CLASSIFICATION OF TREE SPECIES AND STANDING DEAD TREES BY FUSING UAV-BASED LIDAR DATA AND MULTISPECTRAL IMAGERY IN THE 3D DEEP NEURAL NETWORK POINTNET++

S. Briechle¹*, P. Krzystek¹, G. Vosselman²

¹ Munich University of Applied Sciences, Munich, Germany - (sebastian.briechle, peter.krzystek)@hm.edu
² Faculty of Geo-Information Science and Earth Observation (ITC), University of Twente, Enschede, the Netherlands - george.vosselman@utwente.nl

Commission II, WG II/3

KEY WORDS: Object classification, vegetation mapping, deep neural network, point cloud processing

ABSTRACT:

Knowledge of tree species mapping and of dead wood in particular is fundamental to managing our forests. Although individual tree-based approaches using lidar can successfully distinguish between deciduous and coniferous trees, the classification of multiple tree species is still limited in accuracy. Moreover, the combined mapping of standing dead trees after pest infestation is becoming increasingly important. New deep learning methods outperform baseline machine learning approaches and promise a significant accuracy gain for tree mapping. In this study, we performed a classification of multiple tree species (pine, birch, alder) and standing dead trees with crowns using the 3D deep neural network (DNN) PointNet++ along with UAV-based lidar data and multispectral (MS) imagery. Aside from 3D geometry, we also integrated laser echo pulse width values and MS features into the classification process. In a preprocessing step, we generated the 3D segments of single trees using a 3D detection method. Our approach achieved an overall accuracy (OA) of 90.2% and was clearly superior to a baseline method using a random forest classifier and handcrafted features (OA = 85.3%). All in all, we demonstrate that the performance of the 3D DNN is highly promising for the classification of multiple tree species and standing dead trees in practice.

1. INTRODUCTION

Forest inventories based on remote sensing data, particularly lidar point clouds fused with optical imagery, are the most prominent options for the inventory of forest structural variables (Latifi & Heurich, 2019). Forest attributes such as above-ground biomass and growing stock can be estimated from the spatial distribution of tree species and dead wood. Tree-level approaches utilize segmented single trees for forest inventory parameter estimations. For forest managers and nature conservationists, information about tree species, especially the classification of dead trees, is of increasing importance because forests are suffering from changing climatic conditions.

In the past, extensive research has been conducted to apply appropriate classifiers like support vector machines (SVM), random forests (RF), or logistic regression to classify presegmented single trees with respect to tree species (Fassnacht et al., 2016) and dead trees (Yao et al., 2012). Most methods have been based on handcrafted feature sets extracted from airborne laser scanning (ALS) data and multispectral (MS) or hyperspectral imagery. Polewski (2017) successfully combined single 3D tree segments with MS aerial imagery to detect standing dead trees in a binary classification. The authors incorporated MS features generated from the covariance matrix of three image channels and classified dead trees with an overall accuracy (OA) of ca. 88%. Moreover, Degerickx et al. (2018) distinguished healthy (precision = 93%, recall = 83%) from unhealthy (precision = 71%, recall = 88%) deciduous trees using ALS data and hyperspectral imagery in a regression method. Recently, Amiri et al. (2019) reported a combined classification of tree species and standing dead trees with crowns. Using a huge feature set generated from multi-wavelength lidar point clouds, four tree classes could be classified with an OA of 82%. Interestingly, dead trees were only classified with 76% precision and 73% recall. However, all in all, the performance of these approaches for individual tree species classification is still not sufficient for practical use.

Currently, the utilization of high-performance deep learning (DL) methods as a classification tool for 3D sensed data has gained a large amount of interest in the remote sensing community. Various authors have demonstrated that standard machine learning (ML) concepts using, for example, SVM or RF, can be outperformed by DL-based methods (Voulodimos et al., 2018; Liu et al., 2018). One big advantage of deep neural networks (DNNs) is the automatic extraction of features as part of the training process, or so-called representation learning (LeCun et al., 2015). Griffiths & Boehm (2019) emphasized four general types of DL approaches for scene understanding from 3D sensed datasets. To utilize well-proven and efficient 2D convolutional neural networks (CNNs), irregular and unordered 3D point clouds can either be transformed into RGB-depth (RGB-D) images (Zhou et al., 2018) or utilized to render multiview images (Qi et al., 2016). Furthermore, the authors discussed volumetric approaches that discretize raw 3D data, that is, as regular 3D voxel grids, and that use 3D convolutions to extract meaningful information (Zhou & Tuzel, 2018). Finally, powerful network architectures have been developed, enabling a direct input of raw and unstructured point clouds without the need for a prior rasterization or voxelization. These innovative networks like Pointnet (Qi et al., 2017a), PointNet++ (Qi et al., 2017b), PointCNN (Li et al., 2018), and Super Point Graphs (Landrieu & Simonovsky, 2018) allow end-to-end classification.

*Corresponding author
To the best of our knowledge, the application of DNNs for the classification of presegmented single trees has been sparsely investigated. In urban study areas, Wegner et al. (2016) applied latest CNN-based methods to extensive datasets comprising aerial and street view images. The authors demonstrated that multiview imagery significantly improved tree detection and tree species classification, reaching close to human performance. Furthermore, Hartling et al. (2019) classified eight tree species using DenseNet (Huang et al., 2017), data from satellite imagery, and lidar data (approximately 1 point/m²) in urban study areas (OA = 83%). Moreover, Hamraz et al. (2019) generated images from ALS point clouds and made use of a CNN to classify overstory coniferous and deciduous trees in a natural forest with a cross-validated classification accuracy of 92% and 87%, respectively. So far, using “real” 3D DNNs for vegetation mapping has not been researched sufficiently. Recently, Briechle et al. (2019) achieved promising results for adapting PointNet++ to the semantic labeling of extensive ALS point clouds, resulting in an OA = 85% for spruces and beeches.

The key idea of the current study was to adapt a 3D DNN for the classification of multiple tree species based on presegmented single tree objects. Specifically, we applied PointNet++ to a dataset composed of UAV-based lidar (including laser echo pulse width) and five-channel MS imagery. All in all, PointNet++ achieved excellent classification results on the single-tree level and clearly outperformed the baseline method. Furthermore, we demonstrated that MS data clearly enhanced the classification result.

In the following sections, we address the study area, sensors, data preprocessing, and reference data. Subsequently, we present the methodology for tree species classification using PointNet++ and compare it with the baseline method. Next, we demonstrate the conducted experiments and the main outcomes, including a comparison of both methods. Finally, we discuss the results referring to previous research and draw conclusions.

2. MATERIALS

2.1 Study area

In two unmanned aerial vehicle (UAV) flight missions (November of 2017 and April of 2018), both lidar data and MS images were captured in the study area Chornobyl Exclusion Zone (ChEZ), located approximately 1.5 km west of the Chornobyl Nuclear Power Plant (ChNPP) (Figure 1). This densely vegetated area (37 ha) comprises approximately 400 trees/ha with tree heights of up to 30 m (Bonzom et al., 2016). The three main tree species are silver birch (Betula pendula), scots pine (Pinus sylvestris), and black alder (Alnus glutinosa). Moreover, standing dead trees with crowns (solely pines) can be found in the area.

2.2 Sensors and data preprocessing

During both flight missions, an octocopter was utilized; it was developed by a team from the Department of Nuclear Physics Technologies of the Institute of Environment Geochemistry of the National Academy of Sciences of Ukraine. The copter enabled surveys, simultaneously recording with the lidar system and two MS cameras.

2.2.1 Lidar data

Lidar data with a nominal point density of 53 points/m² were collected in five automatic flights using a YellowScan Mapper I laser scanner at a constant altitude of 50 m. To generate a geometrically reliable 3D dataset, various postprocessing steps were conducted. First, differential global navigation satellite system (GNSS) post-processing using a GNSS base station resulted in flight trajectories with centimeter-level precision. Second, the boresight angles provided by the manufacturer were checked in a calibration flight. Third, geometrically consistent lidar point clouds were generated by simultaneously aligning the flight strips (Jalobeanu & Gonçalves, 2014). Fourth, absolute 3D georeferencing was achieved by fitting the ALS point cloud to the enclosing polygons of a nearby building. Additionally, the sensor provided the intensity values for each laser point equivalent to the widths of the echo pulses (EW) measured at a fixed internal sensor threshold. Because tree species classification can benefit from these measurements, we performed a data-driven correction step (Briechle et al., 2020). Finally, we performed single tree segmentation using a normalized cut algorithm, resulting in single tree point clouds and enclosing tree polygons (Reitberger et al., 2009).

2.2.2 MS imagery

Five-band MS images (ground sample distance = 8.9 cm) were captured using two MicaSense.
RedEdge cameras (spectral range 475–840 nm) mounted in a twisted configuration with an angle of 22.5° (50% side overlap). Guaranteeing an extended camera footprint (field of view 70°) equal to the lidar footprint, this setup allowed for a constant line-to-line distance for both lidar and MS sensors in a combined survey. For postprocessing the five-channel images (blue = B, green = G, red = R, red edge = RE, near infrared = NIR), we utilized structure-from-motion software\(^1\). The processing steps included bundle adjustment (mean reprojection error of 1.3 pixels), calibration of reflectance, and the generation of dense photogrammetric 3D point clouds (80 points/m\(^2\)) and 10 cm orthomosaics. Because the overflown study area is inaccessible, no ground control points could be used. Therefore, photogrammetric point clouds were registered to georeferenced lidar point clouds using an iterative closest point algorithm\(^2\), resulting in a root mean squared error of 0.237 m (Briechle et al., 2018).

2.3 Reference data

Because of the high radiation dose rates within the study area, reference data were generated based on visual interpretation of 3D point clouds and MS imagery. In total, we manually labeled 1135 single tree segments assigned to the four tree classes “pine” (368 samples), “birch” (243 samples), “alder” (283 samples), and “dead tree” (241 samples), respectively.

3. METHODOLOGY

In the following, we describe the baseline method including feature engineering, classifier training and feature selection procedure. Furthermore, we give a detailed description of the classification process with the 3D DNN. Specifically, we address the preparation of dataset as well as network training, hereby focusing on hyperparameters and data augmentation.

3.1 Baseline method

3.1.1 Extraction of handcrafted features

The feature set generated from 3D lidar data (Table 1) comprised features based on the tree geometry (GEOM) and the echo characteristics (EC).

| Features | Definition |
|----------|------------|
| GEOM(1-10)\(^1\) | Density distribution of points per height layer. |
| GEOM(11-20) | Vertical distribution of tree substance per height layer. |
| GEOM(21-30) | Mean distance of points to segment center. |
| GEOM(31-32) | Standard deviation (std) of distance from crown points to segment center, in \(x\) and \(y\) direction. |
| EC1 | Mean \(EW\) of points of a single tree. |
| EC(2-11) | \(EW\) of points of a single tree per height layer. |
| EC12 | \(Σ\) single / \(Σ\) first) reflections. |
| EC13 | \(Σ\) single / \(Σ\) first) reflections. |
| EC14 | \(Σ\) first + \(Σ\) middle) / \(Σ\) single + \(Σ\) last) reflections. |

\(^1\) Increasing number from bottom (1) to top (10).

Table 1. 32 GEOM and 14 EC features.

Moreover, we developed distinctive features from the five-channel orthomosaics. For this purpose, we computed five vegetation indexes (VI) from the available spectral channels. First, we calculated the Normalized Difference Vegetation Index (NDVI), a well-known index sensitive to healthy vegetation rich in chlorophyll and robust over a wide range of conditions (Rouse Jr et al., 1973).

\[
NDVI = \frac{NIR - R}{NIR + R} \quad (1)
\]

Second, utilizing both RE and NIR channels, the Red Edge Normalized Difference Vegetation Index (RENDVI) was computed (Gitelson & Merzlyak, 1994). This index is a NDVI modification and has been developed for applications including forest monitoring and vegetation stress detection. RENDVI is capable of detecting small changes in canopy foliage content (Sim & Gamon, 2002).

\[
RENDVI = \frac{NIR - RE}{NIR + RE} \quad (2)
\]

Third, we introduced a NDVI-inspired index. Instead of the NIR channel, the RE channel was used to generate Red Edge Difference Vegetation Index (REDI).

\[
REDVI = \frac{RE - R}{RE + R} \quad (3)
\]

Fourth, we utilized the Modified Red Edge Simple Ratio (MRESR), which is used for forest monitoring and vegetation stress detection, incorporating a correction for leaf specular reflection (Datt, 1999).

\[
MRESR = \frac{NIR - B}{RE - B} \quad (4)
\]

Fifth, we included the Modified Chlorophyll Absorption Ratio Index (MCARI), a well-suited index to indicate the relative abundance of chlorophyll. Daughtry et al. (2000) introduced this index, minimizing the combined effects of soil and non-photosynthetic surfaces.

\[
MCARI = \frac{RE}{R} \cdot (0.8 \cdot RE - R - 0.2 \cdot G) \quad (5)
\]

We superimposed the enclosing tree polygons on the orthomosaic (Figure 2) to mask VI pixels located within the tree segments. For each of these pixels, statistical features were calculated and standardized for each object (Table 2). These resulting 60 MS features were complemented with 10 independent interchannel covariance values generated from the covariance matrix of the five VI channels. Using this feature set, an RF

![Figure 2. Superimposed tree polygons on the orthomosaic.](image-url)
classifier was trained on the labeled dataset and optimized in a three-times-repeated five-fold cross-validation. Finally, we identified the five most important MS features by evaluating the feature ranking based on the mean decrease in accuracy. In descending order, these were NDVI_{skewness}, MRESR_{perc90}, NDVI_{perc90}, RENDVI_{mode}, and MRESR_{mode}.

| Features | Definition |
|----------|------------|
| max, min, interval | Maximum value, minimum value, and range (max-min) |
| mean, std | Mean value and standard deviation. |
| mode | Value that appears most often. |
| skewness | Measure of asymmetry of the probability distribution. |
| kurtosis | Measure of tailedness of the probability distribution. |
| perc(25,50,75,90) | 25th (‘1st quartile’), 50th (‘median’), 75th (‘3rd quartile’), and 90th percentile. |

Table 2. Object-based statistical MS features.

3.2.1 Classifier training
For the baseline method, the dataset comprised 32 GEOX features and 14 EC features (see Table 1), as well as the five most important MS features generated from the VI orthomosaics. In a preprocessing step, highly correlated redundant features were eliminated from the feature set, here based on the application of a threshold (0.9) to feature-to-feature cross-correlation (Briechle et al., 2018). Next, an RF classifier was trained, including recursive feature elimination (RFE) based on Kuhn (2008) and a feature relevance assessment. Finally, the generalization quality of the RF classifier was verified by calculating classification metrics (OA, \(\kappa\), precision, recall, and \(F_1\) score) on the test dataset.

3.2 Classification using 3D DNN

PointNet++ is an advanced version of PointNet and incorporates hierarchical feature learning by extracting features from multiple contextual scales. Therefore, fine-grained local patterns and more general global features can be captured. In the following sections, we demonstrate the methodology for the utilization of PointNet++ to classify three tree species (pine, birch, alder) and standing dead trees using the pytorch implementation from Wijmans (2018).

3.2.1 Preparation of dataset

Point sampling: For object classification, PointNet++ requires a constant number of 3D points per sample (e.g., NUM\_POINT = 1024, see Table 3). In practice, the distribution of points per tree is fairly heterogeneous due to variations in the size, geometry, and species of single trees. Thus, an effective approach must meet the following conditions: First, a constant and adequate number of points per tree has to be guaranteed, and loss of information during downsampling needs to be minimized. Second, deletion of samples containing less points than NUM\_POINT but still exceeding an acceptable number of points should be avoided. Third, synthetic generation of redundant information by extensive upsampling is not reasonable. Therefore, we introduced the two thresholds \(\theta_1\) and \(\theta_2\) in a combined sampling approach. \(\theta_1\) was utilized to randomly reduce the points per tree to a certain value. Figure 4 exemplary shows the number of remaining samples per class, in dependence of \(\theta_1\). To preserve the selected objects comprising less than \(\theta_1\) points in the dataset, we made use of a second threshold, \(\theta_2\). Trees containing at least \(\theta_2\) points were sampled up to \(\theta_1\) points using random copies of points. All in all, our procedure handled the trade-off between upsampling and downsampling, assuming that both thresholds are chosen appropriately.

Dataset generation: Initially, the remaining samples were balanced according to the four occurring tree classes. Next, all single point clouds were standardized by subtraction of the mean \(x\), \(y\), and \(z\) coordinates and division by the \(x\), \(y\), and \(z\) standard deviation. Consequently, all objects were rescaled and had a mean of 0 and a standard deviation of 1. Practically, the purpose of standardization is to make the classification results independent of the geometry within each tree class, for example, the tree height and the crown width. Moreover, the \(EW\) values were standardized as well. Subsequently, we calculated surface normals (Figure 3) using the estimate_normals function.
from the open source library Open3D (Zhou et al., 2018). The two key arguments of the function, radius and max_nn, were set to 0.5 and 30, respectively. The parameter radius specifies the search radius for the neighborhood definition, whereas max_nn defines the maximum number of nearest neighbors to be considered to save computation time. Next, the top five MS features (see section 3.1.1) were integrated by assigning the standardized values to each 3D point of an object (tree species, dead tree). Note that this procedure provides additional point attributes. All in all, we generated a dataset comprising raw point clouds, surface normals, echo widths per point, and five previously calculated handcrafted MS features.

### 3.2.2 Training and validation

Hyperparameters: PointNet++ is an off-the-shelf 3D DNN. Nevertheless, it is essential to consider various options to optimize network performance for specific classification tasks without model overfitting. To get a well-performing network, the most decisive PointNet++ hyperparameters were adjusted using a combination of manual search and automated grid search (Table 3). For some parameters, the default values were convenient and, therefore, remained unchanged.

| Hyperparameter | Value | Declaration |
|----------------|-------|-------------|
| NUM_CLASSES    | 4     | Number of object categories. |
| NUM_POINT      | 1024  | Number of points per sample. |
| MAX_DROPOUT    | 0.5   | Maximal dropout rate. |
| BATCH_SIZE     | 8     | Number of samples per batch. |
| MAX_EPOCH      | 300   | Number of training epochs. |
| BASE_LR        | 1e-3  | Initial learning rate. |
| LR_DECAY       | 0.7   | Initial learning decay. |
| BN_MOMENTUM    | 0.5   | Initial batch norm momentum. |
| BN_DECAY       | 0.5   | Batch norm momentum decay. |
| OPTIMIZER      | adam  | Optimization algorithm. |
| WEIGHT_DECAY   | 1e-4  | L2 regularization coefficient. |

Table 3. Hyperparameters and default / optimized values for PointNet++.

Data augmentation: A popular method to avoid model overfitting on a small training dataset is the utilization of data augmentation. Furthermore, performing data augmentation during network training helps to make the neural network more robust against object variation. Before each training epoch, we shuffled the order of samples to generate random batches. Next, we performed random transformations of the standardized 3D objects by following common practice including scaling (range = [0.80, 1.25]), rotation around vertical axis (range = ±0.1 [m]), and 3D translation of the entire point cloud (range = ±0.1 [m]). Furthermore, we set the random input dropout parameter to MAX_DROPOUT = 50%, thereby increasing the robustness to varying point density and occluded object parts. Practically, the input points for each instance were randomly dropped out, generating subvolumes of the objects.

Model evaluation: For testing of the trained network, class labels were predicted on trees that were not used for the training. We compared these class predictions with the reference labels and calculated standard metrics OA, κ, precision, recall, and F1 score. For final evaluation, we used the model showing the lowest validation loss.

### 4. EXPERIMENTS AND RESULTS

#### 4.1 Experimental setup

The original reference dataset was prepared for object classification, performing point sampling (θ1 = 1024, θ2 = 512) and class balancing (see section 3.2.1). The remaining 668 samples (167 per class) were divided into 464 training and 204 test samples using a split ratio of 0.7. Note that for a fair comparison of 3D DNN and baseline method, the particular training and test datasets were identical. For network training and validation, we used an Intel Xeon Platinum 8160 CPU and a Nvidia Titan V GPU (NVIDIA Corporation, 2019) with 12 GB on Ubuntu 18.04, reaching a processing time of approximately 10 seconds per epoch.

We performed classification with PointNet++ on four different datasets investigating their impact on the classification result. In more detail, the datasets represented geometry (GEOM, see Figure 5), geometry and surface normals (GEOM+normals, see Figure 6), geometry and EW values (GEOM+EW, see Figure 7), and all data subsets (GEOM+EW+MS, see Figure 8). Furthermore, we conducted comparative experiments with the previously described baseline method (RF). For validation, we compared both classifier procedures on the same test dataset.

#### 4.2 General classification results

PointNet++ outperformed the baseline method in all experiments (Table 4). Especially, if only geometry information was used, PointNet++ and automatically extracted features led to a result that was 17.7% better than the baseline method using 32 “standard” handcrafted geometry features. Adding surface normals improved the DNN result by 1.4%. Here, no comparison to the baseline was available. Fusing geometry

![Figure 5. Confusion matrices on the test dataset using only geometry information.](image)

![Figure 6. Confusion matrices on the test dataset using only geometry and surface normals.](image)
5. DISCUSSION

The proposed framework using PointNet++ for the classification of three single tree species and standing dead trees performed fairly good. Especially, when classification was only conducted based on geometry information, the results were significantly better than those of the baseline method. Obviously, handcrafted geometry features are considerably inferior to information automatically extracted in a DNN. If we analyze the confusion matrices in Figure 8a, we notice a higher confusion between alder and dead trees. Very likely, the tree geometry and spectral appearance of alder is similar to dead pines. Stepwise improvement of the results produced by PointNet++ was rather low when we fused surface normals and EW values with geometry data (1.5% and 1.0%, respectively). Interestingly, adding surface normals particularly increased the classification accuracy for dead trees. Also very important, the classification of pine, the only conifer in our study area, profited most by the EW values ($F_1$ score = 0.90), thereby confirming the findings of Reitberger et al. (2009). Furthermore, we included five MS features that were selected by the RF-based feature assessment. Embedding these features, the overall results were considerably enhanced for both methods by approximately 11% (see Table 4). Especially, the classification of birch and dead trees benefited the most from these MS features. Note that at the time of data collection,
birches had already sprouted. Therefore, their characteristic spectral appearance supported the classification significantly. Investigating the related work reveals that our approach achieves very promising and competitive results. For the classification of individual tree species, most previous studies based on classic ML approaches did not reach an acceptable accuracy level of up to 90%. Yu et al. (2017) classified three tree species using multispectral ALS data and an RF classifier (OA = 86%). Moreover, Shi et al. (2018) categorized five species, fusing ALS data with hyperspectral imagery (OA = 84%). Kamińska et al. (2018) classified three tree species (spruce, pine, deciduous), each of them further categorized as “dead” or “alive”. Their approach using an RF classifier and features generated from ALS data and color-infrared imagery reached an OA of 94%. Nevertheless, a comprehensive and, thus, fair comparison to other studies that have addressed classification of presegmented single trees is challenging. Collecting data using a huge variety of sensor platforms and sensor types, utilized datasets strongly differ in their spatial, spectral, and temporal resolution. Additionally, the type of study area (urban, natural, managed) and number of samples and classes fluctuate as well.

We would also like to address some limitations of PointNet++ for classification tasks. Because PointNet++ can only deal with objects comprising a constant number of points, point sampling including upsampling and downsampling must be performed. Thereby, information loss is unavoidable and must be minimized based on reasonable thresholds (see section 3.2.1), depending on the specific point density of the dataset. Nevertheless, this disadvantage is clearly compensated by the DNN performance with its ability to automatically extract meaningful information from 3D datasets. Moreover, 3D DNNs like PointNet++ need to be trained from scratch using a specific and fairly high number of training samples. Contrary to well-known 2D CNNs, no publicly available databases like ImageNet (Deng et al., 2009) can be used for transfer learning and reasonable weight initialization.

6. CONCLUSION

Our experiments demonstrated that 3D DNN PointNet++ could successfully be applied to the classification of three tree species - pine, birch, and alder - and standing dead trees. Fusing UAV-based lidar data and features generated from five-channel MS imagery, we achieved an OA better than 90% on single-tree level. Moreover, classification with PointNet++ was clearly superior to the described baseline method in all cases. All in all, our DL-based approach provided detailed and reliable 3D vegetation maps at the tree level in the study area ChEZ. In a next step, a large scale experiment in an extended forest area is intended to verify the promising results of this current study, thereby demonstrating the suitability for practical use.

ACKNOWLEDGEMENTS

The authors would like to thank N. Molitor from Plejades GmbH as well as V. Antropov, O. Tretyak and the colleagues from the State Central Enterprise for Radioactive Waste Management for the technical support in the Chernobyl Exclusion Zone. The research was funded by Federal Ministry of Education and Research (BMBF), grant number 13FH005IX6.

References

Amiri, N., Krzystek, P., Heurich, M., Skidmore, A., 2019. Classification of Tree Species as Well as Standing Dead Trees Using Triple Wavelength ALS in a Temperate Forest. Remote Sensing, 11(22).

Bonzom, J.-M., Hättenschwiler, S., Lecomte-Pradines, C., Chauvet, E., Gaschak, S., Beaugel- Seiller, K., Dellavedova, C., Dubourg, N., Maksimenko, A., Garnier-Laplace, J., Adam-Guillermin, C., 2016. Effects of radionuclide contamination on leaf litter decomposition in the Chernobyl ex- clusion zone. Science of the Total Environment, 562, 596-603.

Briechle, S., Krzystek, P., Vosselman, G., 2019. Semantic labeling of ALS point clouds for tree species mapping using the deep neural network PointNet++. International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences - ISPRS Archives, 42(2/W13), 951-955.

Briechle, S., Molitor, N., Krzystek, P., Vosselman, G., 2020. Detection of radioactive waste sites in the Chernobyl Exclusion Zone using UAV-based lidar data and multispectral imagery. Under review.
