Review on Convolutional Neural Networks (CNN) in Vegetation Remote Sensing

This is an uncorrected preprint. The peer-reviewed manuscript is published in ISPRS Journal of Photogrammetry and Remote Sensing. The Web- and pdf-version can be found at: https://authors.elsevier.com/a/1cQX33I9x1cgg0.

Teja Kattenborn[1,*], Jens Leitloff[2], Felix Schiefer[3], and Stefan Hinz[2]

[1]Remote Sensing Centre for Earth System Research, Leipzig University, Talstr. 35, 04103 Leipzig, Germany

[2]Institute of Photogrammetry and Remote Sensing (IPF), Karlsruher Institute of Technology (KIT), Englerstr. 7, 76131 Karlsruhe, Germany

[3]Institute for Geography and Geoecology (IFGG), Karlsruher Institute of Technology (KIT), Kaiserstr. 12, 76131 Karlsruhe, Germany

[*]Corresponding author: Teja Kattenborn, teja.kattenborn@uni-leipzig.de

Abstract

Identifying and characterizing vascular plants in time and space is required in various disciplines, e.g. in forestry, conservation and agriculture. Remote sensing emerged as a key technology revealing both spatial and temporal vegetation patterns. Harnessing the ever growing streams of remote sensing data for the increasing demands on vegetation assessments and monitoring requires efficient, accurate and flexible methods for data analysis. In this respect, the use of deep learning methods is trend-setting, enabling high predictive accuracy, while learning the relevant data features independently in an end-to-end fashion. Very recently, a series of studies have demonstrated that the deep learning method of Convolutional Neural Networks (CNN) is very effective to represent spatial patterns enabling to extract a wide array of vegetation properties from remote sensing imagery. This review introduces the principles of CNN and distils why they are particularly suitable for vegetation remote sensing. The main part synthesizes current trends and developments, including considerations about spectral resolution, spatial grain, different sensors types, modes of reference data generation, sources of existing reference data, as well as CNN approaches and architectures. The literature review showed that CNN can be applied to various problems, including the detection of individual plants or the pixel-wise segmentation of vegetation classes, while numerous studies have evinced that CNN outperform shallow machine learning methods. Several studies suggest that the ability of CNN to exploit spatial patterns particularly facilitates the value of very high spatial resolution data. The modularity in the common deep learning frameworks allows a high flexibility for the adaptation of architectures, whereby especially multi-modal or multi-temporal applications can benefit. An increasing availability of techniques for visualizing features learned by CNNs will not only contribute to interpret but to learn from such models and improve our understanding of remotely sensed signals of vegetation. Although CNN has not been around for long, it seems obvious that they will usher in a new era of vegetation remote sensing.

Keywords— Convolutional Neural Networks (CNN), Deep Learning, Vegetation, Plants, Remote Sensing, Earth Observation

Contents

1 Introduction

Locating and characterizing vascular plants in time and space is key to various tasks: For instance, nature conservation in the context of global change and biodiversity decline can only be successfully implemented and supervised with accurate spatial representations of the state, structure and functioning of ecosystems and its flora (Nagendra et al. 2013; Pettorelli et al. 2017; Turner et al. 2003). Forestry requires regular and extensive information on forest stands, including their structure, timber volume, species composition, and forest damage (Fassnacht et al. 2016; McRoberts et al. 2007; White et al. 2016). In agriculture, there is a growing demand for geoinformation that facilitates resource efficiency and a reduction of environmental impacts (cf. precision farming), including fine-scale predictions of yield, weed infestations, and plant vigor (Atzberger et al. 2013; Mulla 2013). Concerning all of these tasks and requirements remote sensing continuously establishes as a key technology.

In the last decades, various technological advances resulted in growing availability of remote sensing data revealing vegetation patterns on both spatial and temporal domains (Colomina et al. 2014; Toth et al. 2016). Novel remote sensing platforms, such as swarms of microsatellites, or unmanned aerial vehicles (UAV), facilitate a bird's eye view on vegetation canopies with increasing spatial detail. Synthetic-aperture radar (SAR), and terrestrial or airborne lasers-scanning enable to capture the three-dimensional structure of multilayered canopies. Additionally, there is an ongoing trend of data sharing and open access (cf. *OpenAerialMap*, *NEON* programme of the US National Science Foundation, EU's and ESA's *Copernicus Open Access Hub*).

These growing opportunities for vegetation remote sensing come hand in hand with several challenges, including increased data volumes and computational loads as well as more diverse data structures with increasing dimensions (spatial, temporal, spectral) often featuring complex relationships. Moreover, the various vegetation related tasks and applications fields can differ greatly in their inherent processes and requirements. Hence, harnessing remote sensing data for vegetation assessments and monitoring requires efficient, accurate, and flexible analytical methods.

In the context of image analysis and computer vision, deep learning is currently paving new avenues for remote sensing analysis (Chollet 2017; Hoeser et al. 2020; Huang et al. 2018; Ronneberger et al. 2015; L. Zhang et al. 2019; Zhu et al. 2017). In contrast to the previous **shallow** neural network approaches that have been under investigation for decades, **deep** learning is characterized by a significantly increased number of successively connected neural layers. This increased amount of layers and transformations can reveal higher-level features and more abstract concepts uncovering more complex and hierarchical relationships. A series of studies has demonstrated that this increased depth can indeed enhance the retrieval of vegetation-related information contained in remote sensing data (cf. section 3.6). At the same time, increasing transformations and, thus, deeper levels of complexity commonly require more training data and computational loads. Nevertheless, deep learning became very popular due to several, corresponding technical developments, including efficient data processing techniques (e.g. data augmentation or non-linear activation functions, see section 2.3 and 3.2), high-performance graphic cards, cloud-computing, as well as open data initiatives providing annotated data.

These developments enable an efficient calculation of countless non-linear transformations of the respective input data and, thus, form the core for the essential strength of deep learning - namely the ability of **end-to-end-learning**. Previous data analysis methods in remote sensing usually require feature engineering, which is the heuristic selection of appropriate transformations and hand-crafting latent variables from the input data prior to modelling. Examples in the field of vegetation remote sensing are spectral indices (Adam et al. 2010) or texture metrics (Haralick 1979), whereas the numerous ways to derive such variables make it often impossible and inefficient to derive the most effective set of predictors. Moreover, defining the most appropriate predictors for vegetation analysis can be challenging, as this may not only require knowledge on the biochemical and structural plant properties but also on how these interact with the electromagnetic signal measured by the sensor. By contrast, with deep learning, the neural network itself can learn the data transformations that are best to solve the problem at hand.

The class of deep learning algorithms most commonly used for spatial pattern analysis are convolutional neural networks (CNNs or ConvNets). CNNs are designed to learn the spatial features, e.g. edges, corners, textures, or more abstract shapes, that best describe the target class or quantity. The core for learning these features are manifold and successive transformations of the input data (convolutions) on different spatial scales (by pooling operations). This facilitates identifying and combining both low-level features and high-level concepts. The functioning of a CNN can, hence, be regarded as a mimicry of the animal cortex (Angermueller et al. 2016; Cadieu et al. 2014), where analogously numerous visual stimuli at varying scales are perceived in the field of vision (counterpart of an image) and the contained spatial features and their spatial context serves to identify objects. For example, the shape of a leaf does not necessarily indicate the corresponding vegetation type, but its close proximity to branches and the tall and bulky canopy suggest that it belongs to a tree and not to a herb. The effectiveness of deep learning and particularly CNNs undoubtedly revolutionized our possibilities to analyse spatial patterns in Earth observation data. Reference is made here to previous and valuable comments and reviews, including a review by Zhu et al. (2017) on the general principles and potentials of deep learning in remote sensing, Hoeser et al. (2020) summarizing common frameworks and an in depth overview on architectures for Earth observation data analysis, a comment by Brodrick et al. (2019) highlighting potentials of CNN for segmentation tasks in ecology and Reichstein et al. (2019) providing perspectives on how deep learning in general can advance earth system science.

The remote sensing of vegetation is characterized by special requirements and challenges, such as the often complex acquisition of reference data or the understanding of the vegetation specific radiative transfer, the resulting sensor-specific electromagnetic signals and their dynamics across the phenology. The present review therefore concentrates specifically on CNN applications in the field of vegetation remote sensing. A series of recent studies have demonstrated that CNNs enable to reveal accurate spatial representations of vegetation properties, such as detecting individual plant organs or individuals, classifying species and communities or quantifying plant traits, from all kinds of remote sensing sensors and platforms. Still, CNN-based vegetation remote sensing is a very topical but young field of research. People with a background in remote sensing or vegetation science may require procedural knowledge on the working principles of CNNs and the anticipated potentials for vegetation mapping. In contrast, people from computer sciences may require declarative knowledge on application tasks in vegetation science, on types and availability of remote sensing data suitable for vegetation analysis, or on the relationship between remotely sensed signals and vegetation properties. Thus, the overall aim of this review is to link procedural and declarative knowledge and provide an introduction and synthesis on the current state of the art on the utility of CNNs for vegetation remote sensing.

The present review is organized into three main sections: Chapter 2 briefly introduces the basic principles and the general functioning of CNNs and deduces why it is such a promising method for remote sensing of vegetation. Chapter 3 provides a summary and meta-analysis on the corresponding literature and synthesizes the current state of the art and challenges, including:

- common CNN approaches, architectures and strategies for the retrieval of vegetation properties,
- an overview of common applications tasks and demonstrated potentials in the context of agriculture, forestry and conservation,
- challenges and corresponding solutions regarding reference data quantity and quality of continuous and discrete vegetation variables,
- a consideration of spatial and spectral resolution for CNN-based vegetation remote sensing and considerations towards different sensors, platforms and combinations thereof.

Lastly, chapter 4 gives concluding remarks and discusses possible future directions and developments.

2 Principles of CNNs and relevance for vegetation remote sensing

This chapter introduces the basic principles of CNN, including the functioning of convolutions, features that make convolutions suitable for vegetation analysis, and how a CNN is commonly trained and implemented.

2.1 Basic functioning and structure of CNNs

As any typical neural network-type model, CNNs are based on **neurons** that are organized in **layers** and can, hence, learn hierarchical representations. The neurons between layers are connected through weights and biases. The initial layer is the input layer, e.g. remote sensing data, and the last layer is the output, such as a predicted classification into plant species. In between are **hidden layers** transforming the feature space of the input in a way that it matches the output. CNNs include at least one convolutional layer as a hidden layer to exploit patterns (in the context of this review predominantly spatial patterns).

It can also include other non-convolutional layers. Convolutional layers include multiple optimizable filters (Fig. 1) that transform the input or preceding hidden layers. The number of filters defines the **depth** of a convolutional layer. The resulting transformations are aimed to reveal patterns that are decisive for the problem at hand. The decisive patterns are iteratively learned through convolving, which is essentially the sliding of the filter over the layer and the calculation of the dot-product of the filter and the layer's values. The result is a new layer of dot-products for each filter, also called a **feature map** (Fig. 1). The early feature maps in a CNN may include simple and fine scaled patterns, such as corners, circles, or edges. The derived feature maps then serve as input for the next layer, e.g. another convolutional layer or a final layer that predicts an outcome based on the detected features. In deeper layers of a network, convolving usually reveals more abstract patterns and higher-level concepts, such as leaf forms, branching patterns or habit. During model training, randomly initialized filters will be iteratively optimized to detect the relevant image features (the training procedure is described in Section 2.3). The combination of several successive convolutional layers with their numerous filters, hence, enables the network to learn and combine even subtle image features, revealing if a class is present in an image or not (see Fig. 1 for a tree-species-specific activation of the network, more details on class activation mapping in Section 3.6.2)

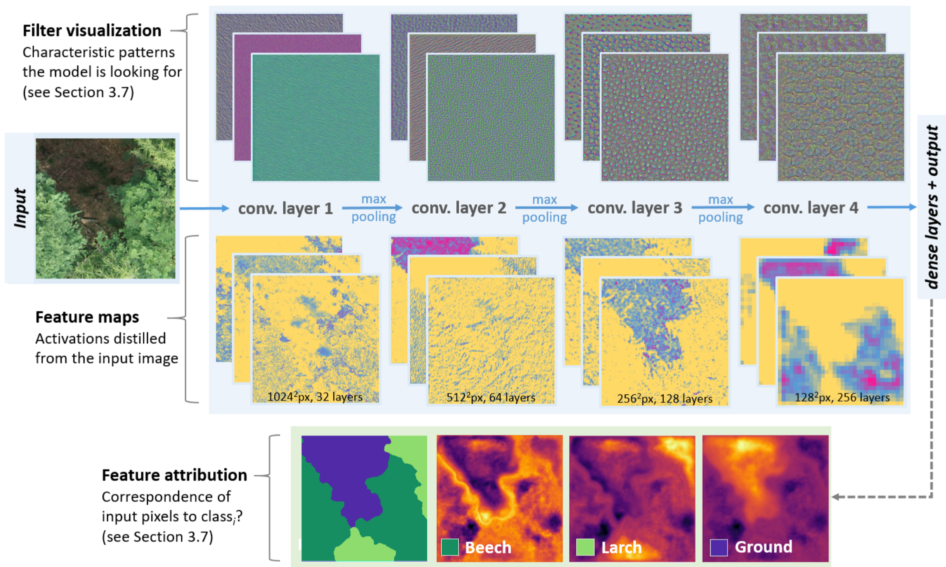


Figure 1: Scheme of a CNN composed of four convolutional layers and subsequent pooling operations trained for tree species classification. The visualization of convolutional filters (top) indicate characteristic patterns the CNN is looking for and were derived by gradient ascent; a technique revealing artificial images maximizing each filter's activation. The feature maps (center) are the dot-product of the preceding layer and individual filters. Feature attribution maps (bottom) can reveal individual pixels that were decisive for the tree species assignment (details on feature attributions 3.6.2).

Between sequences of multiple convolutional layers, the feature maps are commonly spatially down-sampled using spatial **pooling operations** (see Fig. 1). Pooling describes the transformation of multiple cells into one cell, similar to resampling an image to a coarser spatial resolution. Pooling has several advantages: It reduces the data size while preserving discriminant information, which in turn decreases the number of model parameters, thus computational load and the chance of overfitting; and it enables detecting more abstract features as well as spatial context across scales and thereby condenses semantic information. Pooling is defined by a filter size, stride (the distance between consecutive pooling operations), and a reduction operation. The most typical pooling operation is **max-pooling**. The idea of max-pooling (instead of, for instance, average pooling) is that strong activations (e.g. edge or line features) are conserved within the network and not averaged out. A typical max-pooling operation with 2-by-2 filter size and a stride of 2 reduces the size of the input feature map by a factor of 4, whereas the output cells contain the maximum value of the 4 input cells within the 2-by-2 filter.

The layers of CNNs, e.g. convolutional or pooling layers, can be combined in very different ways - commonly described as the CNN **architecture**. CNNs can, hence, have very different architectures, which are basically defined by the task. The task can be the classification of images, the segmentation of multiple classes, or the localization of individual objects within a scene (presented in more depth in chapter 3.2.2). The suitability of a CNN architecture largely depends on the complexity of the task: A more complex problem usually requires a deeper and more sophisticated network. In contrast, limited availability of training data constrains model complexity due to an increased risk of overfitting. The complexity and general performance of a CNN architecture further depends on the **hyper-parameters**, which define amongst others the number and characteristics of hidden layers, pooling operations, regularization techniques, or cost-functions. Accordingly, there exists a wide array of options to implement a CNN towards the specific use case as well as predefined and established architectures. Examples are given in the literature review in chapter 3.2. Comprehensive overview of different architectures is given in Hoeser et al. (2020) and Zhu et al. (2017).

2.2 Why CNN for vegetation remote sensing?

The physiology and morphology of vascular plant canopies is primarily optimized towards the absorption of solar energy using the photosynthetic machinery and the corresponding assimilation of carbon for maintenance, further growth, and reproduction. Despite these common goals among vascular plants, plant life can differ greatly on multiple scales, ranging from various morphological features of the individual, including leaf tissue properties, leaf form, branching patterns, canopy structure, and the general habitus, to large-scale patterns of vegetation communities. Furthermore, anthropogenic land use can determine spatial vegetation patterns, either through indirect influences on floral vitality and species composition or directly through economic activities. Examples include dendritic or fish bone-like deforestation structures in rain forests, crop rows on plantations, or directed changes in species composition as a result of gradual nutrient inputs from agricultural land.

Remote sensing offers several sensors and acquisitions techniques that are sensitive to physiological and morphological properties of vegetation and, hence, allow for spatial representations of vegetation patterns at the scale from plant organs to entire landscapes. This includes close-range observations from terrestrial platforms (e.g., farming robots), fine-resolution data from airborne platforms (UAV or airplanes) as well as more coarser-resolution satellite-based acquisitions that are usually focused on large-scale applications.

So why are CNNs suitable for vegetation analysis with such remote sensing data? CNNs are indeed a revolutionary technique but they do not do magic, meaning that they cannot reveal more information than is contained in the data. The crucial advancement of CNNs is *how* they can extract information from spatial data. Previous parametric or machine learning methods applied in vegetation remote sensing usually required feature engineering, i.e. the careful screening of redundancies in the input data and the extraction of latent variables that best describe the response variable. Simply put, the model needs to be taught how to *see* the relevant features before it can start solving the problem. Feature engineering is, hence, based on an understanding of a system and its processes. This enables to control the model with pre-knowledge but is certainly limited in case of unknown systems that potentially inherit many dimensions and complex interactions. Especially for the analysis of 2D or 3D patterns, there are a plethora of transformations that can be applied to extract spatial features and textures. Examples are Grey Level Co-Occurrence Matrices (Haralick 1979), Fourier Transformations (Bone et al. 1986), or 3D multi-scale metrics derived from point clouds (Brodu et al. 2012; Weinmann et al. 2015). These numerous types of transformations can moreover be applied with different hyper-parameters (e.g., kernel function or size). The potential amount of latent variables extracted this way explodes, considering that one can extract latent variables with such transformations based on different input data available, e.g., different wavelengths of a multispectral sensors or snapshots from a time series. Thus, identifying the best combination of possible predictors on a heuristically basis is often a very inefficient task and often hardly possible.

In contrast, a CNN itself learns the ability to *see* by iteratively optimizing the transformations, i.e., the convolutional layers, during the training process. This *end-to-end* learning principle can make feature engineering obsolete and, thus, providing the raw data (e.g. spectral bands or the point cloud) can be already sufficient. Additional feature engineering, e.g., transformations like vegetation indices or preprocessing such as speckle reduction, may even introduce an information loss and decrease the model accuracy (Geng et al. 2017; Hartling et al. 2019; Sothe et al. 2020). In contrast to statistical modeling or machine learning, deep learning, hence, shifts the focus from *what* a model should learn to *how* a model should learn. The latter is primarily defined by the model architecture and the optimization of its hyper-parameters as discussed in the following sections.

2.3 The training process

Training a CNN model for vegetation mapping requires the remote sensing data and matching reference annotations, also called labels or targets. While machine learning algorithms, such as random forests or support vector machines, require relatively simple array-type data structures, CNN-based training is performed using more sophisticated data structures called **tensors**. Tensors are essentially stacked arrays that typically have 4 dimensions, including the individual samples, the spatial dimensions (x, y), a feature dimension (z, e.g. intensity or reflectance), and a layer dimension (e.g. the corresponding wavelength).

During training, the CNN weights are optimized for a certain task, e.g., detecting a certain plant species. This detection is realized by transforming the input data through convolutional and other hidden layers while being propagated through the network. The neurons between layers are connected through **activation func-**

tions determining if a neuron is active – also referred to as firing - or not (ReLU, the most frequently used activation function, is described in chapter 3.2.1.1). If activated, the intensity of a neuron's output is determined by its weights and biases. The weights and biases are usually optimized using the **gradient descent** algorithm, which can briefly be described as follows: The term gradient descent implies the progressing minimization (descent) of errors along a slope (gradient). Gradient descent is performed in iterations, in which predictions of a model with momentary parameterization are compared to the annotations of the training data using a **loss function**. The gradients are derived using the **backpropagation** algorithm. Given a neural network with an input layer (a tensor), an output layer (prediction) and n hidden layers in-between (e.g. convolutional layers), the back propagation algorithm calculates the gradient of the loss function with respect to the weights and biases between the hidden layers. This gradient is then used to evaluate and update the model weights and biases through gradient descent, i.e. trying to find a global minimum in the high-dimensional feature space. The gradient descent procedure is performed for multiple samples, followed by averaging the calculated weights and biases of the hidden layers.

Training a CNN is usually computationally very intensive as the explanatory variables, e.g. image data or point cloud representations, are rich in dimensions (geolocation + layers) resulting in a myriad of feature maps that depict different spatial features and context at varying scales. This obviously results in excessive amounts of data to be processed during CNN training – especially considering that model training may require many samples to memorize the decisive features of the target class. These data volumes may, thus, not fit the memory of our system at once. To overcome this, training is often performed sequentially in **batches** comprising only a share of the entire dataset. The model weights and biases are updated based on one average gradient for the entire batch. Separating the dataset into batches enables to train the model iteratively until it has seen all samples, which is called an **epoch**. The number of iterations to finish an epoch is, thus, the total number of observations divided by the batch size.

Generally, it is unlikely that a CNN trained in a single epoch already reaches maximum performance. For instance, observations (in form of batches) that were shown to the model at the beginning of the training phase may be again useful to extract more features at a later stage of the training process. Moreover, multiple steps in the training procedure described above feature stochasticity: The convolutions are based on randomly initialized filters, the assignment of observations into batches is random, and the gradient descent has a random nature (hence, also referred to as *stochastic* gradient descent). For this very reason, CNNs are commonly optimized within a series of subsequent epochs until the model performance stops to advance (the model converges) or even decreases (the model overfits). The number of epochs eventually depends on the complexity of the problem and model structure.

The fact that gradient descent is an iterative algorithm opens several interesting avenues for CNN-based modelling: Firstly, models can be updated with unseen data at any time without training the model again from scratch substantially saving computation loads and processing time. Secondly, models that have seen a lot of data, e.g. from generic image databases such as *ImageNet*, can be shared and optimized for a specific problem (further discussed in Section 3.2.1.2). The third and probably most future-oriented avenue is **federated learning**, which is the training of local models with local data on distributed clients and the simultaneous sharing of weights coordinated by a central server (Bonawitz et al. 2019). The server thereby merges the locally derived gradients without ever seeing the data. Federated Learning follows, thus, the principle of *bringing the code to the data, instead of the data to the code*, which will be inevitable in the geosciences due to constantly growing data streams. Besides reducing communications costs, this approach avoids problems related to data access rights, security, or privacy.

2.4 Implementation, libraries and frameworks

Most deep learning frameworks can be used on standard operating system (Linux-based, Windows, macOS) and provide bindings for different programming languages. Currently, Python is the most common language in DL research. Training and inference of deep learning models consist of millions of simple computations, i.e. multiplications and additions. Thus, it is helpful to use **graphics processing units** (GPU) rather than central processing units (CPU). In contrast to CPUs, GPUs have rather simple cores but thousands of them, which are optimized to handle thousands of concurrent operations, leading to a drastic reduction of time for training and inference. Mostly *NVIDIA* GPU are used, as these feature the *CUDA Deep Neural Network (cuDNN)* library, which is utilized by common DL frameworks. The *cuDNN* library provides highly performant primitives for convolutions, pooling operations, normalization and activation functions. Furthermore, *AMD* provides different tools for deep learning on Linux-based platforms with the *Radeon Open Compute Platform*. In case of missing hardware it is nowadays possible, to use (partially free) cloud platforms with GPU support, such as *Alibaba Cloud*, *Amazon Web Services*, *Microsoft Azure* or *Google Cloud Platforms* such as *Google Earth Engine*. These platforms have completely configured containers for many frameworks. *Google Colab* https://colab.research.google.com even provides free access to (limited) computing resources including GPUs with no setup.

CNNs can be implemented through different **frameworks**. Overviews of former and current frameworks are given in Hoeser et al. (2020), Nguyen et al. (2019) and on the corresponding Wikipedia page (https://en.wikipedia.org/wiki/Comparison_of_deep-learning_software). The currently most prominent deep learning frameworks are **PyTorch** and **Tensorflow** (Nguyen et al. 2019). Both provide high-level APIs (e.g. Keras) and various tools for training, data augmentation, and visualization (e.g. Tensorboard). Furthermore, many vintage and modern DL architectures can be used directly and with pretrained weights. Extensive documentations, many tutorials, and Jupyter notebooks allow an easy start with both open-source frameworks. Additionally, the Open Neural Network Exchange (ONNX) format allows interoperability between many frameworks such as Pytorch, Tensorflow, Keras, mxnet, scikit-learn, Matlab, SAS, and many more. Thus, already implemented and trained models can be transferred to a favored framework. In Section 5 links to various tools, models and quick start tutorials are provided.

3 Literature review on CNN-based vegetation remote sensing

The literature review was based on a survey on *Google Scholar* and the search terms *CNN*, *convolutional neural networks*, *vegetation*, *plants*, *forestry*, *agriculture*, *land cover*, *conservation*, *mapping*, *Remote Sensing*, *RGB multispectral*, *LiDAR TLS*, *ALS*, *SAR*, *RADAR*, *airborne*, *satellite*, *UAV*. The search results were first filtered by the title, by the abstract and then by the content. We only considered primary research articles that underwent a peer-review process. This resulted in a total of 101 research studies considered in the literature review. All studies were published after 2016 and more than 75 % of the studies were published in 2019 or later (see Fig. 2), underlining that CNN-based vegetation remote sensing is a very young but rapidly developing field.

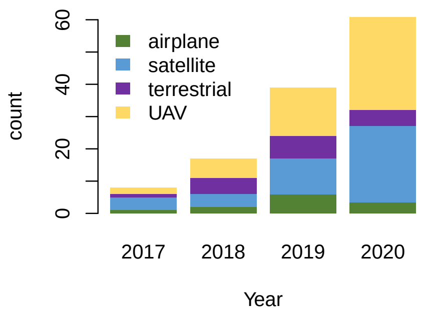


Figure 2: Number of yearly publications based on the literature search indicating a steep increase of studies applying CNNs for vegetation remote sensing. Counts for 2020 were extrapolated based on the number of publications until November.

The resulting literature is very heterogeneous in terms of application areas, vegetation types, target variables, CNN implementations, and remote sensing data (compare Fig. 3). Accordingly, several criteria were defined to structure the literature and identify general trends, including the underlying CNN architecture, remote sensing platform, sensor, spatial resolution of the remote sensing data, mode of reference data acquisition (in-situ or by visual interpretation), number of training and test observations, response type (e.g., object detection or semantic segmentation), geographic location of the study area, accuracy metrics, area of application (agriculture, forestry, conservation or miscellaneous) and specific task (e.g, detecting weed infestation or tree cover mapping). A corresponding spreadsheet including all assessed criteria and studies is available in the Appendix. For the accuracy metrics, we constrained our analysis on the most frequently reported metrics (overall accuracy, precision, recall, F-score and intercept over union). Whenever a study reported multiple accuracy metrics, e.g. when comparing multiple methods, we recorded the best result. The geographic locations of the study areas were derived from place-names using the *Google Geocoding API*, unless the manuscripts explicitly included the longitude and latitude of the study area.

3.1 Reference data

3.1.1 Reference Data Sources

As with any supervised modelling approach, training and validating a CNN requires reference observations, also referred to as annotations, labels, or targets. The large number of parameters in CNNs and the corresponding ability to detect even subtle patterns are associated with the risk in training a model that is based on overly-specific details and does not generalize well - it is overfitting. Accordingly, independent validation of CNNs prior to model deployment is of great importance to evaluate its robustness and transferability. Ideally, such validation should not solely involve iteratively shuffling training and validation data, as frequently done in remote sensing studies (e.g., as with a cross-validation or bootstrapping), but be based on entirely independent data that the model has never seen before. Therefore, most CNN-related studies split their reference data in a 1) training data set, which commonly is split again in **training** and **validation** data during the model training process, and 2) a **testing** data set used to independently evaluate the eventual predictive performance of the final model. Typically, a share of 20 to 30 % of the reference data is used for independent testing (median 21 %).

In the field of remote sensing of vegetation, reference data was most commonly acquired in **ground-based surveys** in the form of in-situ plot or point observations (Fassnacht et al. 2016). The quantity of reference data of ground-based surveys is generally limited as these involve high logistic efforts and costs for transportation, equipment, and personnel. In particular for studies in natural environments, limited accessibility can also greatly hamper the sampling frequency. The effectiveness of ground-based surveys for CNN modelling may, hence, be limited as the latter often requires ample reference data. In particular for complex tasks, such as the differentiation of classes that only differ in subtle features, the quantity of available reference data can be the critical factor for a successful model training and convergence. Moreover, tasks as object detection or the segmentation of individual crown components (3.2.2) require reference data that is spatially explicit and in exact correspondence with the remote sensing data. Especially for analysis of very high spatial resolution remote sensing data at centimetre scale, GNSS-coded reference data acquired in the field is often not directly applicable for two main reasons: Firstly, geolocation errors of GNSS-measurement typically exceed 0.1-1m; particularly under dense vegetation canopies (Branson et al. 2018; Kaartinen et al. 2015; Valbuena et al. 2013). Secondly, for practical reasons, field data is usually measured in form of point observations (e.g., stem position of a tree) or using circular or rectangular plots, which does commonly not allow for a spatially explicit link with remote sensing data (Anderson 2018; Kattenborn et al. 2019d; Leitão et al. 2018). Correspondingly, only 14 % of the studies reviewed here used in-situ data as exclusive reference input.

Instead of using in-situ observations, reference data is most often (62 %) directly acquired in the primary or secondary (e.g., higher resolution) remote sensing data using **visual interpretation**. In contrast to common in-situ point or plot observations, reference data acquired by visual interpretation is commonly spatially explicit as it is directly derived from the imagery or point cloud. Furthermore, there is no position error, as long as the same input data is used for the CNN and visual interpretation. If secondary data (e.g. higher resolution) is used for visual interpretation, the geolocation error is relative to the spatial agreement of primary and secondary data. Visual interpretation provides a very efficient mode of generating reference data, given that the variable of interest is clearly identifiable in the imagery. Accordingly, this mode of reference data acquisition is in particular applicable for discrete classes (e.g. species, plant communities, crop or vegetation types, individuals). The term visual *interpretation* implies a rather imprecise capture of the target metric, but it should be noted that in-situ observations do not necessarily represent (ground) *truth*: As with visual image interpretation, mapping species in the field is commonly based on visual interpretation and, hence, can also be prone to errors and bias (Lepš et al. 1992; Lunetta et al. 1991).

Annotations from visual interpretation are often derived by delineating target classes in a GIS environment. This includes the identification of individuals by points, as often performed for image-based object detection in agricultural environments (Csillik et al. 2018; Freudenberg et al. 2019), or by delineating the vegetation components (e.g. in form of polygons) for semantic or instance segmentation (Flood et al. 2019; Kattenborn et al. 2019a). Many studies have also used special interfaces for an efficient labeling such as *RectLabel*, *LabelMe*, *Labelbox* or *LableImg* (Russell et al. 2008). Instead of manually labeling the spatial extent of target classes, a semi-automatic approach using a prior segmentation may be used. For instance, dos Santos Ferreira et al. (2017) automatically segmented canopy components in RGB imagery of soybean fields using SLIC (Simple Linear Iterative Clustering) superpixels (Achanta et al. 2012), and assigned each segment to weeds or crops by visual interpretation. Natesan et al. (2019) labeled segments derived from a watershed-based segmentation using a Digital Surface model. In particular, for LiDAR-based point cloud data, region growing algorithms may be used to efficiently segment points belonging to individual plants (Wang et al. 2019) or plant components (e.g. stems, branches or foliage; Z. Xi et al. 2018).

Despite the above-mentioned advantages, obtaining reference data by visual interpretation does not rule out misinterpretation. Yet, at the example of mapping plant species, it has been shown that CNNs can to some extent compensate flawed or noisy labels (Hamdi et al. 2019; Kattenborn et al. 2020).

Although in-situ data may not be the ideal for training and validating CNNs, it may be an essential requirement in case the target class (e.g. species) is not readily identifiable in the remote sensing data by means of visual interpretation alone. According to our review, 22 % of the studies that acquired reference data by visual interpretation also incorporated in-situ data for training or validation. 84 % of these studies were either related to forestry or conservation tasks and thus to rather complex environments, in which visual interpretation alone may not be sufficient. For instance, Schiefer et al. (2020) and Kattenborn et al. (2019a) used ground-based full inventory data as a basis to annotate tree species in UAV imagery in temperate forests forests in Germany, and in highly heterogeneous and complex natural forests in Waitutu, New Zealand, respectively. Similarly, Sun et al. (2019) used in-situ data on tree species to map the species diversity in tropical wetlands. Field data may also provide an independent source to validate CNN-based predictions (Flood et al. 2019). Especially in cases when a bias by visual interpretation is assumed, a validation using in-situ reference data is highly recommended.

Visual interpretation may be more efficient for data annotation than using in-situ data alone, but even human labeling through visual interpretation can be very tedious, especially for large datasets or complex vegetation canopies that require very detailed annotations. The effort of annotating data may be reduced by specific training strategies, such as **weakly-** or **semi-supervised learning** (see section 3.2.1.3), that compensate for few or coarse annotations. Alternatively, if no knowledge of a vegetation expert is required, crowdsourcing can be used for labeling. Commercial services are now also available for this purpose. For example, Branson et al. (2018) used the service *Amazon Mechanical Turk* ™to locate individual trees in *Google Street View* imagery.

Although visual interpretation is an effective labeling approach to many tasks, it should be noted that there are many vegetation-related applications where it is not applicable. Particularly, for continuous quantities, such as crop yield or forest biomass (Ayrey et al. 2018; Castro et al. 2020; Yang et al. 2019), reference data acquisition is conceptually more difficult as these are often not directly measurable from the remote sensing data. Here, in-situ measurements or other physically-based retrieval procedures may often present the only applicable solution. A physically-based retrieval of reference data was presented by Du et al. (2020), who aimed at mapping wetland inundation extent in forests on large spatial scales with satellite data (WorldView-2). For parts of their study area, LiDAR data was available enabling accurate detection of surface waters due to its strong absorption in near-infrared wavelengths. Reference data acquisition on yield or biomass in an agricultural context may be automatized by integrating measurement devices on harvesting machines. For instance, Nevavuori et al. (2019) trained a CNN to predict wheat and malting barley yield from UAV imagery using training data derived from a yield measurement device (*John Deere Greenstar 1*) that was coupled with a GNSS receiver and mounted on a harvester.

Concerning biochemical and structural plant traits, an interesting approach is to train CNNs with simulated data derived from physically-based models. Such hybrid approaches, i.e. coupling statistical and process-based models, may not only provide data for training but also enable including priors and realistic constrains in model training (Reichstein et al. 2019). For instance, Annala et al. (2020) trained a 1D-CNN with reflectance spectra simulated with the radiative transfer model (RTM) SLOP (Maier et al. 1999). Although SLOP is a relatively simple leaf reflectance model, Annala et al. (2020) demonstrated promising tests of this hybrid inversion method for UAV hyperspectral acquisitions of forest canopies. More sophisticated RTMs may allow to produce more robust models, e.g. *PROSAIL* (Jacquemoud et al. 2009) enabling to account for bidirectional reflectance effects in plant canopies, whereas 3D-RTMs such as *FLIGHT* (North 1996) or *DART* (Gastellu-Etchegorry et al. 1996) may provide interesting sources for generating synthetic training data for 2D-CNNs (see Section 3.2 for details on 1D-, 2D- and 3D-CNNs).

3.1.2 Reference data quantity

The quantity of reference data required for the convergence of a CNN depends particularly on the complexity of the algorithm and most importantly on the contrast of the features that are decisive for the vegetation property of interest. Fewer reference data may be required if the vegetation property of interest is easily identifiable in the remote sensing data (e.g., due to a distinct canopy structure or contrasting flowers). Subtle differences and complex relationships in turn require more complex algorithms and more samples to identify the relevant features. Accordingly, the effects of varying the training data size cannot be generalized. The results of Weinstein et al. (2020) suggest that the accuracy first increases rapidly with increasing the reference data quantity and then stagnates. In the context of tree species mapping in urban environments, Hartling et al. (2019) showed that using 10 % of their available training samples decreased the overall accuracy from 82.58 % to 70.77 %. Using 200-3940 samples and multiple CNN architectures, Fromm et al. (2019) showed that the reference data quantity can have a large influence on the overall accuracy for tree seedling mapping (up to 18 %). Using UAV data for segmenting growth forms in wetlands, T. Liu et al. (2018b) demonstrated that the effect of sample size (700-3500 samples) can greatly differ across different model architectures and complexities.

Overall, the amount of reference data used in the reviewed studies differed greatly - most notably between studies using different remote sensing platforms. Studies based on terrestrial data acquisitions, e.g., terrestrial or mobile LiDAR scanning, used around 340 reference observations (median). UAV- or airborne-related studies used a median of 2795 reference observations and studies based on satellite observations 6001 observations. These large differences may be the result of two factors: Firstly, studies at the satellite-scale typically cover larger spatial extents and are, hence, more likely to benefit from previously acquired reference data sets (cf. Schmitt et al. (2020)), whereas, the coarser spatial resolutions also allow to incorporate reference data with higher geolocation errors. Secondly, data acquired at higher resolutions, often TLS or MLS LiDAR data, contains finer information on vegetation structures and may thus include more characteristic features. This may hence facilitate model convergence and decreases the amount of reference data required.

A common training strategy that aims to compensate for few reference data is **data augmentation**, which inflates the number of reference data by introducing small manipulations to the existing data or creating synthetic data (see details section 3.2.1.1). Instead of collecting new reference data, it may be more efficient to use existing reference data, e.g. from previous research projects or authorities (e.g. environmental agencies, forestry offices). Accordingly, the establishment of open access **databases** incorporating labeled remote sensing data is increasingly demanded but still lacking (Zhu et al. 2017). Such databases would not only facilitate the efficiency of model training due to ample training data, but would also allow to assess and improve the extrapolation and transferability of these models to new domains. This is particularly important as geoscientific models are often under-constrained due to limited representatives of the training data (Reichstein et al. 2019). Accordingly, databases can enable to test and improve the model transferability towards new domains, such as different remote sensing acquisitions (daytime or sensors), vegetation types, or growth stages. Moreover, databases of sufficient size could also play an important role to develop backbones that are specifically oriented to vegetation remote sensing (further discussed in Section 3.2.1.2). Freely accessible databases can also facilitate more comprehensive and universal comparisons of algorithms and the identification of improvement opportunities.

Despite the described benefits, there exist still only a few databases providing labeled remote sensing data, which may be explained by the novelty of the scientific field (cf. Fig., 2), associated costs for data storing and sharing (especially in regard to high-resolution data), various fields of application with individual annotation requirements, and lastly the diversity in remote sensing sensors, acquisition and processing modes. A prime example is the voluntarily organized *ImageCLEF* initiative (imageclef.org). The latter hosts an evaluation platform and mostly annually recurring competitions for cross-language annotation of images (Kelly et al. 2019). The first competition was hosted in 2003 and aimed at classifications of generic photograph datasets, whereas in 2011 the first vegetation-specific competition followed, which was centered on plant species identification from ordinary photographs. Since 2017, *ImageCLEF* also hosts the *GeoCLEF* competitions, which focus on plant species identification by means of environmental and remote sensing data, including high-resolution remote sensing imagery and respective land cover products. Another example is the *NSF NEON* database (Kampe et al. 2010; Kao et al. 2012; Marconi et al. 2019) including a wide array of (partly multitemporal) reference and remote sensing data (most importantly from RGB, LiDAR, hyperspectral airborne campaigns) on natural and semi-natural ecosystems. This database has already been proven to be of immense value to train and validate models across ecosystems and remote sensing acquisitions (Ayrey et al. 2018; Weinstein et al. 2020). For instance, Weinstein et al. (2020) tested cross flight performance of a CNN for tree crown segmentation in different environments. Their results underlined the value of large databases for model training, as the model generalization with additional datasets greatly improved - even when the target class was not present in all datasets. Example centered on developing and benchmarking deep learning towards vegetation types and land-cover mapping with Sentinel-2 imagery are the *SEN12MS* (Schmitt et al. 2019), *BigEarthNet* (Sumbul et al. 2019) and *EUROSAT* (Helber et al. 2019) datasets. In the agricultural context, the *Global Wheat Dataset* (global-wheat.com) includes standardized images on weeds (1024×1024 pixels) with subcentimetre resolution, providing the basis for public challenges, such as the 2020 challenge to count wheat ears (David et al. 2020).

An alternative approach could also be the use of databases that only refer to vegetation information but can be linked to existing remote sensing data in other ways, e.g. by taxonomic identities or geo-coordinates. Valuable resources in this context are the *TRY* database (try-db.org, kattge2020try), which contains a wealth of morphological, physiological and phenological plant traits, the *opentrees* database (opentrees.org, providing species and location information of individual trees in urban areas, or *GBIF* (gbif.org, providing several huge datasets on citizen-science-based plant photographs together with species names and geo-coordinates, including the popular *iNaturlist* dataset.

3.2 Common CNN approaches and architectures

3.2.1 Training strategies

Training a CNN can be challenging due to a restricted amount of labeled observations, computation load required for model convergence, and model overfitting. This chapter lists the most common strategies and methods applied during training to alleviate these challenges.

3.2.1.1 Normalization and regularization techniques

A famous problem in training artificial neural networks with gradient-based learning is the **vanishing** or **exploding gradient problem** (Hochreiter 1991, 1998). During backpropagation, the weights of each node are updated proportionally to its gradient in respect to the loss. The gradients are derived by calculating the derivative of an activation function. For a common sigmoid function, this derivative becomes increasingly small for very low or high values. The derivative of a layer is calculated by the chain rule and so gradients and corresponding updates of weights in earlier layers of the network can approach zero (vanish). The opposite effect, i.e. exploding gradients, can occur for large derivatives. This imbalance in the network ultimately impairs the network's ability to find the ideal updates for the weights.

A common counter-measure is **batch normalization**, which is applied in 26 % of the reviewed studies, particularly in networks with many parameters such as for semantic segmentations (Kattenborn et al. 2019a; Ronneberger et al. 2015; F. Wagner et al. 2019). Batch normalization normalizes the output of activation functions to zero-mean and unit variance and thereby prevents the network from becoming imbalanced due to excessively high or low activations. This smooths the optimization problem of the gradient descent function and allows for larger ranges of learning rates and hence facilitates network convergence.

The vanishing gradient problem can also be greatly reduced by using the **Rectified Linear Unit** (**ReLU**) activation function. The output weight of the ReLU function equals the weighted sum of the inputs as long as this sum is > 0 (values < 0 are ignored). For > 0, ReLU is a simple linear function such that the derivative is always 1, hence, preventing the vanishing gradient problem. The probably more important characteristics of ReLU are its non-linearity and its **regularization** function of the network. The large amount of parameters in deep networks makes them prone to overfitting and, therefore, regularization aims to facilitate a network's ability to generalize. ReLU regularizes the network by reducing the parameters of the model as it ignores values < 0 - these values are in theory not activated anyway. The reduction of parameters also greatly decreases the computing time in contrast to conventional hyperbolic tangent functions (Krizhevsky et al. 2012). Only few studies reported that they used other activation functions suggesting that in fact most of the studies used ReLU.

One of the most common and effective regularization technique is **Dropout**(Srivastava et al. 2014) (used in at least 31 % of the reviewed studies), and stands for randomly removing a fraction (typically 50 %) of a layer's output features during the training process (these output features are set to zero). The core idea of dropout is to artificially introduce stochasticity to the training process preventing the model from learning statistical noise in the data.

Still, overfitting does not only depend on the number of parameters in the model, but also on the representatives of the sampling. Particularly in the context of vegetation mapping, samples are often taken under limited conditions, while a model is deployed to further, foreign conditions. The associated risk is therefore an over-fitting of the model to the situation with limited conditions and representatives (e.g, with regard to scene illumination or local vegetation properties). An obvious solution is a larger amount of training data or covered variation, respectively. To reduce the costs of creating labeled observations, a commonly applied procedure is to synthetically increase the sample quantity and diversity using

data augmentation procedures (Chatfield et al. 2014; Krizhevsky et al. 2012). Data augmentation is the process of producing more samples from existing data by introducing manipulations them (Shorten et al. 2019). These changes may include randomly changing the spatial extent of the imagery, e.g., to make a model more robust for detecting individuals of a plant species with varied sizes. Random transformations, such as flipping, rotating or translating the imagery, can increase the generality towards varying sun-azimuth angles and corresponding cast-shadows (also described as rotational invariance). Random spectral shifts may compensate for variation in illuminations caused by topography or atmospheric conditions and may further alleviate data calibration issues or sensor-specific differences. In most CNN-related studies using LiDAR data, the detection process is not based on the point cloud, but 2D projections derived from the point cloud (cf. Section 3.5.2). Here, data augmentation can be performed by varying the viewing geometry prior to generating the 2D image – also referred to as multi-view-data generation (Jin et al. 2018; Ko et al. 2018; Su et al. 2015; Zou et al. 2017). The overall effectiveness of data augmentation is highlighted by the fact that 47 % of the studies used data augmentation. Fromm et al. (2019) and Safonova et al. (2019) explicitly tested the effect of data augmentation and found significant improvements for the detection of tree seedlings and bark beetle-infected trees, respectively.

Data augmentation may also be performed by not introducing minor manipulations, but creating new, synthetic observations from the existing data. Gao et al. (2020) presented an automated procedure for the creation of synthetic images and labels from original images for detecting weed infestation (*Calystegia sepium*) in sugar beet fields. Their approach involved the creation of masks for individual plants from the original images used for cropping and transferring the corresponding RGB information to other base images. Adding data created from this (very simply said copy & paste) approach to the original training data indeed increased the precision from 0.75 to 0.83. For training a CNN for detecting individual tree crowns, Braga et al. (2020) used the same principle and created synthetic Worldview-3 observations by randomly placing manually-delineated tree crowns on background tiles.

Probably the most elegant framework for generating synthetic data is Generative Adversarial Networks (GANs). Inspired by game theory, GANs are driven by the competition of a generator module, creating synthetic data (e.g., images) and a discriminator module aiming to disambiguate between synthetic and real data Frid-Adar et al. (2018) and Goodfellow et al. (2014). During training, a GAN, hence, simultaneously improves on how to synthesize observations from noise and how to classify them (synthetic vs real data or further classes). At the example of segmenting weed infestation in crop fields in UAV imagery, Kerdegari et al. (2019) demonstrated a GAN architecture, composed of a generator and the discriminator modules with four convolutional layers each. The proposed GAN produced realistic synthetic visual and near-infrared scenes. Moreover, it was demonstrated that using the discriminator module for semantic segmentation of unknown images resulted in comparable accuracy to a pure CNN - even when using only 50 % of the available labels. The fact that the discriminator was originally trained to detect another problem, i.e. differentiating synthetic from real data, suggest that applying this trained discriminator to real world problems could also be considered as a form of transfer learning - an approach discussed in more detailed in the next chapter.

3.2.1.2 Transfer learning and backbones

As described earlier, training data for vegetation attributes is often limited as its acquisition is commonly costly and limited by accessibility. Furthermore, the training itself is often associated with high computing costs.

A common practice to alleviate this problem is to apply **transfer learning** during CNN model training. Transfer learning includes **pre-training** of the CNN model on other, presumably very large and heterogeneous datasets. Such datasets do not necessarily have to include the target metric or class (e.g. a certain plant species) and can, for instance, be derived from public and generic databases. Popular examples are the image databases *MSCOCO* or *ImageNet*, which contain thousands of images from various objects, such as cars, buildings, or people. A very elegant approach of transfer learning is to built on pre-trained models directly, commonly referred to as **pre-trained backbone**, which can potentially reduce data storage and processing costs.

The principle of transfer learning can be transcribed as the process where very generic images, not necessarily belonging to vegetation-related situations, are used to teach the CNN the ability to *see* in a general sense. The subsequent step of adjusting the network can be understood as teaching the CNN how to apply the ability to see to a very specific problem, such as the differentiation of certain plant species.

There exist various transfer learning approaches (Pires de Lima et al. 2020; Too et al. 2019; Tuia et al. 2016), which can be roughly grouped into two primary strategies: The shallow strategy adopts very general, lower-level image features such as edge detectors from the pre-trained backbone or the generic training dataset. Only the last layers of the CNN are then fine-tuned for higher level and task-specific features using imagery corresponding to the specific problem (e.g. plant species detection). The deep strategy, in contrast, involves fine-tuning the entire network, i.e. start back-propagation with all layers on the pre-trained network.

The use of pre-trained backbones is restricted to available architectures. Yet, backbones can be customized with output layers (e.g. to apply it on regression or classification problems), cost functions, and other components or integrated in existing CNNs. There exist a variety of backbones for popular CNN architectures (cf. Section 3.2.2), such as *VGG*, *ResNet* or Inception. It should be noted that the popular backbones are usually trained on 3-channel (RGB) data, whereas remote sensing information often provides more predictors, such as multiple bands, time steps, or sensor types. In this case, band selection or feature reduction algorithms provide a promising avenue (Rezaee et al. 2018).

According to our review, 30.5 % used pre-trained backbones (e.g., Brahimi et al. (2018), Branson et al. (2018), Fromm et al. (2019), Gao et al. (2020), Mahdianpari et al. (2018), and Rezaee et al. (2018)). Mehdipour Ghazi et al. (2017) compared the utility of three backbones based on *GoogLeNet*, *AlexNet*, *VGGNet*, to identify plant species in photographs. Brahimi et al. (2018) assessed the value of pre-training for plant disease recognition based on RGB imagery and multiple CNN architectures. They showed deep pre-training strategy, i.e. back-propagation on all layers of the pre-trained model, delivered the highest accuracy. The shallow strategy was usually worse than training a model from scratch. Fromm et al. (2019) showed that pre-training not always significantly improved the detection of tree seedlings and that the value of pre-training depends on the network's complexity, while more shallow architectures are less likely to benefit from pre-training. Mahdianpari et al.

(2018) report that full training resulted in better accuracy than fine-tuning existing backbones trained on *ImageNet*. This suggests that the detection of vegetation patterns may not necessarily benefit from features learned on generic datasets. This also agrees with recent research by He et al. (2018) suggesting that transfer learning may indeed be useful if training data is scarce and computation power limited, but otherwise an exhaustive training on task-specific data will result in higher accuracy than using generic datasets.

3.2.1.3 Weakly- and semi-supervised learning

Besides a lack of reference data, it may occur that reference data already exist, but do not meet the ideal requirements for the intended application. Accordingly, several concepts and strategies have evolved to compensate for limited availability or conceptual incompatibilities of reference data.

The aim of **Weakly supervised learning** is to decrease costs for human labeling or to make use of existing, lower quality reference data. This concept is particularly interesting for semantic segmentation tasks, where usually an annotation for each sample (point or pixel) is required. Weakly supervised-learning can, for instance, involve annotations at an image level instead of at a pixel level, or sparsely annotated data at a pixel level, such as bounding boxes, lines, or points. Adhikari et al. (2019) applied weakly supervised learning using the principle of semantic graphics to map crop rows and individual weed plants in rice paddies. Semantic graphics defines target objects or concepts through abstract forms. Accordingly, Adhikari et al. (2019) defined crop rows as line features and weeds as solid circles and showed that an encoder-decoder CNN is capable of accurately learning and mapping these concepts. Their findings are particularly interesting because plant rows are rather fuzzy and not clearly delimitable. The higher-level concept of a row, however, is clearly definable for humans by abstracting the spatial context of the individual plants and obviously also reproducible by CNNs. The concept of weakly supervised learning is also applicable when explicit 'ground truth' is scarce but frequent datasets from other studies exist that come with their own errors or lower spatial resolutions. Promising results of this approach were presented by Schmitt et al. (2020), who predicted vegetation types with Sentinel data and used training data derived from MODIS land cover maps at 500m resolution (this dataset is freely available; SEN12MS, Schmitt et al. (2019)). Using a high resolution imagery, they demonstrated that the Sentinel-based predictions reached even higher accuracy than the datasets used for training. Another variant of weakly supervised learning for semantic segmentation is based on saliency maps. The basis for this approach is a CNN trained for image classification, which can be analyzed through class activation mapping (cf. Section 3.6.2 and Fig. 1 showing an example for tree species) to identify those pixels that are decisive for assigning an image$_j$ to a class$_i$) These pixels are then used to segment the target class$_i$ based on the assumption that these pixels highlight the components of the respective class in the image$_j$ (e.g., the canopy of a tree species). Although no study has been published to date that has applied this approach to vegetation remote sensing, the potential has been demonstrated several times in other disciplines (Lee et al. 2019; K. Li et al. 2018). This approach could, hence, provide a promising way for an efficient and automatic segmentation (e.g., of plant species) based on large image databases without spatially explicit labels, such as the *iNaturalist* data.

Semi-supervised learning describes the training of a model with only a small number of reference data and, hence, can be located between supervised and unsupervised learning. Weinstein et al. (2019) applied semi-supervised learning framework for detecting single tree crowns in airborne imagery using a two-step approach: The first step, which can be considered as unsupervised or weakly-supervised learning, involved training a CNN with labels (bounding boxes, n = 435,551) derived automatically from LiDAR data and a tree crown segmentation algorithm (Roussel et al. 2017). In the second step, the CNN was optimized using a few hand-annotated samples derived from the airborne imagery (n = 2,848). Thereby, Weinstein et al. (2019) demonstrated that only few high-quality samples may be required for training a robust CNN.

However, the number of samples required for a specific task is difficult to estimate in advance. In this regard, **Active learning**, which can be considered as a special case of supervised learning, can be an efficient solution. Active Learning describes the iterative optimization of a model by repeatedly adding new reference data until the predictive accuracy saturates or reaches a desired threshold. Ghosal et al. (2019) exemplified an active learning approach for sorghum head detection in UAV imagery. Starting point was a single image together bounding boxes of sorghum heads to train a CNN, which was then applied to another random image. The image and predictions were afterward fed into an annotation app in which a human interpreter corrected the predictions before they were added to the training dataset. The initial model was then optimized using the enlarged training dataset and the entire procedure was repeated in multiple iterations. In their case study, the model accuracy already converged between 5-10 iterations, highlighting the efficiency of active learning for finding the right balance between costs of human labeling and model performance.

3.2.2 Approaches and architectures

Depending on the components and architecture, CNNs can be implemented in many different ways, which in turn enables a wide range of different applications in the field of vegetation remote sensing. CNNs can initially be grouped into **1D-**, **2D-** and **3D-CNNs**, where the number refers to the dimensions of the kernel. 1D-CNNs are less often used (8 % of the reviewed studies) since they do not explicitly consider spatial context and are, hence, primarily applied to analyze optical spectra or multitemporal data (Annala et al. 2020; Guidici et al. 2017; Kussul et al. 2017; Liao et al. 2020; Y. Xi et al. 2019; Zhong et al. 2019). Most studies applied 2D-CNNs (88 %), as these readily exploit spatial patterns in common imagery (e.g., RGB or multispectral imagery, cf. Fromm et al. (2019), Kattenborn et al. (2020), Milioto et al. (2017), Neupane et al. (2019), F. H. Wagner et al. (2020), and Weinstein et al. (2019). The added value of spatial patterns, i.e. of 2D versus 1D-CNNs, was even demonstrated with relatively coarse-resolution Landsat data (Kussul et al. 2017). 3D-CNNs are rarely used (4 %), but are the means of choice when successive layers have a directional relationship to be considered (e.g. canopy height profiles, hyperspectral reflectance, or time-series data, e.g., Ayrey et al. (2018), Barbosa et al. (2020), Jin et al. (2019), Liao et al. (2020), Lottes et al. (2018), Nezami et al. (2020), and Zhong et al. (2019)). 2D- and 3D-CNNs can be applied to solve different problems, including assigning values or classes to entire images, detecting individual objects within images, segmenting the extent of classes, or simultaneously detecting individual objects and segmenting their extent (Fig. 10b). The major differences, including the required structure of labels and resulting outputs, are described in the following sections:

3.2.2.1 Image classification / regression

Image classification is the assignment of a class to an entire image (Fig. 9a). For example, an image may be assigned to the class *shrub* if at least a fraction is covered with *Ulex europaeus* or *Sambucus nigra*. Training image classification or regression-based CNNs requires comparably simple annotations in the form of class correspondences or continuous values, respectively, for each image. Typical CNN-architectures for image classification and regression include *VGG*, *ResNet*, *Inception* or *EfficientNet*. *VGG* uses blocks of consecutive convolutions and non-linear activations. Between those building-blocks max-pooling with stride of 2 reduces the resolution of the layers. The filter size of the convolution is restricted to 3x3, leading to less parameters and thus more possible layers. The small filter size is still common in more recent networks. Finally, some fully connected layers are added for classifying the output of the building-blocks (Fig. 4). ResNet also consists of building-blocks with consecutive convolutions and activations (Fig. 5) but with some major difference: First, the depth of the layers is drastically reduced before the 3x3 convolution with a bottleneck 1x1 convolution. Thus, the number of parameters is much lower compared to *VGG*, even so *ResNet* has up to 10 times more layers. Second, to compensate for the *vanishing gradient problem* (cf. Section 3.2.1.1) with such a high number of layers (e.g. 152), skip connection with identity or convolution shortcuts are introduced. Such skip connections are still used in the current design, allowing very deep networks. Third, *ResNet* only uses one max pooling layer. Instead, convolution with stride 2 are used for resolution reduction. Most modern architectures such as *EfficientNet* also dismiss max-pooling operation to reduce possible information loss during pooling.

A typical procedure to map vegetation patterns in remote sensing imagery with CNN-based image classification or regression is to subset the original imagery into regular tiles (e.g., 128 x 128 pixels) on which the model is subsequently applied (details see Section 3.5.1). This procedure was for instance applied to LiDAR and airborne imagery to map tree species (Sun et al. 2019) or the detection of forest types using a combination of high-resolution satellite imagery and LiDAR data (C. Sothe et al. 2020). Image classification or regression may also be applied to segments derived from previously applied unsupervised image segmentation methods (dos Santos Ferreira et al. 2017; Hartling et al. 2019; Ko et al. 2018; T. Liu et al. 2018a). Image regression is used when a continuous quantity is assigned to an entire tile. For example, (Kattenborn et al. 2020) predicted continuous cover values [%] of plant species and communities in UAV-based tiles (2-5m) along smooth vegetation gradients. Yang et al. (2019) and Castro et al. (2020) estimated rice grain yield and forage biomass in pastures, respectively, from UAV-based tiles. Barbosa et al. (2020) mapped continuous crop yield on coarser scales based on satellite data. Ayrey et al. (2018) used regression on airborne LiDAR data to predict forest biomass and tree density.

3.2.2.2 Object detection

Object detection aims at locating individual occurrences of a class (e.g. trees) within an image (Fig. 9b). The detection typically includes the localization of the object center and an approximation of its extent using a simple rectangular bounding box.

Widely applied architectures for object detection are region-based CNNs (*R-CNN*, Girshick et al. (2014)), which involve a two-step approach; region proposals of the object's location and extent followed by a classification. *R-CNN* was followed by two successors, i.e. *Fast R-CNN* (Girshick 2015) and the most widely applied and efficient *Faster R-CNN* (Ren et al. 2017). The more recent *Faster-R-CNN* forwards feature maps (often derived using a *VGG*-type backbone) to a region proposal branch that performs an initial prediction on potential object locations (also referred to as anchors). These rather rough region proposals are then used to crop areas of the feature maps as input for a fine-scaled object localization and classification (Fig. 6).

Object detection is suitable for countable things with definable spatial extent within the field of view. Such conditions are often found in agricultural settings and accordingly 45 % of the studies related to agriculture apply object detection techniques, such as locating and counting palm or tree individuals in plantations (Csillik et al. 2018; Freudenberg et al. 2019), individual maize plants in TLS-point-clouds of crop fields (Jin et al. 2018) or individual strawberry fruits and flowers in sub-centimeter UAV-imagery (Chen et al. 2019). The application of object detection in natural environments is less frequent, which can be explained by the presence of continuous gradients and smooth transitions in species cover, traits, and communities. In forestry or conservation, only 14 % and 10 % of the studies used object detection. Examples include the localization of fir trees infested by bark beetle (Safonova et al. 2019), the mapping of individual tree crowns across several ecosystems (Weinstein et al. 2020) or the detection of *Cactae* (López-Jiménez et al. 2019).

Object detection-based CNNs are typically trained using bounding boxes of desired classes as labels. Several tools exists for a fast annotation of bounding boxes (see Section 3.1.1). However, a problem with bounding boxes in vegetation analysis is that they often do not explicitly define vegetation boundaries (vegetation is not rectangular). This in turn can make validation difficult, as inaccurate reference data do not allow a final assessment of the prediction (Weinstein et al. 2020, 2019). From this point of view semantic (Section 3.2.2.3) or instance segmentation (Section 3.2.2.4) may be more spatially explicit, but also require more sophisticated annotations.

3.2.2.3 Semantic segmentation

While image classification and object detection aim to detect the presence or location of an object, semantic segmentation aims to delineate the explicit spatial extent of the target class within the image (Fig. 9c). In contrast to object detection, semantic segmentation assigns all pixels in an image to a class. It is especially suited to segment uncountable and amorphous *stuff* (frequently used term to illustrate the contrast to countable *things* (cf. Kirillov et al. (2019)). The training process is typically based on labels in the form of spatially explicit masks to provide a class assignment for each single pixel (e.g., absence or presence or species a, b, c).

The challenge with semantic segmentation is that CNNs usually include multiple pooling operations to reveal spatial context in the feature maps derived from the convolutions and, thereby, spatial reference and detail is initially lost. One solution often referred to as **patch-based**, is to perform a semantic segmentation by predicting only values for the center pixel of the input image and iteratively slide the field of view over the image data until every pixel received a label (Baeta et al. 2017; Fricker et al. 2019; Kussul et al. 2017; Mahdianpari et al. 2018; Rezaee et al. 2018; M. Zhang et al. 2018). However, this method requires an individual prediction for each pixel and is rather inefficient considering that the CNN analyses the neighbouring pixels at the same time anyway. A more elegant and effective way is to build a semantic segmentation on **fully convolutional networks** (FCN) as first demonstrated by Long et al. (2015). FCN conserve the spatial reference, by memorizing the pixels that caused activations in earlier stages of the network and forwarding it to an output segmentation map (see Fig. 7). This way, FCN do not only allow detecting the presence of a target class within an image (e.g., a species) but also the individual pixels that correspond to the target class. A more recent and frequently applied architecture for semantic segmentation is the *U-Net* (named after its 'U'-like shape, Ronneberger et al. (2015)). *U-Net* features encoder-decoder structure, while the spatial scale is subsequently reduced after consecutive pooling operations and again increased in a contracting path (see Fig. 8). The activations from the contracting path are forwarded using skip connections to the expanding path to reconstruct the spatial identity. Further commonly applied CNN-architectures for semantic segmentation are *SegNet* (Badrinarayanan et al. 2017) or *FC-DenseNet* (Jégou et al. 2017). Semantic segmentation is widely used in several contexts, ranging from mapping of plant species (Fricker et al. 2019) and plant communities (Kattenborn et al. 2019a; F. Wagner et al. 2019), to mapping deadwood (Fricker et al. 2019; Jiang et al. 2019). Torres et al. (2020) compared amongst other architectures *U-Net*, *SegNet*, *FC-DenseNet* for mapping *Dipteryx alata* trees in an urban context. Their results suggest that the segmentation accuracy of the three latter algorithms was quite similar, whereas it was found that more simpler architectures (e.g., *U-net*) require less effort for model training.

3.2.2.4 Instance segmentation

Instance segmentation aims at detecting individual *things*, such as individual plants or plant elements, and segmenting their spatial extent. Instance segmentation may, hence, be considered as a combination of object detection and semantic segmentation (Fig. 9d). A few studies used CNN-based object detection and subsequently applied segmentation techniques, such as region growing in the case of point cloud data, to detect individuals (Wang et al. 2019). However, here we define instance segmentation as an end-to-end, CNN-based segmentation of individuals. One of the most popular algorithms for instance segmentation is *Mask-R-CNN* (He et al. 2017); a derivative from *R-CNN* described in section 3.2.2.4. Alike *Faster-RCNN*, it comprises a two-step approach, including an initial region proposal followed by the localization and classification of the feature maps, while in the case of *Mask-R-CNN*, the proposed region is subject to a segmentation branch (Fig. 6). Similar to semantic segmentation, fully connected layers are used to create masks at the original resolution of the input imagery. Despite the potential utility of instance segmentation, the literature search only comprised few respective studies; Jin et al.

(2019) used instance segmentation to map individual leaves and stems in maize plants, Braga et al. (2020) delineated individual tree crowns in tropical forests and Chiang et al. (2020) detected individual dead trees. The rare use of instance segmentation could be explained by the more sophisticated collection of reference data, which involves both the identification of individuals and delineating their explicit spatial extent. In an agricultural context, the identification of instances of multiple classes may often not be necessary, as most tasks are situated in mono-cultures. Instance segmentation in a forestry or conservation context may often not be applicable because natural canopies often feature smooth transitions or overlapping crowns.

3.3 Geographic and thematic areas of CNN application

CNN-based vegetation remote sensing has already been applied in many countries (see Fig. 11), whereas a large amount of studies were carried out in Europe, USA, Brazil, and China. The pattern suggests that CNN applications are found in many of the World's biomes and are hence applicable for a wide range of vegetation types and applications.

Our literature survey revealed that CNN-based vegetation remote sensing is applied to a wide spectrum of thematic categories (Fig. 12). A classification of the studies into broad categories showed that 44 % of the studies are related to agriculture, 26 % of the studies have relevance for both conservation and forestry. 8 % and 22 % exclusively tackled research questions for forestry and conservation, respectively. Within these broad categories, the specific tasks are very diverse (the interested reader can find the explicit references of each task in the appendix):

Examples in the context of **agriculture** include the mapping of individual crop fields at regional scales using medium and high-resolution satellite data, e.g. coffee crop fields (Baeta et al. 2017), rice paddies (M. Zhang et al. 2018), safflower, corn, alfalfa, tomatoes, and vineyards (Zhong et al. 2019). Several studies used high-resolution imagery from airborne and satellite platforms to map individual plants in plantations, e.g. citrus trees, palm trees or bananas (Csillik et al. 2018; Freudenberg et al. 2019; W. Li et al. 2017; Mubin et al. 2019; Neupane et al. 2019). Besides detecting individual citrus trees, Ampatzidis et al. (2019) quantified their crown diameter, health status (NDVI-based), and respective canopy gaps in plantation rows. A large share of the studies used imagery with milli- or centimeter pixel size acquired terrestrially or from UAVs. A prime example of such detailed input data is the detection of weed infestations, e.g., in soybean (dos Santos Ferreira et al. 2017) or sugar beet fields (Gao et al. 2020; Milioto et al. 2017; Sa et al. 2018)). Lottes et al. (2018) presented an automatic approach for mapping weed infestation in imagery acquired by a farming robot equipped with a mechanical actuator that can stamp detected weeds into the ground. Adhikari et al. (2019) used subcentimeter imagery to map crop lines of rice plants in paddy fields to aid navigation of weeding robots for the eradication of weeds (*Panicum miliaceum*). Jin et al. (2018) tested the detection and height estimation of individual maize plants. Other studies used high-resolution imagery for yield estimation, e.g., based on counting individual flowers at sub-centimeter resolution as a proxy for strawberries yield (Chen et al. 2019), segmenting sorghum panicles (Malambo et al. 2019) or applying CNN-based regression for rice grain yield estimation (Yang et al. 2019).

In the **forestry** context, most studies use high-resolution data from UAV or airborne platforms. Ayrey et al. (2018) used airborne LiDAR data to map forest biomass and tree density in temperate forests. Weinstein et al. (2020) tested the localization of individual tree crowns (object detection) across ecosystems using airborne data. Braga et al. (2020) used very high-resolution satellite data to delineate individual tree crowns (instance segmentation) in tropical forests. A series of studies dealt with the mapping of tree species or genera in forests (Fricker et al. 2019; Kattenborn et al. 2020; Natesan et al. 2019; Nezami et al. 2020; Pinheiro et al. 2020; Schiefer et al. 2020; Trier et al. 2018; Zou et al. 2017) and urban areas (dos Santos et al. 2019; Hartling et al. 2019; Torres et al. 2020). Fromm et al. (2019) tested the detection of individual conifer seedlings in high resolutions airborne imagery for monitoring of tree regeneration. A substantial interest exists towards assessments of forest damage, e.g., caused by wind throw (Hamdi et al. 2019; Korznikov 2020) or bark beetle infestations (Safonova et al. 2019).

Examples in **conservation** with medium resolution data include the mapping of wetland types at regional scales with multispectral Landsat and polarimetric RADARSAT-2 data (Mahdianpari et al. 2018; Mohammadimanesh et al. 2019; Pouliot et al. 2019). de Bem et al. (2020) mapped deforestation in the Amazon using stacked pairs of Landsat imagery from consecutive years. In the context of dryland mapping program by *FAO* (Food and Agriculture Organization of the United Nations), (Guirado et al. 2020) mapped tree cover (%) using airborne orthoimagery and exemplified that CNN-based mapping outperformed previous assessments by FAO based on photo-interpretation. Examples for mapping at high spatial resolution include the mapping of rainforest types and disturbance (F. Wagner et al. 2019), plant succession stages in a glacier-related chronosequence (Kattenborn et al. 2019a), herbaceous and woody invasive species species in several environments (Kattenborn et al. 2019a; T. Liu et al. 2018c; Qian et al. 2020), shrub cover (Guirado et al. 2017), ecosystem structure-relevant plant communities in the Arctic tundra (Langford et al. 2019) or the rehabilitation of native tussock grass (*Lomandra longifolia*) after weed eradication campaigns (Hamylton et al. 2020).

3.4 Remote sensing platforms

Approximately, 17 % of the studies acquired data from the ground or **terrestrial** platforms, including stationary photography (Ma et al. 2019), mobile mapping data from *Google Street View* (Barbierato et al. 2020; Branson et al. 2018), farming robots (Lottes et al. 2018), and terrestrial laser scanning (e.g., Bingxiao et al. (2020) and Wang et al. (2019)). The major part of studies using terrestrial platforms took place in an agriculture context with a focus on precision farming.

With 36 %, the largest share of studies assessed in this review used data captured from **UAV**. This can be explained as UAV feature two important features; they enable to autonomously acquire spatially continuous data with automated georeferencing - a feature that recently revolutionized possibilities for fast, flexible, repeated, and cost efficient remote sensing data acquisition for vegetation analysis. At the same time, UAV can be operated at low altitudes capturing vegetation canopies with high spatial detail. High-resolution data acquired by UAV and CNN-based pattern analysis provide powerful synergies for spatially continuous vegetation analysis. Due to the inevitable trade-off of spatial resolution and image footprint, a drawback of any high-resolution remote sensing is the limited area coverage decreasing the efficiency for vegetation assessments on large scales.

One approach to overcome this limitation is the spatial up-scaling of UAV-based vegetation maps with satellite data (Kattenborn et al. 2019b), where UAV-based maps are used as a reference for coarse-resolution but large-scale satellite-based predictions.

Depending on the spatial scale of the vegetation analysis and the size of the decisive spatial features, **airplanes** may feature a more efficient compromise between area coverage and resolution. 11 % of the studies in this review used airborne sensors. In addition to increased spatial coverage, an advantage of airplane platforms is their increased potential payload supporting more sophisticated and high-quality sensors. Accordingly, a large proportion of airplane-related studies used LiDAR or hyperspectral data or a combination of both.

Aerial data from UAV and airplanes are often generated by matching single frames from imaging sensors in concert with photogrammetric processing techniques. Due to the relatively low height of both platforms, the single image frames usually feature a substantial variation in viewing geometry and bidirectional reflectance effects. At first sight, this may challenge the retrieval of vegetation characteristics, but as T. Liu et al. (2018a,c) have shown, this variation can also be a valuable source for increasing the amount of training data and generating more robust models. In a case study on mapping vegetation types in UAV imagery, they demonstrated increasing model performance when using a multi-view approach that combined tiles from orthoimagery and the spatially corresponding single image frames.

In total 35 % of the studies used data acquired from **satellites**. The potential of CNN-based pattern recognition combined with the unprecedented amount of high-resolution satellite data was demonstrated by Brandt et al. (2020) who mapped more than 1.8 billion trees across the Sahara and Sahel zone with a mosaic of 11,128 satellite scenes (GeoEye-1, WorldView-2, WorldView-3 and QuickBird-2). This pioneering study suggest how high resolution data from small satellites (weight $<$ 500 kg) and microsatellites (weight $<$ 100 kg) will offer ground braking opportunities for CNN-based vegetation analysis. Examples are the *Planet Labs* constellation of *PlanetScope* data, which image the entire Earth Surface on a daily basis at 3.7 m resolution or *SkySat*, which enable to image targeted areas at 0.72 m resolution. These satellite constellations may provide sufficient spatial detail for various large-scale CNN-based vegetation assessments.

3.5 Sensors, spatial and spectral resolution

CNN are most frequently applied on passive optical sensors (RGB, multispectral, or hyperspectral). Only a few studies (7 %) used products from SAR systems. Passive optical and SAR data are commonly analyzed with raster-based methods and, hence, discussed together in Section 3.5.1. The second-largest share of studies (10 %), incorporated LiDAR data, whereas 3 % used terrestrial LiDAR data, and 7 % used airborne LiDAR. The common methods for the analysis of LiDAR-based point clouds are presented in Section 3.5.2. The fusion of multiple sensor types is discussed in Section 3.5.3.

3.5.1 Passive optical and SAR data analysis

CNNs involve numerous transformations of the input data and the available (mostly GPU-based) memory may, hence, limit the maximum size of the input data. However, raster data, such as airborne or spaceborne acquisitions from passive optical or SAR-sensors, usually feature multiple layers (e.g., bands of different wavelengths or multitemporal data) and can, thus, occupy large data volumes. Moreover, for some CNN approaches, e.g. image classification, it would not be meaningful to make a single prediction for an entire raster, but, instead, make multiple smaller-scaled predictions to reveal the spatial variation within the area covered by the raster. For these reasons, CNN training and inference is not performed on entire rasters but instead on equally sized **tiles** extracted from a raster. The trained CNN can then be used to create spatial maps using a **sliding window** principle. Thereby, the CNN is applied to regularly extracted tiles that have the same size as the tiles used for training.

The most efficient approach is the seamless extraction of tiles without overlap, whereas combining the results of multiple, overlapping tiles may be useful to increase redundancy and compensate for edge effects (Brandt et al. 2020; Du et al. 2020). Similarly, Neupane et al. (2019) showed that combining the tiling results from different orthophotos acquired at multiple resolutions enhances the detection of palm trees. Generally, the tile sized should be maximized as determined by memory capacities, as larger sizes increase the CNN's field of view and, hence, amplifies the available spatial context and thus accuracy of the model. This effect was demonstrated in (Kattenborn et al. 2020), where the accuracy in estimating the cover of plant species and communities from UAV imagery increased considerably from smaller (2m) to larger tile sizes (5m). Likewise, at the example of predicting crop yield from UAV imagery, Nevavuori et al. (2019) demonstrated that larger tile sizes (10, 20, 40m) resulted in more accurate predictions. Especially for very high-resolution data, it should also be considered that increasing the tile size can furthermore decrease the effect spatially inaccurate reference data (e.g., geolocation errors of in-situ data or inaccurately delineated masks or bounding boxes). However, in the case of image regression or classification (Section 3.2.2.1), which results in a single prediction per tile, increasing the tile size decreases the spatial grain of the mapping output (Kattenborn et al. 2020). For segmentation approaches (Section 3.2.2.3), the spatial extent of the input tiles will have no effect on the output resolution. The processing speed of the sliding window approach can be enhanced by first pre-filtering areas of the target raster using a region proposal. For instance, in the context of shrub cover segmentation in arid areas, Guirado et al. (2017) used brightness thresholds and edge-detectors, as these are already a good indicator to show the general occurrence of shrubs.

In addition to the spatial context or tile size, the **spatial resolution** is a decisive factor. The spatial resolution most strongly varies with the remote sensing platform (Fig. 13) and additionally depends on operating altitude and sensor properties. Although CNN applications are designed for pattern analysis, the highest possible resolution will not ultimately be the most operational solution, as higher resolution comes with increased storage and computation loads. In addition, data acquisition at higher spatial resolution leads to smaller area coverage. The ideal spatial resolution is determined by the spatial scale at which the characteristic patterns of the target class or quantity occur. For instance in the context of tree species mapping, Schiefer et al. (2020) showed decreasing the spatial resolution from 2 to 8 cm decreases the accuracy (F-score) by at least 25 %. Fromm et al. (2019) showed that the detection accuracy for tree seedlings based on different UAV-image resolutions (0.3-6.3 cm) can vary up to 20 %. Similarly, Neupane et al. (2019) found a 17% decrease in 17 detection accuracy for banana palms in plantations when decreasing the pixel size from 40 cm to 60 cm. Weinstein et al. (2020) assessed the relationship between object size and spatial resolution the other way around. They did not change the spatial resolution of the remote sensing data, but analyzed different ecosystems with characteristic tree sizes and concluded that treetop detection for small trees (in alpine forests) was the least accurate.

Regarding **spectral resolution** of passive optical sensors, the literature search revealed that with 52 % the largest share of studies used RGB imagery, whereas only 31 % used multispectral (defined as RGB and at least one additional band) and only 9 % used hyperspectral data (defined here as $>$ 20 spectral bands). The fact that multispectral, and hyperspectral data are less frequently used is not a surprise; multispectral and hyperspectral sensors feature larger pixel sizes as narrower spectral bands receive less radiation and given that the amount of radiation received by the sensor must clearly surpass its signal to noise ratio. Accordingly, everything else being equal, multispectral and hyperspecral sensors have a lower spatial resolution than RGB data. As CNNs are particularly designed for pattern analysis RGB data may often be preferred.

Accordingly, the results of several studies suggest, that for many tasks no high-spectral-resolution information may be needed: For instance, Osco et al. (2020) found that counting citrus trees did not clearly improve when combining multispectral with RGB data. Zhao et al. (2019) found no improvement in using multispectral over RGB data for rice damage assessments (rice lodging). Yang et al. (2019) showed that the added value of multispectral on top of RGB information only slightly improved the estimation accuracy of rice grain yield. In the context of tree species classification, Nezami et al. (2020) did not report clear improvements in using UAV-based hyperspectral data over RGB data. Similarly, Kattenborn et al. (2019a) showed that CNN-based species identification is more accurate than a pixel-based hyperspectral classification of plant species (Kattenborn et al. 2019c; Lopatin et al. 2019).

Yet, for several fields of application spectral data may be absolutely necessary. For instance, analysis related to chemical constituents in plant tissue, e.g. as a proxy for plant health status or plant diseases (Zarco-Tejada et al. 2019, 2018) may not be possible without sufficient spectral information as biochemistry particularly changes absorption properties and not patterns.

Finally, it should be noted that high spectral and spatial resolution can also be combined. For example, pan-sharpening algorithms, such as *local-mean variance matching* or *Gramm-Schmidt spectral sharpening*, can be used to sharpen coarser multi-spectral bands with spatially high-resolution imagery. Such pan-sharpening algorithms are often applied to imagery from very high-resolution satellite sensors that feature a panchromatic band, as for instance WorldView, QuickBird, or Pleiades (cf. Braga et al. (2020), Hartling et al. (2019), Korznikov (2020), and W. Li et al. (2017)). Recently, more sophisticated pan-sharpening algorithms based on CNNs were proposed (Masi et al. (2016) and Yuan et al. (2018), see also Section 3.5.3).

SAR backscatter is known to be particularly sensitive to vegetation 3D-structure and therefore has a great potential for differentiating vegetation types and growth forms. The fact that microwaves penetrate clouds makes it especially suitable for extracting continuous temporal features and large scale assessments and. Accordingly, SAR data was most frequently used as input for CNN for land cover and vegetation type mapping "Comparing Deep Learning and Shallow Learning for Large-Scale Wetland Classification in Alberta, Canada" (2019), Liao et al. (2020), and Mohammadimanesh et al. (2019). Although SAR data have been used overall relatively rarely so far in combination with CNNs, it can be assumed that CNNs are excellently suited to unravel the relatively complex SAR signals and will thus play a major role in Earth observation in the long term (see Zhu et al. (2020) for a review on analyzing SAR data with deep learning).

3.5.2 LiDAR-based point cloud analysis

The analysis of spatial point clouds is basically more computationally intensive than for raster data since there is no spatial discretization (and thus no normalization) in cells, which often results in larger data sets and more complex spatial representations. A strategy to increase the processing speed is to run the analysis on subsets of point clouds, for instance, by detecting key features of the target plant, which are then used as seeds to apply region growing algorithms. This approach was for instance applied for detecting individual maize plants *Zea parviglumis* (Jin et al. 2018) and rubber plants *Hevea brasiliensis* (Wang et al. 2019). The most frequently applied strategy to handle point clouds (mostly terrestrial LiDAR) is the conversion to simpler and discrete feature representations prior to the CNN analysis, including 3D voxels or 2D projections (e.g. depth maps) (Jin et al. 2018; Ko et al. 2018; Windrim et al. 2020; Zou et al. 2017).

Voxels are volumetric representations of the point cloud that are defined by regular and non-overlapping 3D cube-like cells. During the conversion of point clouds to voxel datasets, a voxel is created in a delimitable area (x,y,z) if it contains one or a minimum number of points. Voxels can be analyzed in a similar way as multi-layered rasters, where a layer corresponds to an elevation section of the original point cloud. Jin et al. (2019) used a 0.4 cm voxel space with terrestrial LiDAR data to separate leaves and stems from individual maize plants. Ayrey et al. (2018) used $25 \times 25 \times 33cm$ voxels created from airborne LiDAR-based point clouds to map forest properties, whereas each voxel was assigned the number of points it included.

Projections are 2D representations of the point cloud from a certain position (x,y,z) and viewing angle (azimuth, zenith). The projections can be created by different spatial or spectral criteria, e.g. as depth maps, prior extracted 3D-metrics describing the local neighbourhood, intensity, or color information (Jin et al. 2018; Ko et al. 2018; Zou et al. 2017). For airborne LiDAR data, projections are commonly created using nadir view, for instance, to extract digital height models or to extract height percentiles. For terrestrial LiDAR, projections are typically created using oblique viewing angles. The transformation from TLS-based point clouds to depth images (2D) is usually applied multiple times using different viewing geometries, which can be considered as a form of data augmentation (see section 3.2.1.1). For instance, to train a CNN for detecting individual maize plants in TLS point clouds, Jin et al. (2018) created 32 2D-projections with varying oblique angles.

Despite such possibilities to decrease computation load, it has to be considered that projections or voxel representations of the point cloud will result in a loss of the original spatial detail. Therefore, it may be desirable to use end-to-end learning directly with the raw point cloud data as input. Using the raw point clouds instead of voxel or projections may be more computationally demanding but it can be assumed that ongoing developments in processing and algorithms will advance capabilities to harness point clouds directly. Another challenge is that point clouds are unordered sets of vectors (in contrast to elements in raster layers) and their analysis requires a spatial invariance with respect to rotations and translations. A well-known CNN architecture that considers these challenges is *PointNet*, which, hence, enables efficient end-to-end learning on point clouds. The foundation of *PointNet* are symmetric functions to ensure permutation invariance with regard to the unordered input and transforms the data into a canonical feature space to ensure spatial invariance. Even though *PointNet* or similar algorithms have been used comparatively rarely so far, the results are very promising: Jin et al. (2020) applied *PointNet* to detect ground points under dense forest canopies and found greater accuracy than for traditional non-deep learning methods. Briechle et al. (2020) tested *PointNet* to classify temperate tree species in UAV LiDAR data and reported an overall accuracy of up to 90 %. Bingxiao et al. (2020) and Windrim et al. (2020) used modified versions of *PointNet*, which besides point coordinates also considers the LiDAR return intensity, and demonstrated high accuracy in differentiating woody elements and foliage for multiple coniferous and deciduous tree species (up to 93-96 % overall accuracy). The results of the aforementioned studies are especially remarkable, considering that this approach performs a classification at the highest possible detail, i.e. at the level of individual points.

3.5.3 Sensor and data fusion

Multimodal remote sensing analysis or data fusion is the combination of acquisitions of different sensors types (LiDAR, SAR, passive optical). The different characteristics of the sensor types result in different sensitivities towards plant properties: Passive optical data is largely shaped by absorption and scattering properties at the top of the canopy. SAR signals are composed of directional scattering processes originating in a few centimeters or even meters depth in the canopy (depending on the wavelength). LiDAR measures backscattered radiation of commonly very small footprints enabling to look deep into plant canopies. These different sensing modes can hence reveal different plant characteristics and their synergistic use can be used to harness complementary information.

A conceptually rather simple fusion approach is to merge the resulting predictions of multiple, dataset-specific CNNs. This can, for instance, be done by majority voting (Baeta et al. 2017) or by probabilistic approaches, such as Conditional Random Fields (Branson et al. 2018). However, this way only the output space is combined, but not the features contained in the different data sources so that their synergies cannot be directly integrated and exploited. Therefore it is usually more expedient to simultaneously integrate the different data sources in a single neural network - also known as **feature level fusion**. Feature level fusion requires either preprocessing of the data or an adaption of the CNN architectures to comply with different data structures (e.g. point cloud vs. raster data), sensing modalities such (e.g., viewing angles from oblique SAR vs. nadir passive optical acquisitions).

A frequently used approach for feature level fusion is converting and normalizing the spatial dimensions of the different sensor products and a subsequent **stacking to a common tensor**. Based on this tensor, a CNN can be applied to simultaneously extract features from both data sources. This approach is easy to implement and most frequently applied. For instance, Trier et al. (2018) stacked hyperspectral data with normalized Digital Surface Models (also referred to as canopy height model) for classifying tree species. Hartling et al. (2019) stacked LiDAR intensities, hyperspectral and panchromatic bands for tree species classification in urban areas. Prior to applying the CNN, they also used the LiDAR data extract tree crown segments by height. In the context of large scale mapping of vegetation types in the arctic, Langford et al. (2019) stacked multiple satellite products, including high spatial resolution SPOT data, high spectral resolution EO1-Hyperion data and a height model derived from SAR-interferometry. In the context of mapping crop cover types, Liao et al. (2020) stacked multi-temporal polarimetric RADARSAT-2 SAR data with VENμS multispectral data using a 1D-CNN. While multispectral was superior to SAR data, combining multi-temporal SAR data with multispectral data increased the model performance. Kattenborn et al. (2019a, 2020), Nezami et al. (2020), and Sothe et al. (2020) used UAV imagery for mapping plant species and stacked RGB orthoimagery and canopy height models (CHM) derived from photogrammetric processing pipelines. Interestingly, Nezami et al. (2020) found minor improvements when using CHM information for UAV-based tree species classification. (Kattenborn et al. 2020; Sothe et al. 2020) found that CHM information does not significantly improve the accuracy, whereas Kattenborn et al. (2020) suggested that at these high spatial resolutions the information represented by the CHM is already indirectly visible in the orthoimagery itself through shadows and illumination differences. In contrast, at the example of coarser-resolution satellite imagery and forest type classification, C. Sothe et al. (2020) reported that stacking LiDAR-derived canopy height information with pan-sharpened Worldview-2 contributed important information.

Overall, these studies demonstrated that merging the different data sources into a single tensor can potentially facilitate the extraction of complementary signals through convolutions. This approach is easy to implement as it does not require manipulating common CNN structures. However, stacking datasets may not be ideal as the normalization to a common tensor may introduce a critical loss of the original the information, e.g., by converting point clouds to coarse voxels or depth maps (cf. Section 3.5.2), or the viewing geometries and acquisitions modes may not be directly compatible, e.g., oblique SAR vs. nadir optical data. Instead of fusing datasets through a common tensor, it may, therefore, be more advantageous to process the different data sources in parallel branches and perform a **feature concatenation** at a later stage in the network; that is linking the activations or feature maps derived from multiple, sensor- or data-specific CNN. These networks are also referred to as **multi-stream networks**. At the example of mapping rice grain yield from UAV imagery, (Yang et al. 2019) applied a concatenation of feature maps resulting from two CNN branches, namely RGB imagery with high and multispectral imagery with low spatial resolution, respectively. A prime example on how feature concatenation enables to integrate different data types and structures was presented by Branson et al. (2018), who classified tree species in an urban environment by concatenating a branch fed with nadir airborne RGB imagery and a branch fed with multiple *Google Street View* scenes extracted with varying viewing angles and zoom levels. Lottes et al. (2018) used feature concatenations for detecting crop plants and weed infestations in image sequences taken by a farming robot. Their approach takes into account that planting patterns in agricultural fields (e.g. row structures) provide additional spatial information for differentiating crops from weeds. Accordingly, their approach included the parallel segmentation of successive image frames using encoder-decoder CNN structures and the subsequent concatenation of the resulting feature maps.

Barbosa et al. (2020) compared data fusion based on both stacking datasets and feature concatenation for crop yield mapping based on heterogeneous input data, including remote sensing reflectance and elevation data and in-situ maps on nitrogen, seed rate, and soil electroconductivity. They tested multi-stream approaches with branches being concatenated at an early stage and a later stage in the network, that is before and after applying fully connected layers, respectively. The best performance was achieved with a concatenation after fully connected layers, followed by a feature concatenation at an earlier stage in the network. The worst performance was found when stacking all predictors before applying the CNN, which was attributed to a sometimes complex relationship among different input datasets.

Another noteworthy application of multi-stream networks is CNN-based pan-sharpening, i.e. the process of fusing high spectral information from the coarser-resolution bands with high spatial resolution information. Pan-sharpening is frequently applied to data from very high-resolution satellites as these are often equipped with pan-chromatic bands that have wider spectral bandwidths enabling an increased sensitivity for incoming radiance and thus higher spatial resolution than the other bands with narrower bandwidths. The fusion of spatial and spectral information requires the representation of highly complex and non-linear relationships - an application for which CNN are ideally suited (C. Dong et al. 2016; Yuan et al. 2018). A case study on this seminal technique was presented by Brook et al. (2020), who used a multi-scale pan-sharpening algorithm (Yuan et al. 2018) to fuse both multispectral and -temporal information from Sentinel-2 satellite data with the high spatial information from UAV-imagery at the centimetre scale. The corresponding case study demonstrated that this approach can reveal the temporal variation of leaf biochemical status of individual vineyard rows.

It should be noted that multitemporal analysis (e.g., change detection, time series analysis) can also be considered as feature level fusion. As discussed in more detail in Section 3.5.4, multitemporal analysis can be performed using both of the above presented modes, that is **stacking** multidate inputs (de Bem et al. 2020) or **concatenating** them in multiple CNN branches operating in parallel (Branson et al. 2018; Mazzia et al. 2019).

3.5.4 Multi-temporal analysis

Almost all plant life is subject to seasonal variation as a consequence of reoccurring changes of abiotic factors, such as radiation driving photosynthesis, temperature controlling its efficiency or water input providing the primary oxidation source. The seasonal phases or dynamics, also known as phenology, of plants is expressed through biochemical and structural properties which in turn determine how plants are represented in remote sensing data. This implies that temporal variation in plant traits can limit the transferability of our models through time. At the same time, temporal dynamics can also provide essential information for plant characterization, e.g. phenological features such as flowers revealing the taxonomic identity and or the length of the growing season as an essential factor for productivity and yield.

A few studies assessed model performances based on comparing or combining multitemporal datasets. For instance, Ma et al. (2019) assessed the biomass estimation with subcentrimetre imagery in wheat crops across 17 acquisition dates and found a strong variation in accuracy (R^2 0.60-0.89) highlighting that timing can play an important role. Rezaee et al. (2018) successfully tested the transferability of a CNN for wetland segmentation on a *RapidEye* scene that was not included in the training process. Yang et al. (2019) tested the transferability of CNN models across time for rice grain yield estimation, in terms of how good a CNN trained on one or multiple phenological phases is applicable to a phenological phase it has not seen before. As expected, the models became better the more times were considered in the training process. Similarly, M. Zhang et al. (2018) showed that stacking multidate Landsat scenes increased the accuracy of segmenting rice paddies.

In the context of satellite-based land cover classification, (Mazzia et al. 2019) incorporated spatial patterns of temporal dynamics by concatenating the pixel-wise branches of **recurrent neural networks (RNNs)**, followed by the subsequent application of a CNN. RNNs are a type of deep learning approach to analyse recurring patterns and are therefore perfectly suitable for multitemporal remote sensing analysis (Zhong et al. 2019; Zhu et al. 2017). A primary strength of RNNs is their ability to resemble temporal patterns despite the presence of data gaps introduced by missing scenes, cloud cover, snow, or artefacts. Similar to CNN for spatial patterns, RNNs allow for end-to-end analysis of temporal signals and therefore makes a heuristic definition and engineering of temporal or phenological metrics obsolete. Thus, combining CNNs with RNNs enables an end-to-end processing scheme in both the spatial and temporal domain. It can, hence, be assumed that the combination of RNNs and CNNs will be a milestone for vegetation analysis with time series data as for instance derived from satellite constellations (Reichstein et al. 2019).

In contrast to recurring phases, natural disturbances or anthropogenic impacts can also cause acute or gradual, directed changes. Such anomalies in temporal vegetation dynamics may be tracked with **change detection** of remote sensing data. de Bem et al. (2020) stacked pairs of Landsat imagery to track deforestation in the Amazon rainforest. Compared to earlier change detection approaches, which were mostly based on metrics for temporal comparison (e.g., NDVI), the approach used here is simple and flexible as it does not require sophisticated pre-processing, such as the radiometric cross-calibration of the raw data. A disadvantage is the requirement of training data, such as binary classification of changed and stable areas. However, the required number of reference data is not very high as deforestation is typically clearly visible in remote sensing imagery, and often institutional data can be accessed. de Bem et al. (2020). Another change detection approach was presented by Branson et al. (2018), who used multi-date *Google Street View* imagery to detect changes of urban trees. As the viewing geometries are not steady in street view imagery, a pixel exact stacking is not possible and accordingly, they concatenated **Siamese CNNs** fed with images from the different time steps. Siamese CNNs include identical CNNs that operate in parallel branches (Daudt et al. 2018). During training, the weights are shared between the branches, which reduces the number of learnable parameters but most importantly secures that both branches have the same statistics so that their outputs are comparable. The outputs are then concatenated into fully connected layers to classify similarity.

3.6 CNN model assessment, understanding, and interpretation

3.6.1 Numeric evaluation of the predictive performance

The performance of a CNN model can be determined by different metrics that are primarily determined by the model approach (cf. 3.2.2): For CNN-based regressions, the **coefficient of determination** (R2) and the **Root Mean Squared Error** (RMSE) are the means of choice to quantify the correspondence between predictions and reference observations. The majority (91 %) of the studies reviewed here performed classification tasks, which can be evaluated with several metrics (see Tab.1 for the most ones). The most used and intuitive metric is the **overall accuracy** (used in 71 % of the reviewed studies), which quantifies the proportion of correct predictions.

However, the overall accuracy is prone to bias introduced by class imbalance and in such case an accuracy assessment based on **precision**, indicates the performance regarding false positives, and **recall**, sensitive to false negatives, should be preferred. The **F-score** is the harmonic mean of precision and recall and provides a single metric for the overall model performance that is robust for unsymmetrical datasets.

For object detection and instance segmentation, the question is not how well is the average agreement of all predicted pixels, but how accurately are individual objects or segments detected. Here, an F-score may be strongly biased by object size. A metric that is robust against size variation of objects is the **Intersect over Union** (**IoU**), which is the ratio of correctly classified pixels and the total amount of pixels per segment. Note that recall is also known as producer's accuracy or sensitivity, precision as user's accuracy, F-score as dice coefficient, and IoU as Jaccard-index.

Despite the standardization of accuracy measures, there are several issues that constrain a direct comparison between studies. Firstly, it is hard to compare the different approaches, i.e. object detection, semantic segmentation, and instance segmentation, as these differ in dimensions and thematic complexity. Secondly, the mode of reference data acquisition and quality may greatly constrain the informative value of accuracy assessments (cf. in-situ vs. visual interpretation in Section 3.2.2). Thirdly, the remote sensing data and the site characteristics may differ considerably among studies. For instance, (Weinstein et al. 2020) demonstrated with multiple datasets from the *NEON* project that the detection accuracy of individual tree crowns in airborne imagery greatly depends on the site conditions, such as tree species composition or crown size distribution. Lastly, albeit a common application task (e.g. tree species classification), the definition of the classification problem and presence of classes among studies may differ, which in turn greatly limits comparison of different mapping methods. For example, the present literature search comprises nine studies on tree species classification, none of which examined the same composition of tree species. Clearly, these challenges for comparing different studies, e.g., in terms of CNN architectures, highlights the need for free accessible datasets for comparative studies (cf. section 4).

Despite the challenges related to comparing the different studies, the literature review revealed unprecedented predictive accuracy of CNN-based vegetation remote sensing approaches (see Fig. 14). For instance, studies that targeted the classification of tree species reported at average an overall accuracy of 89 %. In comparison, a review on tree species classification with a focus on shallower machine learning methods (e.g, Random Forest or Support Vector Machines) by Fassnacht et al. (2016) reported an overall accuracy of 83.5 %. This is particularly interesting, as the reviewed studies in Fassnacht et al. (2016) primarily used sophisticated sensors (e.g., hyperspectral or LiDAR data or their combination), while a large share (43 %) of the CNN-based studies assessed here used merely RGB data. The overall superior performance of CNNs compared to shallower machine learning algorithms was demonstrated in several studies and applications tasks (Ayrey et al. 2018; Barbosa et al. 2020; Briechle et al. 2020; de Bem et al. 2020; L. Dong et al. 2020; dos Santos Ferreira et al. 2017; Guidici et al. 2017; Hartling et al. 2019; Knauer et al. 2019; Liao et al. 2020; T. Liu et al. 2018a,b; Mazzia et al. 2019; Mohammadimanesh et al. 2019; Rezaee et al. 2018; Y. Xi et al. 2019; M. Zhang et al. 2018; Zhong et al. 2019)

3.6.2 Understanding and interpretation: Opening the *black box*

Assessing the functioning of a model is important to compare and improve algorithms, to test causal or physical consistency as well as to trust in and learn from models. Transferred to CNNs, this may involve the identification and visualization of individual pixels, patterns, or even higher-level concepts that contribute to the decision-making process. It is often claimed that deep learning and especially CNN models are a black box and it is difficult to grasp the basis on which a CNN makes a decision (Reichstein et al. 2019). This can be explained as on one hand, many people are not yet familiar with the principle of the still quite new CNN algorithms and on the other hand by the incomparable depth and number of parameters of these models. However, most CNNs have a linear and clear structure (mostly consecutive sequences of repetitive structures) and the basic operations, such as pooling or activation functions, are relatively simple. Despite the abundance of parameters, these properties facilitate a converting of abstract vectors into interpretable information and understanding of CNN internal processes. CNN interpretation can be grouped into two branches, i.e. feature visualisation and feature attribution. **Feature visualization** is centered on the model and aims to reveal what the network or parts of it are looking for by simulating synthetic outputs. **Feature attribution** is centered on input data and aims to identify which features in the data activate the network in a particular way.

An example of **feature visualization** for tree species mapping is given in Figure 1, where the functioning of individual convolutions was visualized using gradient ascent-based approach. This technique starts by manipulating a blank image (or any other input format) using the gradient ascent, a function that identifies local maxima so that the values assigned to the output pixels maximizes the activation of the network or a particular layer. The resulting layers, therefore, reflect the patterns that the network has learned as decisive patterns in the training process (see also Schiefer et al. 2020). Feature visualization can hence inform about the general behaviour of the model, whereas this branch of understanding CNNs already offers a variety of different approaches (cf. Olah et al. (2017) for a comprehensive and interactive summary on feature visualization techniques).

A limitation of feature visualization is that the synthetic outputs are often unnatural and abstract and it can be very challenging to link these outputs to real-world features such as plant organs or canopy forms as seen in remote sensing data. Moreover, feature visualization primarily focuses to reveal the general behaviour model of a model, e.g., *what are relevant patterns for separating tree species?*, but a question at hand could be much more specific, such as *On the basis of which plant characteristics visible in the image, did the model distinguish the fir tree from the surrounding spruce trees?*

In this regard, **feature attribution** may enable to analyze CNN models in a more intuitive and traceable way as it is directly based on the input data.

The common products of feature attribution are so-called **activation maps**, also known as sensitivity, saliency, or pixel attribution maps, which typically represent how the input data activates individual feature layers within the network in form of heatmaps (see Fig. 1). Activation maps are obtained by forward propagating individual input images (e.g. through a trained CNN (similar procedures are also applied for point cloud data, cf. (B. Zhang et al. 2019)). Mohammadimanesh et al. (2019) for instance derived activation maps of a CNN for classifying wetland types in order to visualize characteristic backscatter features of different SAR polarization. Moreover, they applied the Uniform Manifold Approximation and Projection (UMAP, McInnes et al. (2018)) algorithm, a non-linear dimension reduction technique, on the activation maps derived from the last layers of multiple CNN architectures to compare their ability to discriminate the wetland types. Despite their demonstrated value, activation maps in their simplest form are only input-specific and not output-specific, so they do not inform how an activation contributes to a decision (e.g. predicting a class affiliation).

An output-specific procedure is given by **gradient weighted class activation mapping** (**Grad-CAM**), which distils class-specific gradients to coarsely localizes the spatial regions of the last convolutional layer that are discriminative towards the network output (Selvaraju et al. 2019). However, tracing class activations to input features can be limited, since common CNNs usually involve several pooling operations so that the last convolutional layer of a network and corresponding activation maps have a much lower spatial resolution than the original input data. A fine-grained representation of decisive image features can be obtained by combining **Grad-CAM** with guided backpropagation, known as **guided Grad-CAM** in case of classifications (Selvaraju et al. 2019), which allows tracing the activation of the last convolutional layer to the individual pixels of the input image (see Fig. 1 for an example on tree species). The feature attribution at the pixel-level can be further enhanced by averaging multiple activation maps generated with stochastic noise, as proposed in the **SmoothGrad** approach (Smilkov et al. 2017). Most approaches for feature attributions target on classification problems, but similar principles were also tested for regression problems, such as regression activation mapping (RAM, Z. Wang et al. (2017)).

Although the above-mentioned methods for CNN interpretation are already established in other scientific fields, their application in vegetation remote sensing seems to be still in its infancy (but see Castro et al. 2020; Schiefer et al. 2020). Nevertheless, according to the demonstrated potential in other disciplines, it can be assumed that feature attribution will play an important role in the future: Feature attribution can be harnessed to test for model shortcomings, such as non-causal relationships and artifacts and as a basis for optimizing CNN architectures and training processes. Moreover, feature attribution provides an interesting avenue for weakly-supervised learning (cf. Section 3.2.1.3), where class activation maps derived from a CNN trained with coarse training data (e.g., presence and absence instead of detailed masks) can be used as a proxy to segment classes at the pixel level (Lee et al. 2019; K. Li et al. 2018). Lastly, it stands to reason that the extraction and preparation of insights from artificial intelligence will increase our knowledge and capabilities towards technical aspects ranging from sensor development and data acquisition, biophysical and ecological understanding, as well as the interrelationship of remote sensing signals and vegetation properties.

4 Concluding remarks and future perspectives

The primary findings of the present review can be summarised as follows:

- The reviewed literature revealed that CNN can greatly advance our capabilities for remote sensing-based vegetation mapping in conservation, agriculture, and forestry sectors. A series of studies reported an increased performance of CNNs over shallower machine learning methods. In addition to high accuracy, CNNs are readily implemented as they support end-to-end learning, enabling immediate use of raw data and, hence, making feature engineering and pre-processing in many cases obsolete. This will greatly facilitate vegetation mapping in the era of Big Data, as the self-learning capabilities will allow to more effectively harness the ever growing data streams across temporal and spatial scales.

- CNNs can be customized for various mapping operations, such as image- or tile-based regression and classification (e.g, yield estimation or absence or presence of a class), segmenting classes (e.g., a plant species or communities), or identifying individual objects and their extents (e.g., single tree of a specific species). Due to phenology and the biochemical and structural diversity of plant life, remote sensing of vegetation benefits from multitemporal and multimodal remote sensing like no other land cover. Combining multiple sensors, perspectives or acquisition dates has often been a technical challenge, whereas the modularity of deep learning frameworks facilitates to combine data with varying dimensions and will, hence, enable to further exploit the diversity of earth observation data.

- The challenges of machine learning were in particular focused on feature engineering (*what should a model see*). The new challenge is to design the learning procedure (*how should a model learn to see*). Designing and implementing an effective CNN architecture requires both technical knowledge on deep learning principles in concert with process-understanding of the system - here, the remotely sensed vegetation signal.

- The core of deep learning, gradient descent is an iterative optimization algorithm and thereby opens efficient, sustainable and elegant ways for model training and exchange, including the subsequent optimization of existing models with new samples instead of training a new model from scratch, the use of backbones to incorporate and channel big data, or federated learning, i.e. the distributed training on multiple clients, to combine computing resources and minimize communication costs (*bringing the code to the data, instead of the data to the code*).

- Exposing CNNs to representative and ample reference data is often a bottleneck for achieving high predictive accuracy and generalization. For reasons of efficiency and data compatibility, ground-based reference data is rarely used, whereas most studies use visual interpretation or the combination of both. Various tools and concepts have been developed to efficiently label remote sensing data using visual interpretation or ancillary data, while concepts such data augmentation, generation of synthetic training data or semi- and weakly supervised learning enable to harness even small quantities or inaccurate training data. It seems obvious that the success of further capturing the seemingly infinite variation of the plant world using deep learning and specifically CNN techniques will be stimulated by free access to remote sensing and reference data and the establishment of corresponding open databases. Pooling ressources in joint databases will foster a sustainable and effective benchmarking of CNN algorithms and building transferable and accurate models.

- Most studies reviewed here were related to classification problems, such as mapping taxonomic identities, land cover types or functional groups. However, many vegetation-related properties are of a continuous nature, for which reference data acquisition is usually quite expensive (e.g., biochemical or structural plant traits). For many tasks, effective CNN-based vegetation remote sensing will require creative approaches that go beyond traditional supervised modelling procedures, including weakly- and semi-supervised learning approaches that link remote sensing observations with non-remote sensing databases (e.g., plant trait observations or forestry variables), with process-based models (e.g., radiative transfer models or forest growth simulators) or incorporate citizen science data (e.g., plant photographs).

- For several vegetation-related applications fields, CNN's strength in exploiting spatial patterns could foster paradigm shifts in the utility of remote sensing sensors and platforms. A series of studies reported success in locating and identifying plant species or individuals by means of simple RGB information and, therefore, highlighted that for a variety of vegetation assessments, where previously expensive and complex sensors seemed necessary (e.g. hyperspectral data), more easily available data can now be sufficient. CNN techniques are, hence, likely to facilitate the realization of cost-efficient and powerful remote sensing solutions for a wide range of users. At the same time, the hunger of CNN for spatial detail is likely to catalyse the utility of high-resolution remote sensing data, in particular microsatellites, off-the-shelf rotary or fixed-wing UAVs as well as terrestrial and airborne LiDAR data.

- Contrary to common preconceptions that CNN models are a *black box*, multiple approaches enable a representation and visualization of a trained model, including its behaviour and the key patterns that contribute to decision making process. The respective feature visualization and attribution methods are essential to understand CNN models and trust them. The greatest chance of these methods, however, lies in distilling new knowledge with regard to the interaction of vegetation and its relationship with remote sensing signals, but particularly towards the diversity of plant form and function.

5 Additional resources on CNN theory, implementation and data sources

Acquire new reference data

• with geocoding in a GIS-environment: *QGIS*(open source, https://qgis.org/) or *ArcGIS*(commercial). ArcGIS supports advanced feature for creating polygons, such as easy tablet and styles support and autocompletion functions.
• without geocoding using annotation tools: *LabelMe* (http://labelme.csail.mit.edu/Release3.0/), *LabelImg* (https://github.com/tzutalin/labelImg), *Labelbox* (https://github.com/labelbox/labelbox)
• *cleanlab*: Machine learning-oriented *Python* package for identifying erroneous labels in datasets and learning with noisy labels (https://github.com/cgnorthcutt/cleanlab)

Use existing reference data

• *NEON*: Partly multitemporal airborne LiDAR, RGB, multi- and hyperspectral acquisitions with in-situ reference data on various ecosystems in the US (https://data.neonscience.org/).
• *EuroSat*: Image patches (64x64 @ 10m resolution) from Sentinel-2 radiance data labelled with vegetation types and land cover classes (https://github.com/phelber/eurosat).
• *BigEarth*: Atmospherically corrected Sentinel-2 patches (120x120 @ 10 m resolution) labelled with CORINE land-cover information (http://bigearth.net/).
• *SEN12MS*: Sentinel-1 and -2 data (256x256 @ 10m resolution) labelled with MODIS-based land-cover information (https://dataserv.ub.tum.de/s/m1474000).
• *Awesome Public Datasets*: List of topic-centric public data sources from the fields of biology, earth sciences, agriculture. https://github.com/awesomedata/awesome-public-datasets

Compensate for few reference data or missing computational ressources

• Use pre-trained backbones: Many predefined architectures with trained weights (e.g., derived from *ImageNet*, *MSCOCO*) can be loaded directly. A tutorial for using pre-trained backbones with *Keras* can be found at https://keras.io/guides/transfer_learning/ and for *PyTorch* at https://pytorch.org/tutorials/beginner/transfer_learning_tutorial.html
• Weakly supervised learning using self organizing maps (SOM, Riese et al. 2020, https://doi.org/10.3390/rs12010007 and code: https://doi.org/10.5281/zenodo.2609130).
• Semi-supervised learning with partially unlabelled datasets presented by *Facebook AI* in a Pytorch tutorial: https://pytorch.org/hub/facebookresearch_semi-supervised-ImageNet1K-models_resnext/

First steps to CNN implementation

• *FastAI*: Initiative aiming at introducing AI principles to a wide audience (slogan: 'Making neural nets uncool again') by maintaining a own *Python*-based library designed for easy implementation and a wide range of material, courses and tutorials (https://www.fast.ai)
• *Keras* Developer Guides, including help and tutorials on the Keras API and getting started with CNN (https://keras.io/guides/).
• The textbooks *Deep Learning with R* and *Deep Learning with Python* by F. Chollet and J.J. Allaire offer a didactically high-quality, catchy and application-oriented introduction to *Keras*, including many hands-on sections and sample codes (ISBN: 9781617295546 and 9781617294433).
• *Deep Learning with Pytorch*: Introduction to the Pytorch framework including a CNN-based image classification example (https://pytorch.org/tutorials/beginner/deep_learning_60min_blitz.html)
• Documentation on CNN-based land-cover classification of Sentinel-2 satellite data, including different training strategies such as fine-tuning and pre-trained networks: https://github.com/jensleitloff/CNN-Sentinel

Discover CNN architectures

• *Model Zoo*: Documentation and tutorials on various CNN implementations for various frameworks (https://modelzoo.co).
• *Papers With Code*: Database on scientific publications together with corresponding data and executable code (https://paperswithcode.com/).
• *Keras* examples for CNN: https://keras.io/examples/vision/
• *Segmentation Models library*: High-level *Python* API including multiple segmentation model architectures and backbones for Keras and Tensorflow (https://github.com/qubvel/segmentation_models/).
• *Awesome Semantic Segmentation*: Links list for the most frequently used segmentation (e.g. U-net) and instance segmentation models (e.g. Mask-R-CNN) for various frameworks. The linklist also includes several annotations tools, datasets and additional resources (https://github.com/mrgloom/awesome-semantic-segmentation/).
• *PyTorch Hub*: Out-of-box models with pretrained weights for PyTorch (https://pytorch.org/hub/).
• *PyTorch Ecosystem Tools*: Tools, libraries, and more for PyTorch, such as fast.ai or Detectron2 (https://pytorch.org/ecosystem/).
• *TensorFlow Hub* (https://tfhub.dev/) and *TensorFlow Model Garden* (https://github.com/tensorflow/models) with hundreds of different (pretrained) models .

Feature visualization and attribution (What did the CNN learn?)

• Comprehensive and interactive resource on principles and approaches for CNN feature visualizations of imagery https://distill.pub/2017/feature-visualization/
• *Interpretable Machine Learning* (Molnar 2019): Constantly updated online book providing background and guides for making machine learning decisions interpretable, including a chapter on CNN-based feature visualization (https://christophm.github.io/interpretable-ml-book/).

- Tutorial on visualizing activation maps with Keras: https://keras.io/examples/vision/visualizing_what_convnets_learn/
- Tutorial on creating saliency maps with the Grad-CAM approach: https://keras.io/examples/vision/grad_cam/
- *Uniform Manifold Approximation and Projection* (UMAP): A dimension reduction technique useful for deriving abstract representations of feature maps of a CNN to visualize the input data structure or exploring classification and regression performance. https://umap-learn.readthedocs.io/en/latest/
- *The What-If Tool* (WIT): Provides an plugins and web interfaces for expanding understanding of a machine learning models allowing the interactive manipulation of labels and models and comparing resulting outcomes (https://github.com/pair-code/what-if-tool).

Acknowledgements

The authors thank Etienne Laliberté, Fabian Fassnacht, Felix Leidinger, and Marco Körner for valuable comments and discussions regarding the manuscript.

Appendix

Methodology of the cluster analysis of terms found in the review literature using *VOSviewer*

The cluster analysis was performed using VOSviewer (Van Eck et al. (2010), version 1.6.14) and based on the frequency of terms contained in title and abstracts. Terms similar in content, synonyms and generic terms to be excluded that are not specifically related to the topic were defined in a thesaurus file. The remaining terms were included in the cluster analysis if they occurred at least five times. As normalization method the *LinLog modularity* was used. The minimum cluster size was set to 10.

Data on the reviewed literature

The data extracted from the reviewed literature is available as spreadsheet under the following URL:
https://tinyurl.com/kattenborn-cnn-meta
(link to Google Drive; the host/URL will be changed in case of acceptance)

Commonly used accuracy metrics for classification and object detection purposes.

Information on the inception module

References

Achanta, R., Shaji, A., Smith, K., Lucchi, A., Fua, P., & Süsstrunk, S. (2012). SLIC superpixels compared to state-of-the-art superpixel

Table 1: Overview and brief introduction of the most frequently used accuracy metrics for classification and object detection purposes.

Metric	Unit	Description / formula
Overall Accuracy (OA)	$[0-1]$	The overall accuracy is the ratio of true predictions (positive and negative) and the total number of observations $$OA = \frac{TP + TN}{TP + TN + FP + FN}$$
Precision (also known as user's accuracy)	$[0-1]$	Ratio of true presences classified correctly and the number of all positive predictions. Precision assesses how many of the predicted presences are actually true. $$precision_i = \frac{TP_i}{TP_i + FP_i}$$
Recall (also known as producer's accuracy or sensitivity)	$[0-1]$	Ratio of true presences classified correctly as i and the total number of instances belonging to class i (true positive and false negative). Recall assess how many of the actual presences were classified as true. $$recall_i = \frac{TP_i}{TP_i + FN_i}$$
F-score (also known as Sørensen-Dice coefficient or Dice similarity coefficient)	$[0-1]$	The F-score is the harmonic mean of recall and precision and, thus, provides a balanced accuracy metric that is sensitive to both under- and overestimation. $$F_i = 2 \times \frac{precision_i \times recall_i}{precision_i + recall_i}$$
Intersection over Union (IoU, also known as Jaccard Index)	$[0-1]$	IoU is closely related to the F-score. IoU measures the relative spatial agreement between reference and predicted surfaces (e.g. a segment or bounding box). The intersect is the area shared among both surfaces (Reference AND prediction), whereas the union is the combined area (Reference OR prediction). $$IoU_k = \frac{TP_k}{TP_k + FN_k + FP_k}$$

methods. *IEEE Transactions on Pattern Analysis and Machine Intelligence*. https://doi.org/10.1109/TPAMI.2012.120 (cit. on p. 13)

Adam, E., Mutanga, O., & Rugege, D. (2010). Multispectral and hyperspectral remote sensing for identification and mapping of wetland vegetation: A review. *Wetlands Ecology and Management*, *18*(3), 281–296 (cit. on p. 4).

Adhikari, S. P., Yang, H., & Kim, H. (2019). Learning Semantic Graphics Using Convolutional Encoder-Decoder Network for Autonomous Weeding in Paddy. *Frontiers in Plant Science*. https://doi.org/10.3389/fpls.2019.01404 (cit. on pp. 20, 25)

Ampatzidis, Y., & Partel, V. (2019). UAV-based high throughput phenotyping in citrus utilizing multispectral imaging and artificial intelligence. *Remote Sensing*, *11*(4). https://doi.org/10.3390/rs11040410 (cit. on p. 25)

Anderson, C. B. (2018). Biodiversity monitoring, earth observations and the ecology of scale. *Ecology Letters*. https://doi.org/10.1111/ele.13106 (cit. on p. 12)

Angermueller, C., Pärnamaa, T., Parts, L., & Stegle, O. (2016). Deep learning for computational biology. *Molecular Systems Biology*. https://doi.org/10.15252/msb.20156651 (cit. on p. 4)

Annala, L., Honkavaara, E., Tuominen, S., & Pölönen, I. (2020). Chlorophyll concentration retrieval by training convolutional neural network for stochastic model of leaf optical properties (SLOP) inversion. *Remote Sensing*, *12*(2), 1–22. https://doi.org/10.3390/rs12020283 (cit. on pp. 14, 21)

Atzberger, C., Darvishzadeh, R., Schlerf, M., & Le Maire, G. (2013). Suitability and adaptation of PROSAIL radiative transfer model for hyperspectral grassland studies. *Remote Sensing Letters*, *4*(1), 56–65. https://doi.org/10.1080/2150704X.2012.689115 (cit. on p. 3)

Ayrey, E., & Hayes, D. J. (2018). The use of three-dimensional convolutional neural networks to interpret LiDAR for forest inventory. *Remote Sensing*, *10*(4), 1–16. https://doi.org/10.3390/rs10040649 (cit. on pp. 14, 16, 21, 22, 26, 30, 36)

Badrinarayanan, V., Kendall, A., & Cipolla, R. (2017). SegNet: A Deep Convolutional Encoder-Decoder Architecture for Image Segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, arXiv 1511.00561. https://doi.org/10.1109/TPAMI.2016.2644615 (cit. on p. 24)

Baeta, R., Nogueira, K., Menotti, D., & Dos Santos, J. A. (2017). Learning Deep Features on Multiple Scales for Coffee Crop Recognition. *Proceedings - 30th Conference on Graphics, Patterns and Images, SIBGRAPI 2017*, 262–268. https://doi.org/10.1109/SIBGRAPI.2017.41 (cit. on pp. 24, 25, 31)

Barbierato, E., Bernetti, I., Capecchi, I., & Saragosa, C. (2020). Integrating Remote Sensing and Street View Images to Quantify Urban Forest Ecosystem Services. *Remote Sensing*, *12*(2), 329. https://doi.org/10.3390/rs12020329 (cit. on p. 26)

Barbosa, A., Trevisan, R., Hovakimyan, N., & Martin, N. F. (2020). Modeling yield response to crop management using convolutional neural networks. *Computers and Electronics in Agriculture*, *170*, 105197 (cit. on pp. 21, 22, 33, 36).

Bingxiao, W., Wu, B., Zheng, G., & Chen, Y. (2020). An Improved Convolution Neural Network-Based Model for Classifying Foliage and Woody Components from Terrestrial Laser Scanning Data. *Remote Sensing*, *12*, 1010. https://doi.org/10.3390/rs12061010 (cit. on pp. 26, 31)

Bonawitz, K., Eichner, H., Grieskamp, W., Huba, D., Ingerman, A., Ivanov, V., Kiddon, C., Konecny, J., Mazzocchi, S., McMahan, H. B. Et al. (2019). Towards federated learning at scale: System design. *arXiv preprint arXiv:1902.01046* (cit. on p. 10).

Bone, D. J., Bachor, H.-A., & Sandeman, R. J. (1986). Fringe-pattern analysis using a 2-D Fourier transform. *Applied Optics*. https://doi.org/10.1364/ao.25.001653 (cit. on p. 8)

Braga, J. R. G., Peripato, V., Dalagnol, R., Ferreira, M. P., Tarabalka, Y., Aragão, L. E. O. C., & Velho, H. F. D. C. (2020). Tree Crown Delineation Algorithm Based on a Convolutional Neural Network. *Remote Sensing*, *12*, 1288. https://doi.org/10.3390/rs12081288 (cit. on pp. 18, 25, 26, 29)

Brahimi, M., Arsenovic, M., Laraba, S., Sladojevic, S., Boukhalfa, K., & Moussaoui, A. (2018). *Deep Learning for Plant Diseases: Detection and Saliency Map Visualisation*. Springer International Publishing. https://doi.org/10.1007/978-3-319-90403-0. (Cit. on p. 19)

Brandt, M., Tucker, C. J., Kariryaa, A., Rasmussen, K., Abel, C., Small, J., Chave, J., Rasmussen, L. V., Hiernaux, P., Diouf, A. A., Kergoat, L., Mertz, O., Igel, C., Gieseke, F., Schöning, J., Li, S., Melocik, K., Meyer, J., Sinno, S., ... Fensholt, R. (2020). An unexpectedly large count of trees in the West African Sahara and Sahel. *Nature*, *5563*(August 2019). https://doi.org/10.1038/s41586-020-2824-5 (cit. on pp. 27, 28)

Branson, S., Wegner, J. D., Hall, D., Lang, N., Schindler, K., & Perona, P. (2018). From Google Maps to a fine-grained catalog of street trees. *ISPRS Journal of Photogrammetry and Remote Sensing*, *135*, 13–30. https://doi.org/10.1016/j.isprsjprs.2017.11.008 (cit. on pp. 12, 14, 19, 26, 31–34)

Briechle, S., Krzystek, P., & Vosselman, G. (2020). Classification of Tree Species and Standing Dead Trees by Fusing Uav-Based Lidar Data and Multispectral Imagery in the 3D Deep Neural Network Pointnet++. *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, *5*(2), 203–210. https://doi.org/10.5194/isprs-annals-V-2-2020-203-2020 (cit. on pp. 31, 36)

Brodrick, P. G., Davies, A. B., & Asner, G. P. (2019). Uncovering Ecological Patterns with Convolutional Neural Networks. *Trends in Ecology and Evolution*, *20*, 1–12. https://doi.org/10.1016/j.tree.2019.03.006 (cit. on p. 4)

Brodu, N., & Lague, D. (2012). 3D terrestrial lidar data classification of complex natural scenes using a multi-scale dimensionality criterion: Applications in geomorphology. *ISPRS Journal of Photogrammetry and Remote Sensing*, *68*, 121–134. https://doi.org/10.1016/j.isprsjprs.2012.01.006 (cit. on p. 8)

Brook, A., De Micco, V., Battipaglia, G., Erbaggio, A., Ludeno, G., Catapano, I., & Bonfante, A. (2020). A smart multiple spatial and temporal resolution system to support precision agriculture from satellite images: Proof of concept on Aglianico vineyard. *Remote Sensing of Environment*, *240*(January), 111679. https://doi.org/10.1016/j.rse.2020.111679 (cit. on p. 33)

Cadieu, C. F., Hong, H., Yamins, D. L., Pinto, N., Ardila, D., Solomon, E. A., Majaj, N. J., & DiCarlo, J. J. (2014). Deep Neural Networks Rival the Representation of Primate IT Cortex for Core Visual Object Recognition. *PLoS Computational Biology*. https://doi.org/10.1371/journal.pcbi.1003963 (cit. on p. 4)

Castro, W., Junior, J. M., Polidoro, C., Osco, L. P., Gonçalves, W., Rodrigues, L., Santos, M., Jank, L., Barrios, S., Valle, C., Simeão, R., Carromeu, C., Silveira, E., Jorge, L. A. d. C., & Matsubara, E. (2020). Deep learning applied to phenotyping of biomass in forages with uav-based rgb imagery. *Sensors (Switzerland)*, *20*(17), 1–18. https://doi.org/10.3390/s20174802 (cit. on pp. 14, 22, 37)

Chatfield, K., Simonyan, K., Vedaldi, A., & Zisserman, A. (2014). Return of the devil in the details: Delving deep into convolutional nets, In *Bmvc 2014 - proceedings of the british machine vision conference 2014*. https://doi.org/10.5244/c.28.6. (Cit. on p. 18)

Chen, Y., Lee, W. S., Gan, H., Peres, N., Fraisse, C., Zhang, Y., & He, Y. (2019). Strawberry yield prediction based on a deep neural network using high-resolution aerial orthoimages. *Remote Sensing*, *11*(13), 1–21. https://doi.org/10.3390/rs11131584 (cit. on pp. 23, 25)

Chiang, C. Y., Barnes, C., Angelov, P., & Jiang, R. (2020). Deep Learning-Based Automated Forest Health Diagnosis from Aerial Images. *IEEE Access*, *8*arXiv 2010.08437, 144064–144076. https://doi.org/10.1109/ACCESS.2020.3012417 (cit. on p. 25)

Chollet, F. (2017). Xception: Deep Learning with Depthwise Separable Convolutions. *Proceedings of the IEEE conference on computer vision* *and pattern recognition*, *7*(3), 1251–1258. https://doi.org/10.4271/2014-01-0975 (cit. on p. 3)

Colomina, I., & Molina, P. (2014). Unmanned aerial systems for photogrammetry and remote sensing: A review. https://doi.org/10.1016/j.isprsjprs.2014.02.013. (Cit. on p. 3)

Comparing Deep Learning and Shallow Learning for Large-Scale Wetland Classification in Alberta, Canada. (2019). *Remote Sensing*, *12*(1), 2. https://doi.org/10.3390/rs12010002 (cit. on p. 30)

Csillik, O., Cherbini, J., Johnson, R., Lyons, A., & Kelly, M. (2018). Identification of Citrus Trees from Unmanned Aerial Vehicle Imagery Using Convolutional Neural Networks. *Drones*, *2*(4), 39. https://doi.org/10.3390/drones2040039 (cit. on pp. 13, 23, 25)

Daudt, R. C., Le Saux, B., & Boulch, A. (2018). Fully convolutional siamese networks for change detection, In *2018 25th ieee international conference on image processing (icip)*. IEEE. (Cit. on p. 34).

David, E., Madec, S., Sadeghi-Tehran, P., Aasen, H., Zheng, B., Liu, S., Kirchgessner, N., Ishikawa, G., Nagasawa, K., Badhon, M. A. Et al. (2020). Global wheat head detection (gwhd) dataset: A large and diverse dataset of high resolution rgb labelled images to develop and benchmark wheat head detection methods. *arXiv preprint arXiv:2005.02162* (cit. on p. 16).

de Bem, P. P., de Carvalho Junior, O. A., Fontes Guimarães, R., & Trancoso Gomes, R. A. (2020). Change Detection of Deforestation in the Brazilian Amazon Using Landsat Data and Convolutional Neural Networks. *Remote Sensing*, *12*(6), 901. https://doi.org/10.3390/rs12060901 (cit. on pp. 26, 33, 34, 36)

Dong, C., Loy, C. C., & Tang, X. (2016). Accelerating the super-resolution convolutional neural network. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, *9906 LNCS*arXiv 1608.00367, 391–407. https://doi.org/10.1007/978-3-319-46475-6_25 (cit. on p. 33)

Dong, L., Du, H., Han, N., Li, X., Zhu, D., Mao, F., Zhang, M., Zheng, J., Liu, H., Huang, Z., & He, S. (2020). Application of convolutional neural network on lei bamboo above-ground-biomass (AGB) estimation using Worldview-2. *Remote Sensing*, *12*(6), 958. https://doi.org/10.3390/rs12060958 (cit. on p. 36)

dos Santos Ferreira, A., Matte Freitas, D., Gonçalves da Silva, G., Pistori, H., & Theophilo Folhes, M. (2017). Weed detection in soybean crops using ConvNets. *Computers and Electronics in Agriculture*, *143*(February), 314–324. https://doi.org/10.1016/j.compag.2017.10.027 (cit. on pp. 13, 22, 25, 36)

dos Santos, A. A., Marcato Junior, J., Araújo, M. S., Di Martini, D. R., Tetila, E. C., Siqueira, H. L., Aoki, C., Eltner, A., Matsubara, E. T., Pistori, H., Feitosa, R. Q., Liesenberg, V., & Gonçalves, W. N.

(2019). Assessment of CNN-Based Methods for Individual Tree Detection on Images Captured by RGB Cameras Attached to UAVs. *Sensors*, *19*(16), 3595. https://doi.org/10.3390/s19163595 (cit. on p. 26)

Du, L., McCarty, G. W., Zhang, X., Lang, M. W., Vanderhoof, M. K., Li, X., Huang, C., Lee, S., & Zou, Z. (2020). Mapping forested wetland inundation in the delmarva peninsula, USA using deep convolutional neural networks. *Remote Sensing*, *12*(4), 644. https://doi.org/10.3390/rs12040644 (cit. on pp. 14, 28)

Fassnacht, F. E., Latifi, H., Stere??czak, K., Modzelewska, A., Lefsky, M., Waser, L. T., Straub, C., & Ghosh, A. (2016). Review of studies on tree species classification from remotely sensed data. *Remote Sensing of Environment*, *186*arXiv arXiv:1011.1669v3, 64–87. https://doi.org/10.1016/j.rse.2016.08.013 (cit. on pp. 3, 12, 36)

Flood, N., Watson, F., & Collett, L. (2019). Using a U-net convolutional neural network to map woody vegetation extent from high resolution satellite imagery across Queensland, Australia. *International Journal of Applied Earth Observation and Geoinformation*, *82*(June), 101897. https://doi.org/10.1016/j.jag.2019.101897 (cit. on p. 13)

Freudenberg, M., Nölke, N., Agostini, A., Urban, K., Wörgötter, F., & Kleinn, C. (2019). Large scale palm tree detection in high resolution satellite images using U-Net. *Remote Sensing*, *11*(3), 1–18. https://doi.org/10.3390/rs11030312 (cit. on pp. 13, 23, 25)

Fricker, G. A., Ventura, J. D., Wolf, J. A., North, M. P., Davis, F. W., & Franklin, J. (2019). A convolutional neural network classifier identifies tree species in mixed-conifer forest from hyperspectral imagery. *Remote Sensing*, *11*(19), 2326 (cit. on pp. 24, 26).

Frid-Adar, M., Klang, E., Amitai, M., Goldberger, J., & Greenspan, H. (2018). Synthetic data augmentation using gan for improved liver lesion classification, In *2018 ieee 15th international symposium on biomedical imaging (isbi 2018)*. IEEE. (Cit. on p. 18).

Fromm, M., Schubert, M., Castilla, G., Linke, J., & McDermid, G. (2019). Automated detection of conifer seedlings in drone imagery using convolutional neural networks. *Remote Sensing*, *11*(21). https://doi.org/10.3390/rs11212585 (cit. on pp. 15, 18, 19, 21, 26, 29)

Gao, J., French, A. P., Pound, M. P., He, Y., Pridmore, T. P., & Pieters, J. G. (2020). Deep convolutional neural networks for image-based Convolvulus sepium detection in sugar beet fields. *Plant Methods*, *16*(1), 1–12. https://doi.org/10.1186/s13007-020-00570-z (cit. on pp. 18, 19, 25)

Gastellu-Etchegorry, J.-P., Demarez, V., Pinel, V., & Zagolski, F. (1996). Modeling radiative transfer in heterogeneous 3-d vegetation canopies. *Remote sensing of environment*, *58*(2), 131–156 (cit. on p. 14).

Geng, J., Wang, H., Fan, J., & Ma, X. (2017). Deep supervised and contractive neural network for sar image classification. *IEEE Transactions on Geoscience and Remote Sensing*, *55*(4), 2442–2459 (cit. on p. 8).

Ghosal, S., Zheng, B., Chapman, S. C., Potgieter, A. B., Jordan, D. R., Wang, X., Singh, A. K., Singh, A., Hirafuji, M., Ninomiya, S., Ganapathysubramanian, B., Sarkar, S., & Guo, W. (2019). A Weakly Supervised Deep Learning Framework for Sorghum Head Detection and Counting. *Plant Phenomics*, *2019*, 1–14. https://doi.org/10.34133/2019/1525874 (cit. on p. 21)

Girshick, R. (2015). Fast R-CNN, In *Proceedings of the ieee international conference on computer vision*. https://doi.org/10.1109/ICCV.2015.169. (Cit. on p. 23)

Girshick, R., Donahue, J., Darrell, T., Berkeley, U. C., & Malik, J. (2014). R-CNN. *1311.2524v5*, arXiv 1311.2524. https://doi.org/10.1109/CVPR.2014.81 (cit. on p. 23)

Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., & Bengio, Y. (2014). Generative adversarial nets, In *Advances in neural information processing systems*. (Cit. on p. 18).

Guidici, D., & Clark, M. L. (2017). One-Dimensional Convolutional Neural Network Land-Cover Classification of Multi-Seasonal Hyperspectral Imagery in the San Francisco Bay Area , California. *Remote Sens*, *9*, 629. https://doi.org/10.3390/rs9060629 (cit. on pp. 21, 36)

Guirado, E., Alcaraz-Segura, D., Cabello, J., Puertas-Ruiz, S., Herrera, F., & Tabik, S. (2020). Tree Cover Estimation in Global Drylands from Space Using Deep Learning. *Remote Sensing*, *12*(3), 343. https://doi.org/10.3390/rs12030343 (cit. on p. 26)

Guirado, E., Tabik, S., Alcaraz-Segura, D., Cabello, J., & Herrera, F. (2017). Deep-learning Versus OBIA for scattered shrub detection with Google Earth Imagery: Ziziphus lotus as case study. *Remote Sensing*, *9*(12), 1–22. https://doi.org/10.3390/rs9121220 (cit. on pp. 26, 28)

Hamdi, Z. M., Brandmeier, M., & Straub, C. (2019). Forest Damage Assessment Using Deep Learning on High Resolution Remote Sensing Data. *Remote Sensing*, *11*(17), 1976. https://doi.org/10.3390/rs11171976 (cit. on pp. 13, 26)

Hamylton, S., Morris, R., Carvalho, R., Roder, N., Barlow, P., Mills, K., & Wang, L. (2020). Evaluating techniques for mapping island vegetation from unmanned aerial vehicle (UAV) images: Pixel classification, visual interpretation and machine learning approaches. *International Journal of Applied Earth Observation and Geoinformation*, *89*(March), 102085. https://doi.org/10.1016/j.jag.2020.102085 (cit. on p. 26)

Haralick, R. M. (1979). Statistical and structural approaches to texture. *Proceedings of the IEEE*, *67*(5), 786–804. https://doi.org/10.1109/PROC.1979.11328 (cit. on pp. 4, 8)

Hartling, S., Sagan, V., Sidike, P., Maimaitijiang, M., & Carron, J. (2019). Urban tree species classification using a worldview-2/3 and liDAR data fusion approach and deep learning. *Sensors (Switzerland)*, *19*(6), 1–23. https://doi.org/10.3390/s19061284 (cit. on pp. 8, 15, 22, 26, 29, 32, 36)

He, K., Girshick, R., & Dollár, P. (2018). Rethinking ImageNet Pre-training. *arXiv preprint*, arXiv 1811.08883, 1–10 (cit. on p. 20).

He, K., Gkioxari, G., Dollar, P., & Girshick, R. (2017). Mask R-CNN, In *Proceedings of the ieee international conference on computer vision.* https://doi.org/10.1109/ICCV.2017.322. (Cit. on p. 24)

Helber, P., Bischke, B., Dengel, A., & Borth, D. (2019). Eurosat: A novel dataset and deep learning benchmark for land use and land cover classification. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, *12*(7), 2217–2226 (cit. on p. 16).

Hochreiter, S. (1991). Untersuchungen zu dynamischen neuronalen netzen. *Diploma, Technische Universität München*, *91*(1) (cit. on p. 17).

Hochreiter, S. (1998). The vanishing gradient problem during learning recurrent neural nets and problem solutions. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, *6*(02), 107–116 (cit. on p. 17).

Hoeser, T., & Kuenzer, C. (2020). Object Detection and Image Segmentation with Deep Learning on Earth Observation Data : A Review-Part I : Evolution and Recent Trends. *Remote Sensing*, *12*(May), 1667. https://doi.org/10.3390/rs12101667 (cit. on pp. 3, 4, 7, 10)

Huang, B., Lu, K., Audebert, N., Khalel, A., Tarabalka, Y., Malof, J., Boulch, A., Saux, B. L., Collins, L., Bradbury, K., Lefevre, S., & El-Saban, M. (2018). Large-scale semantic classification: Outcome of the first year of inria aerial image labeling benchmark. *International Geoscience and Remote Sensing Symposium (IGARSS)*, *2018-July*, 6947–6950. https://doi.org/10.1109/IGARSS.2018.8518525 (cit. on p. 3)

Jacquemoud, S., Verhoef, W., Baret, F., Bacour, C., Zarco-Tejada, P. J., Asner, G. P., François, C., & Ustin, S. L. (2009). Prospect+ sail models: A review of use for vegetation characterization. *Remote sensing of environment*, *113*, S56–S66 (cit. on p. 14).

Jégou, S., Drozdzal, M., Vazquez, D., Romero, A., & Bengio, Y. (2017). The one hundred layers tiramisu: Fully convolutional densenets for semantic segmentation, In *Proceedings of the ieee conference on computer vision and pattern recognition workshops.* (Cit. on p. 24).

Jiang, S., Yao, W., & Heurich, M. (2019). Dead Wood Detection Based on Semantic Segmentation of Vhr Aerial Cir Imagery Using Optimized

Fcn-Densenet. *ISPRS - International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, *XLII-2/W16*(September), 127–133. https://doi.org/10.5194/isprs-archives-xlii-2-w16-127-2019 (cit. on p. 24)
test123

Jin, S., Su, Y., Zhao, X., Hu, T., & Guo, Q. (2020). A Point-Based Fully Convolutional Neural Network for Airborne LiDAR Ground Point Filtering in Forested Environments. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, *13*, 3958–3974. https://doi.org/10.1109/JSTARS.2020.3008477 (cit. on p. 31)

Jin, S., Su, Y., Gao, S., Wu, F., Hu, T., Liu, J., Li, W., Wang, D., Chen, S., Jiang, Y., Pang, S., & Guo, Q. (2018). Deep learning: Individual maize segmentation from terrestrial lidar data using faster R-CNN and regional growth algorithms. *Frontiers in Plant Science*, *9*(June), 1–10. https://doi.org/10.3389/fpls.2018.00866 (cit. on pp. 18, 23, 25, 30)

Jin, S., Guan, H., Zhang, J., Guo, Q., Su, Y., Gao, S., Wu, F., Xu, K., Ma, Q., Hu, T., Liu, J., & Pang, S. (2019). Separating the Structural Components of Maize for Field Phenotyping Using Terrestrial LiDAR Data and Deep Convolutional Neural Networks. *IEEE Transactions on Geoscience and Remote Sensing*, *PP*, 1–15. https://doi.org/10.1109/tgrs.2019.2953092 (cit. on pp. 21, 24, 30)

Kaartinen, H., Hyyppä, J., Vastaranta, M., Kukko, A., Jaakkola, A., Yu, X., Pyörälä, J., Liang, X., Liu, J., Wang, Y., Kaijaluoto, R., Melkas, T., Holopainen, M., & Hyyppä, H. (2015). Accuracy of kinematic positioning using global satellite navigation systems under forest canopies. *Forests*. https://doi.org/10.3390/f6093218 (cit. on p. 12)

Kampe, T. U., Johnson, B. R., Kuester, M. A., & Keller, M. (2010). Neon: The first continental-scale ecological observatory with airborne remote sensing of vegetation canopy biochemistry and structure. *Journal of Applied Remote Sensing*, *4*(1), 043510 (cit. on p. 16).

Kao, R. H., Gibson, C. M., Gallery, R. E., Meier, C. L., Barnett, D. T., Docherty, K. M., Blevins, K. K., Travers, P. D., Azuaje, E., Springer, Y. P. Et al. (2012). Neon terrestrial field observations: Designing continental-scale, standardized sampling. *Ecosphere*, *3*(12), 1–17 (cit. on p. 16).

Kattenborn, T., Eichel, J., & Fassnacht, F. E. (2019a). Convolutional Neural Networks enable efficient, accurate and fine-grained segmentation of plant species and communities from high-resolution UAV imagery. *Scientific Reports*, *9*(1), 17656. https://doi.org/10.1038/s41598-019-53797-9 (cit. on pp. 13, 17, 24, 26, 29, 32)

Kattenborn, T., Eichel, J., Schmidtlein, S., Wiser, S., Burrows, L., & Fassnacht, F. E. (2020). Convolutional Neural Networks accurately predict cover fractions of plant species and communities in Unmanned

Aerial Vehicle imagery. *Remote Sensing in Ecology and Conservation*, 1–15. https://doi.org/10.1002/rse2.146 (cit. on pp. 13, 21, 22, 26, 28, 32)

Kattenborn, T., Lopatin, J., Förster, M., Braun, A. C., & Fassnacht, F. E. (2019b). UAV data as alternative to field sampling to map woody invasive species based on combined Sentinel-1 and Sentinel-2 data. *Remote Sensing of Environment*, *227*, 61–73. https://doi.org/10.1016/j.rse.2019.03.025 (cit. on p. 27)

Kattenborn, T., Lopatin, J., Förster, M., Braun, A. C., & Fassnacht, F. E. (2019c). UAV data as alternative to field sampling to map woody invasive species based on combined Sentinel-1 and Sentinel-2 data. *Remote Sensing of Environment*, *227*(January), 61–73. https://doi.org/10.1016/j.rse.2019.03.025 (cit. on p. 29)

Kattenborn, T., & Schmidtlein, S. (2019d). Radiative transfer modelling reveals why canopy reflectance follows function. *Scientific Reports*, *9*(1), 6541. https://doi.org/10.1038/s41598-019-43011-1 (cit. on p. 12)

Kelly, L., Suominen, H., Goeuriot, L., Neves, M., Kanoulas, E., Li, D., Azzopardi, L., Spijker, R., Zuccon, G., Scells, H. Et al. (2019). Overview of the clef ehealth evaluation lab 2019, In *International conference of the cross-language evaluation forum for european languages*. Springer. (Cit. on p. 16).

Kerdegari, H., Razaak, M., Argyriou, V., & Remagnino, P. (2019). Smart Monitoring of Crops Using Generative Adversarial Networks. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, *11678 LNCS*, 554–563. https://doi.org/10.1007/978-3-030-29888-3_45 (cit. on p. 18)

Kirillov, A., He, K., Girshick, R., Rother, C., & Dollar, P. (2019). Panoptic segmentation. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, *2019-June*arXiv 1801.00868, 9396–9405. https://doi.org/10.1109/CVPR.2019.00963 (cit. on p. 23)

Knauer, U., von Rekowski, C. S., Stecklina, M., Krokotsch, T., Pham Minh, T., Hauffe, V., Kilias, D., Ehrhardt, I., Sagischewski, H., Chmara, S., & Seiffert, U. (2019). Tree Species Classification Based on Hybrid Ensembles of a Convolutional Neural Network (CNN) and Random Forest Classifiers. *Remote Sensing*, *11*(23), 2788. https://doi.org/10.3390/rs11232788 (cit. on p. 36)

Ko, C., Kang, J., & Sohn, G. (2018). DEEP MULTI-TASK LEARNING for TREE GENERA CLASSIFICATION. *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, *4*(2), 153–159. https://doi.org/10.5194/isprs-annals-IV-2-153-2018 (cit. on pp. 18, 22, 30)

Korznikov, K. (2020). Automatic Windthrow Detection Using Very-High-Resolution Satellite Imagery and Deep Learning. *Remote Sensing*, *12*(April), 1145. https://doi.org/10.3390/rs12071145 (cit. on pp. 26, 29)

Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks, In *Advances in neural information processing systems*. https://doi.org/10.1201/9781420010749. (Cit. on pp. 17, 18)

Kussul, N., Lavreniuk, M., Skakun, S., & Shelestov, A. (2017). Deep learning classification of land cover and crop types using remote sensing data. *IEEE Geoscience and Remote Sensing Letters*, *14*(5), 778–782 (cit. on pp. 21, 24).

Langford, Z. L., Kumar, J., Hoffman, F. M., Breen, A. L., & Iversen, C. M. (2019). Arctic vegetation mapping using unsupervised training datasets and convolutional neural networks. *Remote Sensing*, *11*(1), 1–23. https://doi.org/10.3390/rs11010069 (cit. on pp. 26, 32)

Lee, J., Kim, E., Lee, S., Lee, J., & Yoon, S. (2019). Ficklenet: Weakly and semi-supervised semantic image segmentation using stochastic inference, In *Proceedings of the ieee conference on computer vision and pattern recognition*. (Cit. on pp. 20, 38).

Leitão, P. J., Schwieder, M., Pötzschner, F., Pinto, J. R. R., Teixeira, A. M. C., Pedroni, F., Sanchez, M., Rogass, C., van der Linden, S., Bustamante, M. M. C., & Hostert, P. (2018). From sample to pixel: multiscale remote sensing data for upscaling aboveground carbon data in heterogeneous landscapes. *Ecosphere*, *9*(8), e02298. https://doi.org/10.1002/ecs2.2298 (cit. on p. 12)

Lepš, J., & Hadincová, V. (1992). *How reliable are our vegetation analyses?* (Tech. rep. No. 1). https://doi.org/10.2307/3236006. (Cit. on p. 13)

Li, K., Wu, Z., Peng, K.-C., Ernst, J., & Fu, Y. (2018). Tell me where to look: Guided attention inference network, In *Proceedings of the ieee conference on computer vision and pattern recognition*. (Cit. on pp. 20, 38).

Li, W., Fu, H., Yu, L., & Cracknell, A. (2017). Deep learning based oil palm tree detection and counting for high-resolution remote sensing images. *Remote Sensing*, *9*(1). https://doi.org/10.3390/rs9010022 (cit. on pp. 25, 29)

Liao, C., Wang, J., Xie, Q., Al Baz, A., Huang, X., Shang, J., & He, Y. (2020). Synergistic Use of Multi-Temporal RADARSAT-2 and VENuS Data for Crop Classification Based on 1D Convolutional Neural Network CSA SOAR-E View project NSERC discovery View project Synergistic Use of Multi-Temporal RADARSAT-2 and VENuS Data for Crop Classifi. *Remote Sensing*, *12*(832), 832. https://doi.org/10.3390/rs12050832 (cit. on pp. 21, 30, 32, 36)

Liu, T., & Abd-Elrahman, A. (2018a). Deep convolutional neural network training enrichment using multi-view object-based analysis of Unmanned Aerial systems imagery for wetlands classification. *ISPRS Journal of Photogrammetry and Remote Sensing*, *139*, 154–170. https://doi.org/10.1016/j.isprsjprs.2018.03.006 (cit. on pp. 22, 27, 36)

Liu, T., Abd-Elrahman, A., Morton, J., & Wilhelm, V. L. (2018b). Comparing fully convolutional networks, random forest, support vector machine, and patch-based deep convolutional neural networks for object-based wetland mapping using images from small unmanned aircraft system. *GIScience and Remote Sensing*, *55*(2), 243–264. https://doi.org/10.1080/15481603.2018.1426091 (cit. on pp. 15, 36)

Liu, T., Abd-Elrahman, A., Zare, A., Dewitt, B. A., Flory, L., & Smith, S. E. (2018c). A fully learnable context-driven object-based model for mapping land cover using multi-view data from unmanned aircraft systems. *Remote Sensing of Environment*, *216*(June), 328–344. https://doi.org/10.1016/j.rse.2018.06.031 (cit. on pp. 26, 27)

Long, J., Shelhamer, E., & Darrell, T. (2015). Fully convolutional networks for semantic segmentation, In *Proceedings of the ieee conference on computer vision and pattern recognition*. (Cit. on pp. 24, 66).

Lopatin, J., Dolos, K., Kattenborn, T., & Fassnacht, F. E. (2019). How canopy shadow affects invasive plant species classification in high spatial resolution remote sensing. *Remote Sensing in Ecology and Conservation*, 1–16. https://doi.org/10.1002/rse2.109 (cit. on p. 29)

López-Jiménez, E., Vasquez-Gomez, J. I., Sanchez-Acevedo, M. A., Herrera-Lozada, J. C., & Uriarte-Arcia, A. V. (2019). Columnar cactus recognition in aerial images using a deep learning approach. *Ecological Informatics*, *52*, 131–138. https://doi.org/10.1016/j.ecoinf.2019.05.005 (cit. on p. 23)

Lottes, P., Behley, J., Milioto, A., & Stachniss, C. (2018). Fully convolutional networks with sequential information for robust crop and weed detection in precision farming. *IEEE Robotics and Automation Letters*, *3*(4), 1806.03412, 2870–2877. https://doi.org/10.1109/LRA.2018.2846289 (cit. on pp. 21, 25, 26, 32)

Lunetta, R. S., Congalton, R. G., Fenstermaker, L. K., Jensen, J. R., McGwire, K. C., & Tinney, L. R. (1991). Remote sensing and geographic information system data integration: error sources and research issues. *Photogrammetric Engineering & Remote Sensing*, *57*(6), 677–687 (cit. on p. 13).

Ma, J., Li, Y., Chen, Y., Du, K., Zheng, F., Zhang, L., & Sun, Z. (2019). Estimating above ground biomass of winter wheat at early growth stages using digital images and deep convolutional neural network. *European Journal of Agronomy*, *103*(June 2018), 117–129. https://doi.org/10.1016/j.eja.2018.12.004 (cit. on pp. 26, 34)

Mahdianpari, M., Salehi, B., Rezaee, M., Mohammadimanesh, F., & Zhang, Y. (2018). Very deep convolutional neural networks for complex land cover mapping using multispectral remote sensing imagery. *Remote Sensing*, *10*(7). https://doi.org/10.3390/rs10071119 (cit. on pp. 19, 24, 26)

Maier, S., Lüdeker, W., & Günther, K. (1999). Slop: A revised version of the stochastic model for leaf optical properties. *Remote Sensing of Environment*, *68*(3), 273–280 (cit. on p. 14).

Malambo, L., Rooney, W., Zhou, T., Popescu, S., Ku, N.-W., & Moore, S. (2019). A deep learning semantic segmentation-based approach for field-level sorghum panicle counting. *Remote Sensing*, *11*(24). https://doi.org/10.3390/rs11242939 (cit. on p. 25)

Marconi, S., Graves, S. J., Gong, D., Nia, M. S., Le Bras, M., Dorr, B. J., Fontana, P., Gearhart, J., Greenberg, C., Harris, D. J. Et al. (2019). A data science challenge for converting airborne remote sensing data into ecological information. *PeerJ*, *6*, e5843 (cit. on p. 16).

Masi, G., Cozzolino, D., Verdoliva, L., & Scarpa, G. (2016). Pansharpening by convolutional neural networks. *Remote Sensing*, *8*(7). https://doi.org/10.3390/rs8070594 (cit. on p. 29)

Mazzia, V., Khaliq, A., & Chiaberge, M. (2019). Improvement in Land Cover and Crop Classification based on Temporal Features Learning from Sentinel-2 Data Using Recurrent-Convolutional Neural Network (R-CNN). *Applied Sciences*, *10*(1), 238. https://doi.org/10.3390/app10010238 (cit. on pp. 33, 34, 36)

McInnes, L., Healy, J., & Melville, J. (2018). Umap: Uniform manifold approximation and projection for dimension reduction. *arXiv preprint arXiv:1802.03426* (cit. on p. 37).

McRoberts, R. E., & Tomppo, E. O. (2007). Remote sensing support for national forest inventories. *Remote Sensing of Environment*. https://doi.org/10.1016/j.rse.2006.09.034 (cit. on p. 3)

Mehdipour Ghazi, M., Yanikoglu, B., & Aptoula, E. (2017). Plant identification using deep neural networks via optimization of transfer learning parameters. *Neurocomputing*, *235*(April 2016), 228–235. https://doi.org/10.1016/j.neucom.2017.01.018 (cit. on p. 19)

Milioto, A., Lottes, P., & Stachniss, C. (2017). REAL-TIME BLOB-WISE SUGAR BEETS VS WEEDS CLASSIFICATION for MONITORING FIELDS USING CONVOLUTIONAL NEURAL NETWORKS. *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, *4*(2W3), 41–48. https://doi.org/10.5194/isprs-annals-IV-2-W3-41-2017 (cit. on pp. 21, 25)

Mohammadimanesh, F., Salehi, B., Mahdianpari, M., Gill, E., & Molinier, M. (2019). A new fully convolutional neural network for semantic segmentation of polarimetric SAR imagery in complex land cover ecosystem. *ISPRS Journal of Photogrammetry and Remote Sensing*, *151*(March), 223–236. https://doi.org/10.1016/j.isprsjprs.2019.03.015 (cit. on pp. 26, 30, 36, 37)

Molnar, C. (2019). *Interpretable machine learning: A guide for making black box models explainable* [https://christophm.github.io/interpretable-ml-book/]. (Cit. on p. 41).

Mubin, N. A., Nadarajoo, E., Shafri, H. Z. M., & Hamedianfar, A. (2019). Young and mature oil palm tree detection and counting using convolutional neural network deep learning method. *International Journal of Remote Sensing*, *40*(19), 7500–7515. https://doi.org/10.1080/01431161.2019.1569282 (cit. on p. 25)

Mulla, D. J. (2013). Twenty five years of remote sensing in precision agriculture: Key advances and remaining knowledge gaps. https://doi.org/10.1016/j.biosystemseng.2012.08.009. (Cit. on p. 3)

Nagendra, H., Lucas, R., Honrado, J. P., Jongman, R. H., Tarantino, C., Adamo, M., & Mairota, P. (2013). Remote sensing for conservation monitoring: Assessing protected areas, habitat extent, habitat condition, species diversity, and threats. *Ecological Indicators*. https://doi.org/10.1016/j.ecolind.2012.09.014 (cit. on p. 3)

Natesan, S., Armenakis, C., & Vepakomma, U. (2019). Resnet-based tree species classification using uav images. *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences - ISPRS Archives*, *42*(2/W13), 475–481. https://doi.org/10.5194/isprs-archives-XLII-2-W13-475-2019 (cit. on pp. 13, 26)

Neupane, B., Horanont, T., & Hung, N. D. (2019). Deep learning based banana plant detection and counting using high-resolution red-green-blue (RGB) images collected from unmanned aerial vehicle (UAV). *PloS one*, *14*(10), e0223906. https://doi.org/10.1371/journal.pone.0223906 (cit. on pp. 21, 25, 28, 29)

Nevavuori, P., Narra, N., & Lipping, T. (2019). Crop yield prediction with deep convolutional neural networks. *Computers and Electronics in Agriculture*, *163*(June), 104859. https://doi.org/10.1016/j.compag.2019.104859 (cit. on pp. 14, 28)

Nezami, S., Khoramshahi, E., Pölönen, I., Nevalainen, O., Honkavaara, E., Honkavaara@nls, E., & Fi, E. H. (2020). Tree Species Classification of Drone Hyperspectral and RGB Imagery with Deep Learning Convolutional Neural Networks Hyperspectral imaging guided skin cancer diagnostics View project DroneKnowledge View project Somayeh Nezami Finnish Geodetic Institute Tre. https://doi.org/10.20944/preprints202002.0334.v1 (cit. on pp. 21, 26, 29, 32)

Nguyen, G., Dlugolinsky, S., Bobák, M., Tran, V., Garcia, A. L., Heredia, I., Malik, P., & Hluchy, L. (2019). Machine learning and deep learning frameworks and libraries for large-scale data mining: A survey. *Artificial Intelligence Review*, *52*(1), 77–124 (cit. on p. 10).

Noack, A. (2007). Energy models for graph clustering. *J. Graph Algorithms Appl.*, *11*(2), 453–480 (cit. on p. 64).

North, P. R. (1996). Three-dimensional forest light interaction model using a monte carlo method. *IEEE Transactions on geoscience and remote sensing*, *34*(4), 946–956 (cit. on p. 14).

Olah, C., Mordvintsev, A., & Schubert, L. (2017). Feature visualization. *Distill*, *2*(11), e7 (cit. on p. 36).

Osco, L. P., de Arruda, M. d. S., Marcato Junior, J., da Silva, N. B., Ramos, A. P. M., Moryia, É. A. S., Imai, N. N., Pereira, D. R., Creste, J. E., Matsubara, E. T., Li, J., & Gonçalves, W. N. (2020). A convolutional neural network approach for counting and geolocating citrus-trees in UAV multispectral imagery. *ISPRS Journal of Photogrammetry and Remote Sensing*, *160*(November 2019), 97–106. https://doi.org/10.1016/j.isprsjprs.2019.12.010 (cit. on p. 29)

Pettorelli, N., Schulte to Bühne, H., Tulloch, A., Dubois, G., Macinnis-Ng, C., Queirós, A. M., Keith, D. A., Wegmann, M., Schrodt, F., Stellmes, M., Sonnenschein, R., Geller, G. N., Roy, S., Somers, B., Murray, N., Bland, L., Geijzendorffer, I., Kerr, J. T., Broszeit, S., . . . Nicholson, E. (2017). Satellite remote sensing of ecosystem functions: opportunities, challenges and way forward. *Remote Sensing in Ecology and Conservation*, 1–23. https://doi.org/10.1002/rse2.59 (cit. on p. 3)

Pinheiro, M., Roberti, D., Almeida, A. D., Almeida, D. D., Baldez, J., Minervino, S., Franklin, H., Veras, P., Formighieri, A., Alexandre, C., Santos, N., Aurélio, M., & Ferreira, D. (2020). Forest Ecology and Management Individual tree detection and species classification of Amazonian palms using UAV images and deep learning. *Forest Ecology and Management*, *475*, 118397. https://doi.org/10.1016/j.foreco.2020.118397 (cit. on p. 26)

Pires de Lima, R., & Marfurt, K. (2020). Convolutional neural network for remote-sensing scene classification: Transfer learning analysis. *Remote Sensing*, *12*(1), 86 (cit. on p. 19).

Pouliot, D., Latifovic, R., Pasher, J., & Duffe, J. (2019). Assessment of convolution neural networks for wetland mapping with landsat in the central Canadian boreal forest region. *Remote Sensing*, *11*(7). https://doi.org/10.3390/rs11070772 (cit. on p. 26)

Qian, W., Huang, Y., Liu, Q., Fan, W., Sun, Z., Dong, H., Wan, F., & Qiao, X. (2020). UAV and a deep convolutional neural network for monitoring invasive alien plants in the wild. *Computers and Electronics in Agriculture*, *174*(May), 105519. https://doi.org/10.1016/j.compag.2020.105519 (cit. on p. 26)

Reichstein, M., Camps-Valls, G., Stevens, B., Jung, M., Denzler, J., Carvalhais, N., & Prabhat. (2019). Deep learning and process understanding for data-driven Earth system science. *Nature*, *566*(7743), 195–204. https://doi.org/10.1038/s41586-019-0912-1 (cit. on pp. 4, 14, 15, 34, 36)

Ren, S., He, K., Girshick, R., & Sun, J. (2017). Faster R-CNN, arXiv 1506.01497. https://doi.org/10.1109/TPAMI.2016.2577031 (cit. on p. 23)

Rezaee, M., Mahdianpari, M., Zhang, Y., & Salehi, B. (2018). Deep Convolutional Neural Network for Complex Wetland Classification Using Optical Remote Sensing Imagery. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, *11*(9), 3030–3039. https://doi.org/10.1109/JSTARS.2018.2846178 (cit. on pp. 19, 24, 34, 36)

Riese, F. M., Keller, S., & Hinz, S. (2020). Supervised and semi-supervised self-organizing maps for regression and classification focusing on hyperspectral data. *Remote Sensing*, *12*(1), 7 (cit. on p. 40).

Ronneberger, O., Fischer, P., & Brox, T. (2015). U-net: Convolutional networks for biomedical image segmentation, In *International conference on medical image computing and computer-assisted intervention*, Cham., Springer. https://doi.org/10.1007/978-3-319-24574-4_28. (Cit. on pp. 3, 17, 24)

Roussel, J.-R., Auty, D., De Boissieu, F., & Sánchez Meador, A. (2017). Lidr: Airborne lidar data manipulation and visualization for forestry applications. r package version 1.2. 0. (Cit. on p. 21).

Russell, B. C., Torralba, A., Murphy, K. P., & Freeman, W. T. (2008). LabelMe: A database and web-based tool for image annotation. *International Journal of Computer Vision*. https://doi.org/10.1007/s11263-007-0090-8 (cit. on p. 13)

Sa, I., Popović, M., Khanna, R., Chen, Z., Lottes, P., Liebisch, F., Nieto, J., Stachniss, C., Walter, A., & Siegwart, R. (2018). WeedMap: A large-scale semantic weed mapping framework using aerial multispectral imaging and deep neural network for precision farming. *Remote Sensing*, *10*(9), arXiv 1808.00100. https://doi.org/10.3390/rs10091423 (cit. on p. 25)

Safonova, A., Tabik, S., Alcaraz-Segura, D., Rubtsov, A., Maglinets, Y., & Herrera, F. (2019). Detection of Fir Trees (Abies sibirica) Damaged by the Bark Beetle in Unmanned Aerial Vehicle Images with Deep Learning. *Remote Sensing*, *11*(6), 643. https://doi.org/10.3390/rs11060643 (cit. on pp. 18, 23, 26)

Schiefer, F., Kattenborn, T., Frick, A., Frey, J., Schall, P., Koch, B., & Schmidtlein, S. (2020). Mapping forest tree species in high resolution uav-based rgb-imagery by means of convolutional neural networks. *ISPRS Journal of Photogrammetry and Remote Sensing*, *170*, 205–215. https://doi.org/https://doi.org/10.1016/j.isprsjprs.2020.10.015 (cit. on pp. 13, 26, 28, 36, 37)

Schmitt, M., Hughes, L. H., Qiu, C., & Zhu, X. X. (2019). Sen12ms–a curated dataset of georeferenced multi-spectral sentinel-1/2 imagery for deep learning and data fusion. *arXiv preprint arXiv:1906.07789* (cit. on pp. 16, 20).

Schmitt, M., Prexl, J., Ebel, P., Liebel, L., & Zhu, X. X. (2020). Weakly Supervised Semantic Segmentation of Satellite Images for Land Cover Mapping – Challenges and Opportunities. *arXiv preprint*, arXiv 2002.08254. http://arxiv.org/abs/2002.08254 (cit. on pp. 15, 20)

Selvaraju, R. R., Cogswell, M., Das, A., Vedantam, R., Parikh, D., & Batra, D. (2019). Grad-CAM: Visual Explanations from Deep Networks via Gradient-Based Localization. *International Journal of Computer Vision*, *17*arXiv 1610.02391, 331–336. https://doi.org/10.1007/s11263-019-01228-7 (cit. on p. 37)

Shorten, C., & Khoshgoftaar, T. M. (2019). A survey on Image Data Augmentation for Deep Learning. *Journal of Big Data*, *6*(1). https://doi.org/10.1186/s40537-019-0197-0 (cit. on p. 18)

Smilkov, D., Thorat, N., Kim, B., Viégas, F., & Wattenberg, M. (2017). SmoothGrad: removing noise by adding noise, arXiv 1706.03825. http://arxiv.org/abs/1706.03825 (cit. on p. 37)

Sothe, C. [C.], De Almeida, C. M., Schimalski, M. B., Liesenberg, V., La Rosa, L. E., Castro, J. D., & Feitosa, R. Q. (2020). A comparison of machine and deep-learning algorithms applied to multisource data for a subtropical forest area classification. *International Journal of Remote Sensing*, *41*(5), 1943–1969. https://doi.org/10.1080/01431161.2019.1681600 (cit. on pp. 22, 32)

Sothe, C., Almeida, C. M. D., Schimalski, M. B., Rosa, L. E. C. L., Castro, J. D. B., Feitosa, R. Q., Dalponte, M., Lima, C. L., Liesenberg, V., & Miyoshi, G. T. (2020). Comparative performance of convolutional neural network , weighted and conventional support vector machine and random forest for classifying tree species using hyperspectral and photogrammetric data. *GIScience & Remote Sensing*, *00*(00), 1–26. https://doi.org/10.1080/15481603.2020.1712102 (cit. on pp. 8, 32)

Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., & Salakhutdinov, R. (2014). Dropout: A simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research* (cit. on p. 17).

Su, H., Maji, S., Kalogerakis, E., & Learned-Miller, E. (2015). Multi-view convolutional neural networks for 3D shape recognition, In *Proceedings of the ieee international conference on computer vision*. https://doi.org/10.1109/ICCV.2015.114. (Cit. on p. 18)

Sumbul, G., Charfuelan, M., Demir, B., & Markl, V. (2019). Bigearthnet: A large-scale benchmark archive for remote sensing image understanding, In *Igarss 2019-2019 ieee international geoscience and remote sensing symposium*. IEEE. (Cit. on p. 16).

Sun, Y., Huang, J., Ao, Z., Lao, D., & Xin, Q. (2019). Deep Learning Approaches for the Mapping of Tree Species Diversity in a Tropical Wetland Using Airborne LiDAR and High-Spatial-Resolution Remote Sensing Images. *Forests*, *10*(11), 1047. https://doi.org/10.3390/f10111047 (cit. on pp. 13, 22)

Too, E. C., Yujian, L., Njuki, S., & Yingchun, L. (2019). A comparative study of fine-tuning deep learning models for plant disease identification. *Computers and Electronics in Agriculture*, *161*, 272–279 (cit. on p. 19).

Torres, D. L., Feitosa, R. Q., Happ, P. N., La Rosa, L. E. C., Junior, J. M., Martins, J., Bressan, P. O., Gonçalves, W. N., & Liesenberg, V. (2020). Applying fully convolutional architectures for semantic segmentation of a single tree species in urban environment on high resolution UAV optical imagery. *Sensors (Switzerland)*, *20*(2), 1–20. https://doi.org/10.3390/s20020563 (cit. on pp. 24, 26)

Toth, C., & Jóźków, G. (2016). Remote sensing platforms and sensors: A survey. https://doi.org/10.1016/j.isprsjprs.2015.10.004. (Cit. on p. 3)

Trier, Ø. D., Salberg, A. B., Kermit, M., Rudjord, Ø., Gobakken, T., Næsset, E., & Aarsten, D. (2018). Tree species classification in Norway from airborne hyperspectral and airborne laser scanning data. *European Journal of Remote Sensing*, *51*(1), 336–351. https://doi.org/10.1080/22797254.2018.1434424 (cit. on pp. 26, 32)

Tuia, D., Persello, C., & Bruzzone, L. (2016). Domain adaptation for the classification of remote sensing data: An overview of recent advances. *IEEE geoscience and remote sensing magazine*, *4*(2), 41–57 (cit. on p. 19).

Turner, W., Spector, S., Gardiner, N., Fladeland, M., Sterling, E., & Steininger, M. (2003). Remote sensing for biodiversity science and conservation. https://doi.org/10.1016/S0169-5347(03)00070-3. (Cit. on p. 3)

Valbuena, R., Mauro, F., Rodriguez-Solano, R., & Manzanera, J. A. (2013). Accuracy and precision of GPS receivers under forest canopies in a mountainous environment. *Spanish Journal of Agricultural Research*. https://doi.org/10.5424/sjar/2010084-1242 (cit. on p. 12)

Van Eck, N., & Waltman, L. (2010). Software survey: Vosviewer, a computer program for bibliometric mapping. *scientometrics*, *84*(2), 523–538 (cit. on pp. 42, 64).

Wagner, F. H., Sanchez, A., Aidar, M. P. M., Rochelle, A. L. C., Tarabalka, Y., Fonseca, M. G., Phillips, O. L., Gloor, E., & Aragão, L. (2020). Mapping Atlantic rainforest degradation and regeneration history with indicator species using convolutional network. *Plos One*, *15*(2), e0229448. https://doi.org/10.1371/journal.pone.0229448 (cit. on p. 21)

Wagner, F., Sanchez, A., Tarabalka, Y., Lotte, R. G., Ferreira, M. P., Aidar, M. P., Gloor, E., Phillips, O. L., & Aragao, L. (2019). Using the u-net convolutional network to map forest types and disturbance in the atlantic rainforest with very high resolution images. *Remote Sensing in Ecology and Conservation*, *5*(4), 360–375 (cit. on pp. 17, 24, 26).

Wang, Z., & Yang, J. (2017). Diabetic retinopathy detection via deep convolutional networks for discriminative localization and visual explanation. *arXiv preprint arXiv:1703.10757* (cit. on p. 37).

Wang, Chen, Cao, An, Chen, Xue, & Yun. (2019). Individual Rubber Tree Segmentation Based on Ground-Based LiDAR Data and Faster R-CNN of Deep Learning. *Forests*, *10*(9), 793. https://doi.org/10.3390/f10090793 (cit. on pp. 13, 24, 26, 30)

Weinmann, M., Jutzi, B., Hinz, S., & Mallet, C. (2015). Semantic point cloud interpretation based on optimal neighborhoods, relevant features and efficient classifiers. *ISPRS Journal of Photogrammetry and Remote Sensing*. https://doi.org/10.1016/j.isprsjprs.2015.01.016 (cit. on p. 8)

Weinstein, B. G., Marconi, S., Bohlman, S. A., Zare, A., & White, E. P. (2020). Cross-site learning in deep learning RGB tree crown detection. *Ecological Informatics*, *56*(December 2019), 101061. https://doi.org/10.1016/j.ecoinf.2020.101061 (cit. on pp. 15, 16, 23, 26, 29, 35)

Weinstein, B. G., Marconi, S., Bohlman, S., Zare, A., & White, E. (2019). Individual tree-crown detection in rgb imagery using semi-supervised deep learning neural networks. *Remote Sensing*, *11*(11), 1–13. https://doi.org/10.3390/rs11111309 (cit. on pp. 21, 23)

White, J. C., Coops, N. C., Wulder, M. A., Vastaranta, M., Hilker, T., & Tompalski, P. (2016). Remote Sensing Technologies for Enhancing Forest Inventories: A Review. https://doi.org/10.1080/07038992.2016.1207484. (Cit. on p. 3)

Windrim, L., & Bryson, M. (2020). Detection, segmentation, and model fitting of individual tree stems from airborne laser scanning of forests using deep learning. *Remote Sensing*, *12*(9). https://doi.org/10.3390/RS12091469 (cit. on pp. 30, 31)

Xi, Y., Ren, C., Wang, Z., Wei, S., Bai, J., Zhang, B., Xiang, H., & Chen, L. (2019). Mapping Tree Species Composition Using OHS-1 Hyperspectral Data and Deep Learning Algorithms in Changbai Mountains, Northeast China. *Forests*, *10*(9), 818. https://doi.org/10.3390/f10090818 (cit. on pp. 21, 36)

Xi, Z., Hopkinson, C., & Chasmer, L. (2018). Filtering stems and branches from terrestrial laser scanning point clouds using deep 3-D fully convolutional networks. *Remote Sensing*, *10*(8). https://doi.org/10.3390/rs10081215 (cit. on p. 13)

Yang, Q., Shi, L., Han, J., Zha, Y., & Zhu, P. (2019). Deep convolutional neural networks for rice grain yield estimation at the ripening stage using UAV-based remotely sensed images. *Field Crops Research*, *235*(February), 142–153. https://doi.org/10.1016/j.fcr.2019.02.022 (cit. on pp. 14, 22, 25, 29, 32, 34)

Yuan, Q., Wei, Y., Meng, X., Shen, H., & Zhang, L. (2018). A Multiscale and Multidepth Convolutional Neural Network for Remote Sensing Imagery, *11*(3), 978–989 (cit. on pp. 29, 33).

Zarco-Tejada, P. J., Hornero, A., Beck, P. S., Kattenborn, T., Kempeneers, P., & Hernández-Clemente, R. (2019). Chlorophyll content estimation in an open-canopy conifer forest with Sentinel-2A and hyperspectral imagery in the context of forest decline. *Remote Sensing of Environment*, *223*, 320–335. https://doi.org/10.1016/j.rse.2019.01.031 (cit. on p. 29)

Zarco-Tejada, P. J., Camino, C., Beck, P. S., Calderon, R., Hornero, A., Hernández-Clemente, R., Kattenborn, T., Montes-Borrego, M., Susca, L., Morelli, M., Gonzalez-Dugo, V., North, P. R., Landa, B. B., Boscia, D., Saponari, M., & Navas-Cortes, J. A. (2018). Previsual symptoms of Xylella fastidiosa infection revealed in spectral plant-trait alterations. *Nature Plants*, *4*(7), 432–439. https://doi.org/10.1038/s41477-018-0189-7 (cit. on p. 29)

Zhang, B., Huang, S., Shen, W., & Wei, Z. (2019). Explaining the pointnet: What has been learned inside the pointnet?, In *Proceedings of the ieee conference on computer vision and pattern recognition workshops*. (Cit. on p. 37).

Zhang, L., Shao, Z., Liu, J., & Cheng, Q. (2019). Deep learning based retrieval of forest aboveground biomass from combined LiDAR and landsat 8 data. *Remote Sensing*, *11*(12). https://doi.org/10.3390/rs11121459 (cit. on p. 3)

Zhang, M., Lin, H., Wang, G., Sun, H., & Fu, J. (2018). Mapping paddy rice using a Convolutional Neural Network (CNN) with Landsat 8 datasets in the Dongting Lake Area, China. *Remote Sensing*, *10*(11). https://doi.org/10.3390/rs10111840 (cit. on pp. 24, 25, 34, 36)

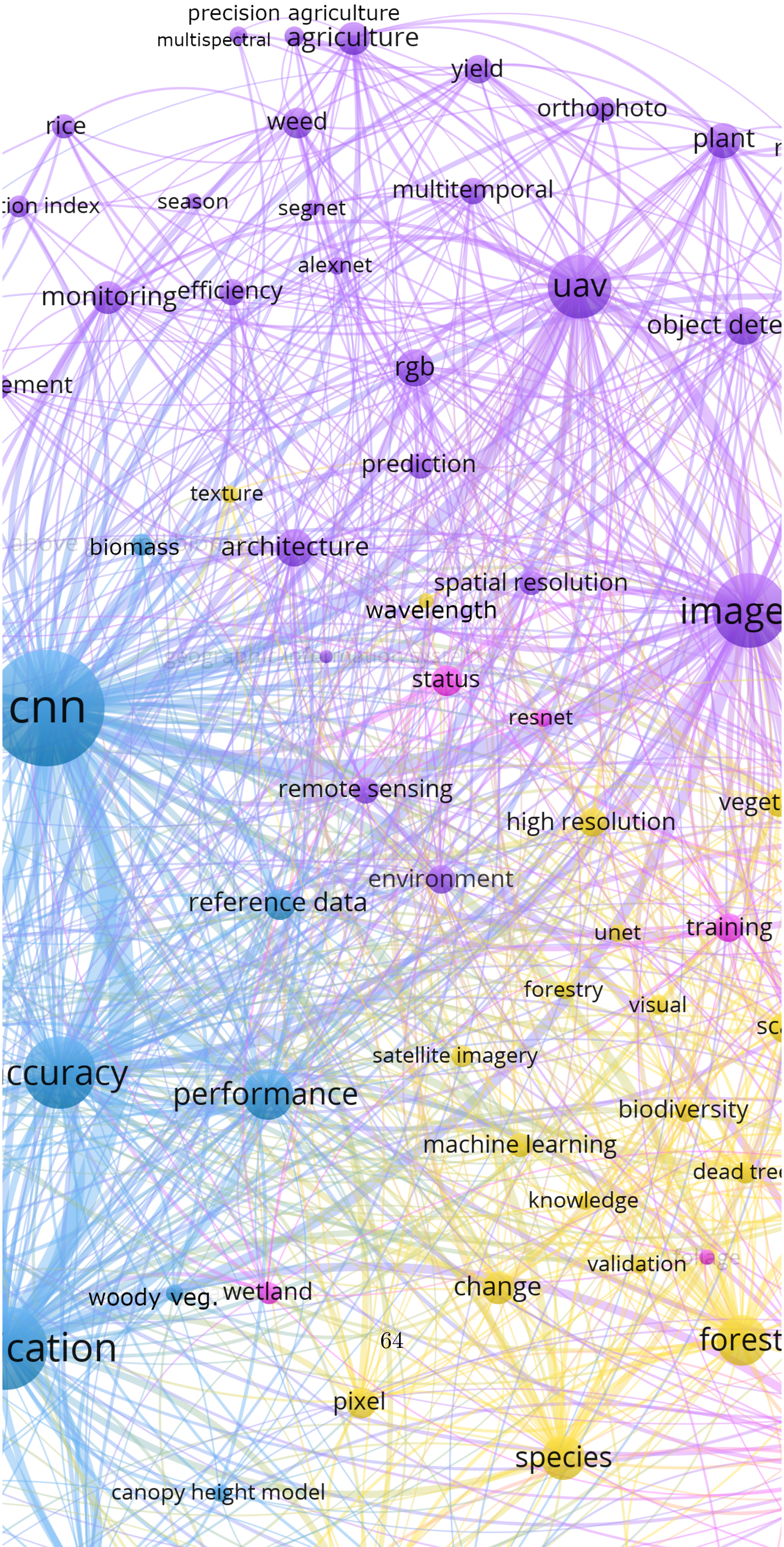
Zhao, X., Yuan, Y., Song, M., Ding, Y., Lin, F., Liang, D., & Zhang, D. (2019). Use of unmanned aerial vehicle imagery and deep learning unet to extract rice lodging. *Sensors (Switzerland)*, *19*(18), 1–13. https://doi.org/10.3390/s19183859 (cit. on p. 29)

Zhong, L., Hu, L., & Zhou, H. (2019). Deep learning based multi-temporal crop classification. *Remote Sensing of Environment*, *221*, 430–443. https://doi.org/10.1016/j.rse.2018.11.032 (cit. on pp. 21, 25, 34, 36)

Zhu, X. X., Montazeri, S., Ali, M., Hua, Y., Wang, Y., Mou, L., Shi, Y., Xu, F., & Bamler, R. (2020). Deep learning meets sar. (Cit. on p. 30).

Zhu, X. X., Tuia, D., Mou, L., Xia, G. S., Zhang, L., Xu, F., & Fraundorfer, F. (2017). Deep Learning in Remote Sensing: A Comprehensive Review and List of Resources. *IEEE Geoscience and Remote Sensing Magazine*, *5*(4), 8–36. https://doi.org/10.1109/MGRS.2017.2762307 (cit. on pp. 3, 4, 7, 15, 34)

Zou, X., Cheng, M., Wang, C., Xia, Y., & Li, J. (2017). Tree Classification in Complex Forest Point Clouds Based on Deep Learning. *IEEE Geoscience and Remote Sensing Letters*, *14*(12), 2360–2364. https://doi.org/10.1109/LGRS.2017.2764938 (cit. on pp. 18, 26, 30)



precision agriculture
multispectral
agriculture
yield
orthophoto
plant
rice
weed
ion index
season
segnet
multitemporal
alexnet
monitoring
efficiency
uav
object dete
ement
rgb
prediction
texture
biomass
architecture
spatial resolution
wavelength
image
status
cnn
resnet
remote sensing
high resolution
veget
environment
reference data
unet
training
forestry
visual
satellite imagery
ccuracy
performance
biodiversity
machine learning
dead tre
knowledge
validation
woody veg.
wetland
change
cation
forest
pixel
species
canopy height model

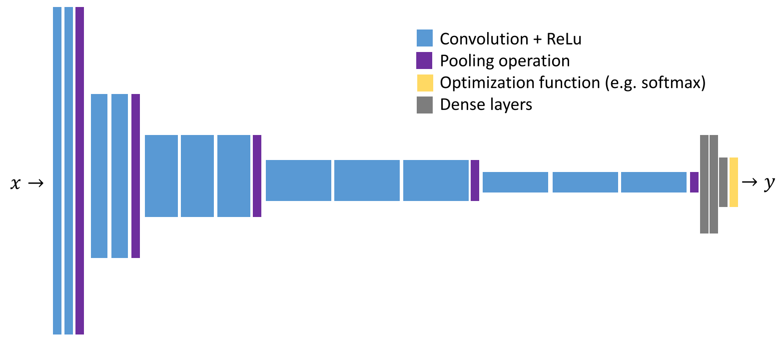


Figure 4: Schematic diagram of the *VGG-16* architecture. The 16 stands for the number of convolutional and dense layers. Frequently used alternatives are *VGG-8* and *VGG-19*.

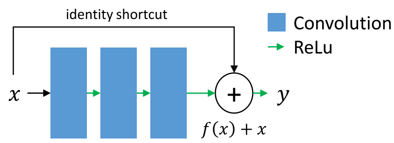


Figure 5: Schematic diagram of a residual building block used in repeated sequence in common *ResNet* architectures.

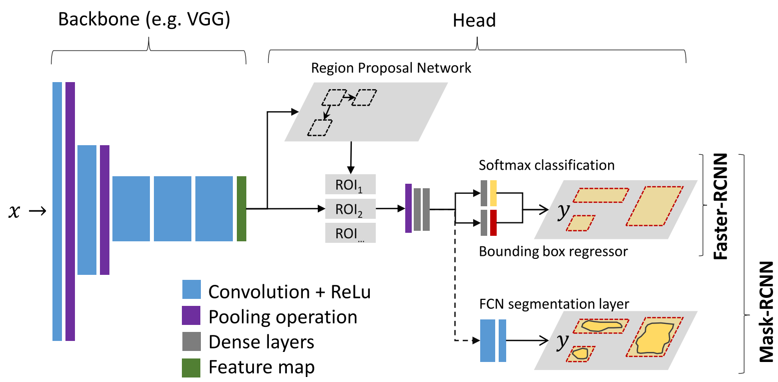


Figure 6: Faster-R-CNN and Mask-RCNN, respectively.

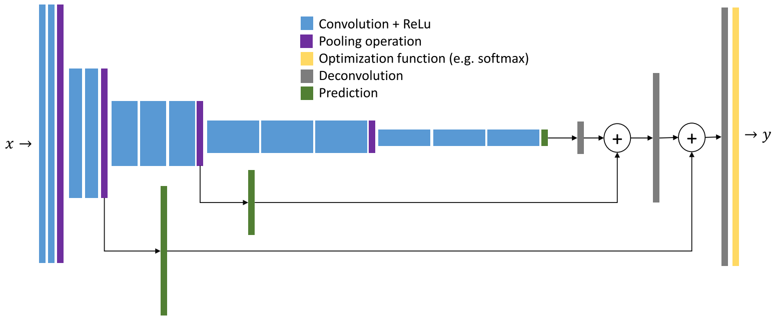


Figure 7: Schematic diagram of the FCN architecture as proposed by Long et al. (2015). Predictions (also referred to as 'scores') within the network are forwarded to deeper layers to relate respective activations to the original spatial resolution.

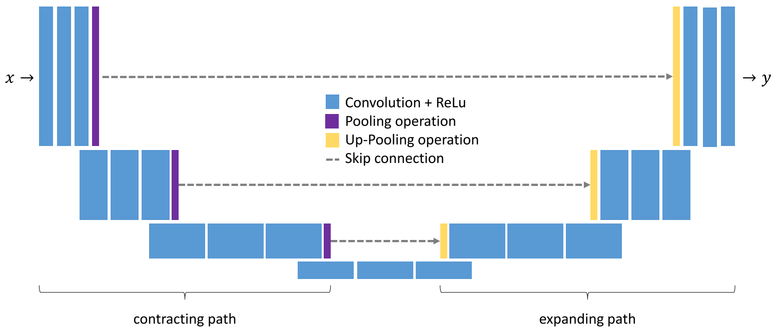


Figure 8: Schematic diagram of the *U-Net* architecture depicting its encoder-decoder structure using an contracting and expanding path.

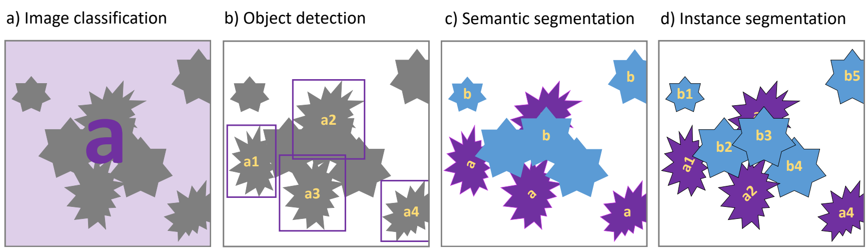
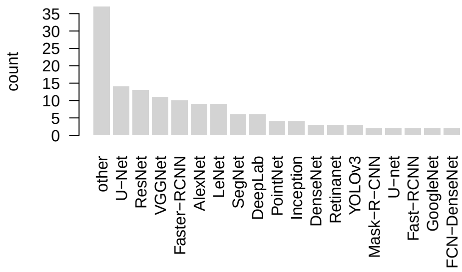
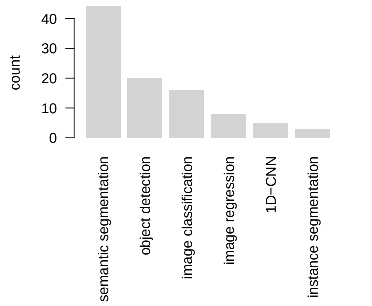


Figure 9: Schemes illustrating the conceptual differences between different CNN approaches, including a) image classification, where the entire image is assigned to a class; b) object detection, where individual occurrences are localized and their extent estimated with bounding boxes; c) semantic segmentation, which assigns each pixel of the input image to the target classes; and d) instance segmentation, where individuals belonging to a class are mapped.



(a) Architectures



(b) Approaches

Figure 10: Barplots characterizing the reviewed literature in terms of frequency of a) different architectures, including direct implementations as well as modifications of the original architecture and b) different approaches

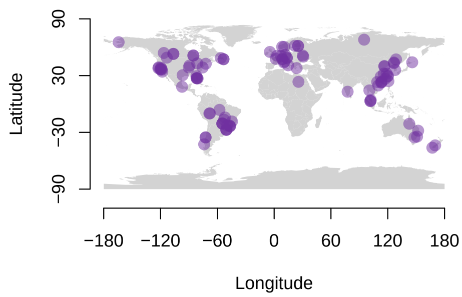


Figure 11: Study areas of the reviewed studies

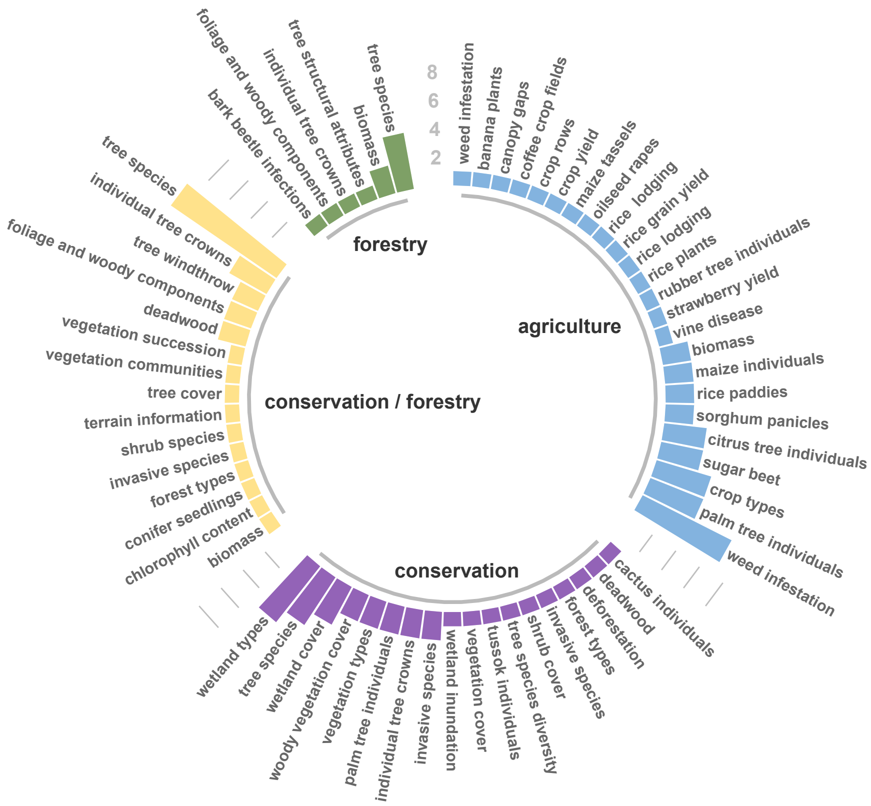


Figure 12: Frequency of studies in the context of agriculture, forestry, and conservation. The class *forestry/conservation* includes studies that are relevant for both fields.

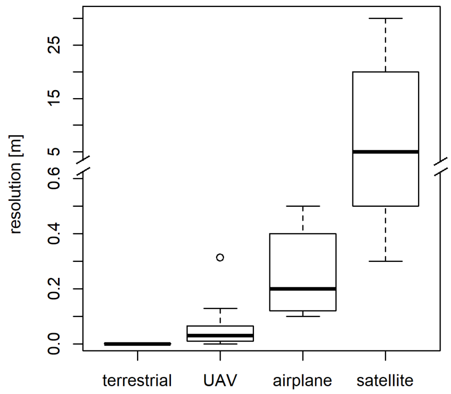


Figure 13: Frequency distribution of spatial resolutions by different remote sensing platforms among the reviewed studies (only raster products considered).

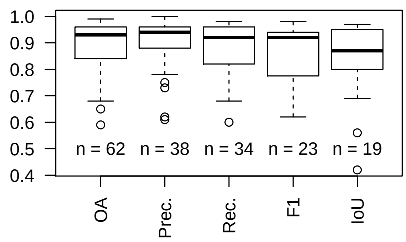


Figure 14: Validation results of the CNN-based predictions derived from the reviewed studies. The studies used different metrics (frequency = n), including Overall Accuracy (OA), Precision (Prec.), Recall (Rec.), F-score (F) and IoU (Intersect over Union).

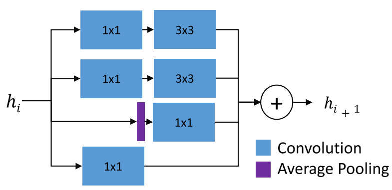


Figure 15: A schematic representation of an Inception-module