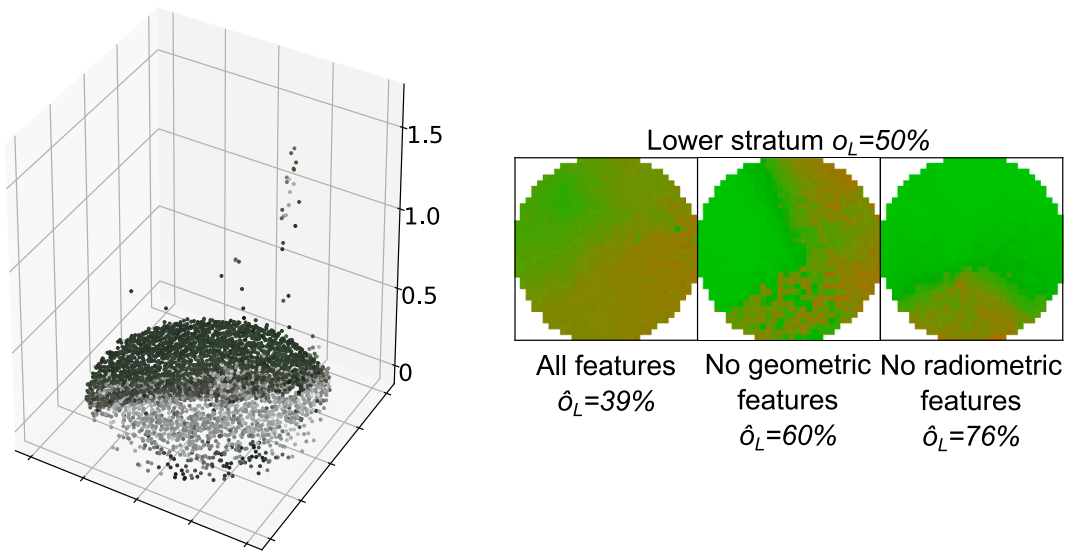


Fig. 9. Illustrative Example. Lower stratum occupancy maps produced by our model using different sets of features for the plot in 7a. We observe that our original model produced a more accurate occupancy map, confirming the importance of usi.ng all features.

forestry dataset with plot-based occupancy annotations.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This study has been co-funded by CNES (TOSCA FRISBEE Project, convention n°200769/00) and CONFETTI Project (Nouvelle Aquitaine Region project, France).

Appendix A. PointNet Model Baseline

Our algorithm is based on PointNet model for semantic segmentation (Charles et al., 2017). Given a point cloud $X \in \mathbf{R}^{N \times D_0}$, where D_0 is the number of point features, the model predict the scores $P = \{p_0, \ldots, p_N\} \in \mathbf{R}^{N \times C}$ that a point in X belongs to each of C classes.

The model process the data in the following manner:

- To homogenise the input data, we first subsample each point cloud into M points (10).
- The first MultiLayer Perceptron (MLP) $\mathrm{MLP}_1 : D_0 \mapsto D_1$ is applied to each point $i = 1 \cdots M$ in parallel and maps raw point features to a learned point descriptors f_i^1 of size D_1. This MLP is composed of a sequence of 1D convolutional layers (Goodfellow et al., 2016), batch normalization (Ioffe and Szegedy, 2015), and Rectified Linear Units (ReLU) (Nair and Hinton, 2010) (11).
- $\mathrm{MLP}_2 : D_1 \mapsto D_2$ operates in the similar way (12).
- The maxpooling operation is applied over M points to extract a global shape descriptor (GSD) G of size $1 \times D_2$ for each point cloud (13).
- G is concatenated with the output of MLP_1 block, then each point is processed by $\mathrm{MLP}_3(D_1 + D_2) \mapsto C$ to extract the classes predictions as in previous MLP blocks. Note, that ReLU activation and BatchNorm are not applied to the last layer (14).
- Finally, we upsample the obtained predictions to N points by using the nearest neighbour algorithm (15).

The PointNet model for semantic segmentation is presented on Fig. 10 and can be summarized by the following equations:

$$M = \mathrm{sample}(M, N), \tag{10}$$

$$f_m^1 = \mathrm{MLP}_1(x_m), \ \forall m \in M, \tag{11}$$

$$f_m^2 = \mathrm{MLP}_2(f_m^1), \ \forall m \in M, \tag{12}$$

$$G = \mathrm{MAX}(F_2), \ F_2 = \{f_1^2, \ldots, f_M^2\}, \tag{13}$$

$$p_m = \mathrm{MLP}_3([f_m^1 || G]), \ \forall m \in M, \tag{14}$$

$$N = \mathrm{upsample}(N, M), \tag{15}$$

where $[||.]$ is the concatenation operator.

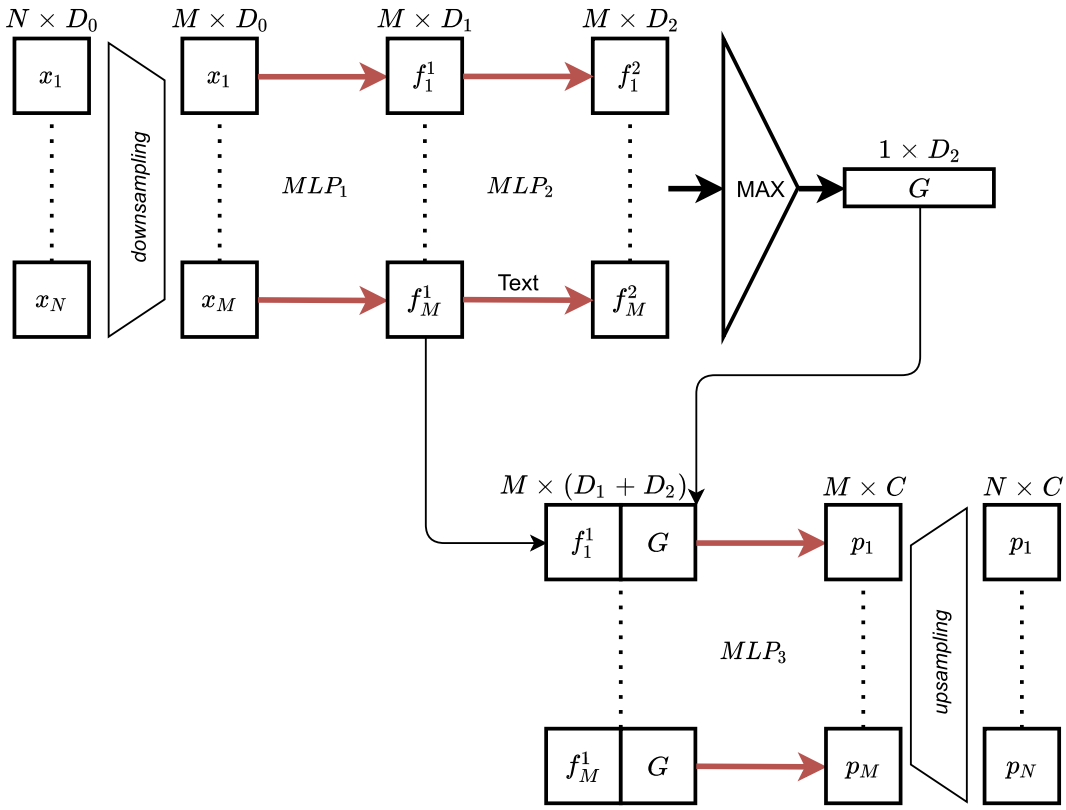


Fig. 10. PointNet model for Semantic Segmentation (Charles et al., 2017). The preprocessing step is consists in data subsampling from N to M points with $D_0 = 9$ features. MLP_1 and MLP_2 consecutively extract features F_1 and F_2 from 3D point cloud data, afterwards the maxpooling operator produces a global shape descriptor G. It is then concatenated with the first set of extracted features F_1, which are processed by MLP_3 to extract the final predictions P for C classes. In the postprocessing step, we upsample the point cloud to its original size of. N points.

Appendix B. PointNet Baseline

We compare our method to a simple deep learning baseline in which the PointNet network is used to directly predict the different layer occupancies from raw observations. We consider a point cloud $X \in \mathbf{R}^{N\times 9}$ with N points each characterized by the 9 radiometric and geometric point features described in Section 2.1.

Architecture. As in the original PointNet model, we first subsample the point cloud in M points that are then processed by a $\text{MLP}_1 : \mathbf{R}^9 \hookleftarrow D$ to extract D point features $f_i, i \in [1,\ldots,N]$. In the following step, we extract a GSD G. Finally, the second MLP $\text{MLP}_2 : \mathbf{R}^D \hookleftarrow 3$ maps the GSD to three values defining the predicted stratum occupancy o_L, o_M and o_H. Contrary to MLP_1, MLP_2 is composed of linear layers paired with ReLU activation for all the layers, except the last one that uses the Softmax activation (Nwankpa et al., 2018).

Loss Function. We use the same ℓ_1 norm loss function as for our model (4).

Implementations Details. The PointNet baseline is trained with a batch size of 20, the ADAM (Kingma and Ba, 2015) optimizer with a learning rate of 0.001 divided by 10 every 50 epochs, and default parameters. We add a dropout layer (Srivastava et al., 2014) with probability 0.4 before the last layer to prevent overfitting of the model. The output sizes of MLP blocks of the model are respectively: [32,32,64,128] and [64,32]. The number of subsampled points for each plot is $M = 2048$ (See Fig. 11).

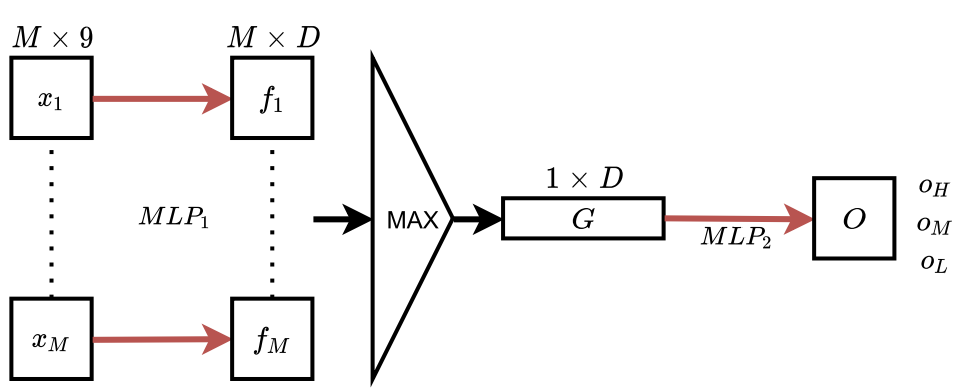


Fig. 11. Schematic view of our adapted PointNet model. The subsampled point cloud is processed by MLP_1 to extract a set of point features F, then the maxpooling operator produces a global shape descriptor G. Finally, MLP_2 is applied to G to extract the stratum occupancy values o_L, o_M, o_H.

Appendix C. Bayesian Elevation Modeling

We propose a probabilistic model linking the following random variables: the raw observation $X \in \mathbf{R}^{N\times 9}$, the ground/non-ground nature of each point $S \in \{\mathrm{G}, \mathrm{NG}\}^N$, and $Z \in \mathbf{R}_+^N$ the points' elevations. The conditional dependencies between these variables are represented in Fig. 12. We propose to consider S the ground/ non-ground variable as a discrete latent variable connecting the observation and the elevation.[1]

We can write the likelihood of the observed elevation $z \in \mathbf{R}_+^N$ conditionally to the observations according to this model:

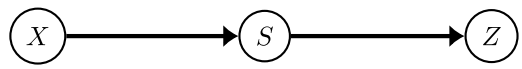


Fig. 12. Graphical model representation of the proposed probabilistic model linking the observation X, the ground/non-ground status of points S, and the elevation Z. By using a generative model for the elevation of a point conditionally to its stratum, we can quantify the compatibility between a stratum prediction and its corresponding elevation.

$$\ell(z) = \prod_{n=1}^{N} P(z_n|X) = \prod_{n=1}^{N} P(z_n, S = \mathrm{G}|X) + P(z_n, S = \mathrm{NG}|X). \tag{16}$$

After applying Bayes theorem, we obtain:

$$\ell(z) = \prod_{n=1}^{N} P(S = \mathrm{G}|X)P(z_n|X, S = \mathrm{G}) + P(S = \mathrm{NG}|X)P(z_n|X, S = \mathrm{NG}), \tag{17}$$

and using the conditional independence between Z and X with respect to S, we have that:

$$\ell(z) = \prod_{n=1}^{N} P(S = \mathrm{G}|X)P(z_n|S = \mathrm{G}) + P(S = \mathrm{NG}|X)P(z_n|S = \mathrm{NG}). \tag{18}$$

The conditional elevation distributions $P(z_n|S = \mathrm{G})$ and $P(z_n|S = \mathrm{G})$ are parameterized by Γ_{G} and Γ_{NG} respectively. The posterior probabilities belonging to the ground or nonground stratum are given by the network's prediction: $P(S = \mathrm{G}|X) = Y_L + Y_{\mathrm{BS}}$ and $P(S = \mathrm{NG}|X) = Y_M + Y_H$. Finally, we obtain that:

$$\ell((z) = \prod_{n=1}^{N} (Y_L + Y_{\mathrm{BS}})\Gamma_{\mathrm{G}}(z_n) + (Y_M + Y_H)\Gamma_{\mathrm{NG}}(z_n). \tag{19}$$

We define the loss $\ell_{\text{elevation}}$ as the negative log-likelihood of the observed elevations conditionally to the observations:

$$\mathscr{L}_{\text{elevation}} = -\log(\ell((z)) \ = -\sum_{n=1}^{N} \log((Y_L + Y_{BS})\Gamma_{\mathrm{G}}(z_n) + (Y_M + Y_H)\Gamma_{NG}(z_n)). \tag{20}$$

References

Abramowitz, M., Stegun, I.A., 1948. Handbook of mathematical functions with formulas, graphs, and mathematical tables.

Bergen, K., Goetz, S., Dubayah, R., Henebry, G., Hunsaker, C., Imhoff, M., Nelson, R., Parker, G., Radeloff, V., 2009. Remote sensing of vegetation 3-D structure for biodiversity and habitat: Review and implications for lidar and radar spaceborne missions. J. Geophys. Res.: Biogeosci.

Boulch, A., 2020. ConvPoint: Continuous convolutions for point cloud processing. Comput. Graph.

Boulch, A., Guerry, J., Le Saux, B., Audebert, N., 2018. SnapNet: 3D point cloud semantic labeling with 2D deep segmentation networks. Comput. Graph.

Bouvier, M., Durrieu, S., Fournier, R., Renaud, J.-P., 2015. Generalizing predictive models of forest inventory attributes using an area-based approach with airborne LiDAR data. Rem. Sens. Environ.

Campbell, M.J., Dennison, P.E., Hudak, A.T., Parham, L.M., Butler, B.W., 2018. Quantifying understory vegetation density using small-footprint airborne LiDAR. Remote Sens. Environ.

Charles, R.Q., Su, H., Kaichun, M., Guibas, L.J., 2017. PointNet: deep learning on point sets for 3D classification and segmentation. CVPR.

Chaton, T., Chaulet, N., Horache, S., Landrieu, L., 2020. Torch-Points3D: a modular multi-task framework for reproducible deep learning on 3D point clouds. 3DV.

Chen, Q., 2007. Airborne LiDAR data processing and information extraction. Photogram. Eng. Remote Sens.

Chen, Q., Baldocchi, D., Gong, P., Kelly, M., 2006. Isolating individual trees in a savanna woodland using small footprint LiDAR data. Photogram. Eng. Remote Sens.

Chen, X., Jiang, K., Zhu, Y., Wang, X., Yun, T., 2021. Individual tree crown segmentation directly from UAV-borne LiDAR data using the PointNet of deep learning. Forests.

Cheney, E.W., Kincaid, D.R., 2012. Numerical Mathematics and Computing. Cengage Learning.

Choy, C., Gwak, J., Savarese, S., 2019. 4D spatio-temporal convnets: Minkowski convolutional neural networks. CVPR.

Coops, N.C., Tompalski, P., Goodbody, T.R., Queinnec, M., Luther, J.E., Bolton, D.K., White, J.C., Wulder, M.A., van Lier, O.R., Hermosilla, T., 2021. Modelling lidar-derived estimates of forest attributes over space and time: a review of approaches and future trends. Rem. Sens. Environ.

Daubenmire, R., 1959. A canopy-coverage method of vegetation analysis. Northwest Sci.

[1] Note that since the elevation is already comprised in the observation, adding the arbitrary notion of stratum as an intermediate variable does not provide any rich insight in terms of modeling. It allows us however to link in a natural manner the stratum prediction and the elevation distribution.

Dechesne, C., Mallet, C., Le Bris, A., Gouet-Brunet, V., 2017. Semantic segmentation of forest stands of pure species combining airborne lidar data and very high resolution multispectral imagery. ISPRS J. Photogram. Rem. Sens.

Diedershagen, O., Koch, B., Weinacker, H., 2004. Automatic segmentation and characterisation of forest stand parameters using airborne LiDAR data multispectral and FOGIS data. Int. Arch. Photogram. Rem. Sens. Spatial Inform. Sci.

Dutta, D., Wang, K., Lee, E., Goodwell, A., Woo, D.K., Wagner, D., Kumar, P., 2017. Characterizing vegetation canopy structure using airborne remote sensing data. IEEE Trans. Geosci. Rem. Sens.

Ferraz, A., Bretar, F., Jacquemoud, S., Gonçalves, G., Pereira, L., Tomé, M., Soares, P., 2009. 3-d mapping of a multi-layered mediterranean forest using als data. Rem. Sens. Environ.

Ferraz, A., Saatchi, S., Mallet, C., Meyer, V., 2016. LiDAR detection of individual tree size in tropical forests. Remote Sens. Environ.

Goodfellow, I., Bengio, Y., Courville, A., 2016. Deep Learning. MIT Press.

Graham, B., Engelcke, M., Van Der Maaten, L., 2018. 3D semantic segmentation with submanifold sparse convolutional networks. CVPR.

Guo, Y., Wang, H., Hu, Q., Liu, H., Liu, L., Bennamoun, M., 2020. Deep learning for 3D point clouds: a survey. Trans. Pattern Anal. Mach. Intell.

Hamraz, H., Contreras, M.A., Zhang, J., 2016. A robust approach for tree segmentation in deciduous forests using small-footprint airborne LiDAR data. Int. J. Appl. Earth Observ. Geoinform.

Hamraz, H., Jacobs, N., Contreras, M., Clark, C., 2018. Deep learning for conifer/deciduous classification of airborne LiDAR 3D point clouds representing individual trees. ISPRS J. Photogram. Rem. Sens.

Hauglin, M., Rahlf, J., Schumacher, J., Astrup, R., Breidenbach, J., 2021. Large scale mapping of forest attributes using heterogeneous sets of airborne laser scanning and national forest inventory data. For. Ecosyst.

Hildebrandt, G., 1990. Operational remote sensing for forest damages. ISPRS J. Photogram. Rem. Sens.

Hyyppa, J., Kelle, O., Lehikoinen, M., Inkinen, M., 2001. A segmentation-based method to retrieve stem volume estimates from 3D tree height models produced by laser scanners. Trans. Geosci. Rem. Sens.

Ioffe, S., Szegedy, C., 2015. Batch normalization: accelerating deep network training by reducing internal covariate shift. ICML.

Kalinicheva, Ekaterina, 2021. Vegetation Stratum Occupancy Prediction from Airborne LiDAR 3D Point Clouds [Data set]. Zenodo. https://doi.org/10.5281/zenodo.5555758.

Ke, Y., Quackenbush, L.J., Im, J., 2010. Synergistic use of quickbird multispectral imagery and lidar data for object-based forest species classification. Rem. Sens. Environ.

Kingma, D.P., Ba, J., 2015. Adam: A method for stochastic optimization. ICLR.

Landrieu, L., Simonovsky, M., 2018. Large-scale point cloud semantic segmentation with superpoint graphs. CVPR.

Lang, N., Kalischek, N., Armston, J., Schindler, K., Dubayah, R., Wegner, J.D., 2021. Global canopy height estimation with GEDI LiDAR waveforms and bayesian deep learning. CoRR.

Latifi, H., Heurich, M., Hartig, F., Müller, J., Krzystek, P., Jehl, H., Dech, S., 2015. Estimating over- and understorey canopy density of temperate mixed stands by airborne LiDAR data. Forestry: An Int. J. Forest Res.

Latifi, H., Heurich, M., Hartig, F., Müller, J., Krzystek, P., Jehl, H., Dech, S., 2015. Estimating over- and understorey canopy density of temperate mixed stands by airborne LiDAR data. Forestry.

Liu, Z., Peng, C., Work, T., Candau, J.-N., DesRochers, A., Kneeshaw, D., 2018. Application of machine-learning methods in forest ecology: recent progress and future challenges. Environ. Rev.

Liu, Z., Wang, L., Zhang, Q., Gao, Z., Niu, Z., Zheng, N., Hua, G., 2019. Weakly supervised temporal action localization through contrast based evaluation networks. CVPR.

Lechner, A.M., Foody, G.M., Boyd, D.S., 2020. Applications in remote sensing to forest ecology and management. One Earth.

Lefsky, M.A., Cohen, W.B., Acker, S.A., Parker, G.G., Spies, T.A., Harding, D., 1999. Lidar remote sensing of the canopy structure and biophysical properties of douglas-fir western hemlock forests. Rem. Sens. Environ.

MacLean, D.A., 1996. Forest management strategies to reduce spruce budworm damage in the fundy model forest. The Forestry Chronicle.

Mallet, C., Bretar, F., Roux, M., Soergel, U., Heipke, C., 2011. Relevance assessment of full-waveform lidar data for urban area classification. ISPRS J. Photogram. Rem. Sens.

Martinuzzi, S., Vierling, L., Gould, W., Falkowski, M., Evans, J., Hudak, A., Vierling, K., 2009. Mapping snags and understory shrubs for a lidar-based assessment of wildlife habitat suitability. Remote Sens. Environ.

Maturana, D., Scherer, S., 2015. VoxNet: a 3D convolutional neural network for real-time object recognition. IEEE Int. Conf. Intell. Rob. Syst.

McKenzie, D., Crystal, L., 2011. Modeling understory vegetation and its response to fire. Models Plan. Wildl. Conserv. Large Landsc.

Milberg, P., Bergstedt, J., Fridman, J., Odell, G., Westerberg, L., 2008. Observer bias and random variation in vegetation monitoring data. J. Veg. Sci.

Morsdorf, F., Mårell, A., Koetz, B., Cassagne, N., Pimont, F., Rigolot, E., Allgöwer, B., 2010. Discrimination of vegetation strata in a multi-layered mediterranean forest ecosystem using height and intensity information derived from airborne laser scanning. Rem. Sens. Environ.

Naesset, E., 2007. Airborne laser scanning as a method in operational forest inventory: status of accuracy assessments accomplished in scandinavia. Scandinavian J. For. Res.

Nair, V., Hinton, G., 2010. Rectified linear units improve restricted Boltzmann machines. ICML.

Nwankpa, C., Ijomah, W.L., Gachagan, A., Marshall, S., 2018. Activation functions: comparison of trends in practice and research for deep learning.

Pekkarinen, A., Reithmaier, L., Strobl, P., 2009. Pan-European forest/non-forest mapping with Landsat ETM+ and CORINE Land Cover 2000 data. ISPRS J. Photogram. Rem. Sens.

Price, O.F., Gordon, C.E., 2016. The potential for lidar technology to map fire fuel hazard over large areas of australian forest. J. Environ. Manage.

Ratner, A., Bach, S., Varma, P., Ré, C., 2019. Weak supervision: the new programming paradigm for machine learning. Hazy Res.

Reitberger, J., Schnörr, C., Krzystek, P., Stilla, U., 2009. 3D segmentation of single trees exploiting full waveform LiDAR data. ISPRS J. Photogram. Rem. Sens.

Riegler, G., Ulusoy, A.O., Geiger, A., 2017. OctNet: learning deep 3D representations at high resolutions. CVPR.

Ruizhongtai Qi, C., Yi, L., Su, H., Guibas, L., 2017. PointNet++: Deep hierarchical feature learning on point sets in a metric space. NeurIPS.

Sandberg, D.V., Ottmar, R.D., Cushon, G.H., 2001. Characterizing fuels in the 21st century. Int. J. Wildland Fire.

Secord, J., Zakhor, A., 2007. Tree detection in urban regions using aerial LiDAR and image data. Geosci. Rem. Sens. Lett.

Seidel, D., Annighöfer, P., Thielman, A., Seifert, Q.E., Thauer, J.-H., Glatthorn, J., Ehbrecht, M., Kneib, T., Ammer, C., 2021. Predicting tree species from 3D laser scanning point clouds using deep learning. Front. Plant Sci.

Simonovsky, M., Komodakis, N., 2017. Dynamic edge-conditioned filters in convolutional neural networks on graphs. CVPR.

Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., Salakhutdinov, R., 2014. Dropout: a simple way to prevent neural networks from overfitting. J. Mach. Learn. Res.

Stefanidou, A., Gitas, I.Z., Korhonen, L., Georgopoulos, N., Stavrakoudis, D., 2020. Multispectral lidar-based estimation of surface fuel load in a dense coniferous forest. Rem. Sens.

Strîmbu, V.F., Strîmbu, B.M., 2015. A graph-based segmentation algorithm for tree crown extraction using airborne LiDAR data. ISPRS J. Photogram. and Rem. Sens.

Su, H., Maji, S., Kalogerakis, E., Learned-Miller, E., 2015. Multi-view convolutional neural networks for 3D shape recognition. ICCV.

Thomas, H., Qi, C.R., Deschaud, J.-E., Marcotegui, B., Goulette, F., Guibas, L., 2019. KPConv: flexible and deformable convolution for point clouds. ICCV.

Tong, P., Zhang, X., Han, P., Bu, S., 2021. Point in: Counting trees with weakly supervised segmentation network. Int. Conf. Pattern Recogn.

Velthof, G., Lesschen, J., Schils, R., Smit, A., Elbersen, B., Hazeu, G., Mucher, C., Oenema, O., 2014. Grassland areas, production and use. Methodol. Stud. Field Agro-Environ. Indic.

Venier, L.A., Swystun, T., Mazerolle, M.J., Kreutzweiser, D.P., Wainio-Keizer, K.L., McIlwrick, K.A., Woods, M.E., Wang, X., 2019. Modelling vegetation understory cover using LiDAR metrics. PloS ONE.

Vega, C., Hamrouni, A., El Mokhtari, S., Morel, J., Bock, J., Renaud, J.-P., Bouvier, M., Durrieu, S., 2014. PTrees: a point-based approach to forest tree extraction from LiDAR data. Int. J. Appl. Earth Observ. Geoinform.

Venier, L.A., Swystun, T., Mazerolle, M.J., Kreutzweiser, D.P., Wainio-Keizer, K.L., McIlwrick, K.A., Woods, M.E., Wang, X., 2019. Modelling vegetation understory cover using LiDAR metrics. PLoS ONE.

Wan Mohd Jaafar, W.S., Woodhouse, I.H., Silva, C.A., Omar, H., Abdul Maulud, K.N., Hudak, A.T., Klauberg, C., Cardil, A., Mohan, M., 2018. Improving individual tree crown delineation and attributes estimation of tropical forests using airborne LiDAR data. Forests.

Wei, X., Yu, R., Sun, J., 2020. View-GCN: view-based graph convolutional network for 3D shape analysis. CVPR.

Willem, W., Mead, B.R., 2000. Ocular estimates of understory vegetation structure in a closed picea glauca/betula papyrifera forest. J. Vegetation Sci.

Williams, J., Schönlieb, C., Swinfield, T., Lee, J., Cai, X., Qie, L., Coomes, D., 2020. 3D segmentation of trees through a flexible multiclass graph cut algorithm. Trans. Geosci. Remote Sens.

Wing, B.M., Ritchie, M.W., Boston, K., Cohen, W.B., Gitelman, A., Olsen, M.J., 2012. Prediction of understory vegetation cover with airborne LiDAR in an interior ponderosa pine forest. Remote Sens. Environ.

Young, D.S., Chen, X., Hewage, D.C., Nilo-Poyanco, R., 2019. Finite mixture-of-gamma distributions: estimation, inference, and model-based clustering. Adv. Data Anal. Class.

Yu, X., Hyyppä, J., Holopainen, M., Vastaranta, M., 2010. Comparison of area-based and individual tree-based methods for predicting plot-level forest attributes. Rem. Sens.

Zaheer, M., Kottur, S., Ravanbakhsh, S., Poczos, B., Salakhutdinov, R., Smola, A., 2017. Deep sets. NeurIPS.

Zou, X., Cheng, M., Wang, C., Xia, Y., Li, J., 2017. Tree classification in complex forest point clouds based on deep learning. Geosci. Remote Sens. Lett.