Robust Joint Stem Detection and Crop-Weed Classification using Image Sequences for Plant-specific Treatment in Precision Farming

Philipp Lottes
University of Bonn
Photogrammetry & Robotics Lab
Nussallee 15, 53115 Bonn
philipp.lottes@igg.uni-bonn.de

Jens Behley
University of Bonn
Photogrammetry & Robotics Lab
Nussallee 15, 53115 Bonn
jens.behley@igg.uni-bonn.de

Nived Chebrolu
University of Bonn
Photogrammetry & Robotics Lab
Nussallee 15, 53115 Bonn
nived.chebrolu@igg.uni-bonn.de

Andres Milioto
University of Bonn
Photogrammetry & Robotics Lab
Nussallee 15, 53115 Bonn
andres.milioto@igg.uni-bonn.de

Cyrill Stachniss
University of Bonn
Photogrammetry & Robotics Lab
Nussallee 15, 53115 Bonn
cyrill.stachniss@igg.uni-bonn.de

Abstract

Conventional farming still relies on large quantities of agrochemicals for weed management which have several negative side-effects on the environment. Autonomous robots offer the potential to reduce the amount of chemicals applied, as robots can monitor and treat each plant in the field individually and thereby circumventing the uniform chemical treatment of the whole field. Such agricultural robots need the ability to identify individual crops

and weeds in the field using sensor data and must additionally select effective treatment methods based on the type of weed. For example, certain types of weeds can only be effectively treated mechanically due to their resistance to herbicides, whereas other types can be treated trough selective spraying. In this article, we present a novel system that provides the necessary information for effective plant-specific treatment. It estimates the stem location for weeds, which enables the robots to perform precise mechanical treatment, and at the same time provides the pixel-accurate area covered by weeds for treatment through selective spraying. The major challenge in developing such a system is the large variability in the visual appearance that occurs in different fields. Thus, an effective classification system has to robustly handle substantial environmental changes including varying weed pressure, various weed types, different growth stages, changing visual appearance of the plants and the soil. Our approach uses an end-to-end trainable fully convolutional network that simultaneously estimates plant stem positions as well as the spatial extent of crop plants and weeds. It jointly learns how to detect the stems and the pixel-wise semantic segmentation and incorporates spatial information by considering image sequences of local field strips. The jointly learned feature representation for both tasks furthermore exploits the crop arrangement information that is often present in crop fields. This information is considered even if it is only observable from the image sequences and not a single image. Such image sequences, as typically provided by robots navigating over the field along crop rows, enable our approach to robustly estimate the semantic segmentation and stem positions despite the large variations encountered in different fields. We implemented and thoroughly tested our approach on images from multiple farms in different countries. The experiments show that our system generalizes well to previously unseen fields under varying environmental conditions—a key capability to deploy such systems in the real world. Compared to state-of-the-art approaches, our approach generalizes well to unseen fields and not only substantially improves the stem detection accuracy, i.e., distinguishing crop and weed stems, but also improves the semantic segmentation performance.

1 Introduction

Large amounts of agrochemicals such as pesticides, herbicides, and fertilizer are currently being used to satisfy the increasing demand of a growing population given the limited amount of arable land. These chemicals,

however, can have a negative impact on the environment by polluting the groundwater and thereby affecting human health. Therefore, reducing the amount of agrochemicals by using them more effectively is a major goal towards attaining a sustainable crop production. One of the solutions to achieve a reduction in the usage of agrochemicals is given by selectively treating individual plants based on its requirement, as opposed to treating the whole field uniformly with the same dose. However, until now this kind of targeted plant-treatment and weed control is done mostly manually and is usually a very labor-intensive task.

In this context, agricultural robots performing continuous per-plant monitoring and targeted treatment offer an attractive solution for drastically reducing the amount of chemicals applied. Additionally, such robots can cover large areas and therefore provide timely more frequent and spatially more dense information about the plants. Also, as these robots can be equipped with different actuators for weed control, such as selective sprayers, mechanical tools, or even lasers, it allows for performing plant-specific treatments only at places where it is needed. Equipped with a variety of tools, the robot can choose the most effective treatment based on the perceived type of the weed. For example, precise mechanical and laser-based treatments are most effective when applied to the stem location of dicot kinds of weed which have a well defined stem. In contrast, grass-like and bigger weeds are most effectively treated by spraying the agrochemicals over their entire leaf area. Thus depending on the type of weeds, both the spatial extent of a weed as well as its stem location are crucial information to effectively guide the robots' actuation system.

To realize such a selective and plant-dependent treatment, farming robots need an effective plant classification system. Such a system needs to reliably identify both, the stem location of dicot weeds (weeds whose seeds have two embryonic leaves) and also the cover of grass-like weeds given by its leaf area. Moreover, such systems must robustly work on different fields with minimal or no retraining effort. This generalization capabilities to new fields is essential for actually deploying precision farming robots with selective intervention capabilities in the real world (Slaughter et al., 2008).

In this paper, we address exactly this problem such that a robot can perform targeted, plant-specific treatment on different fields with no or minimal re-training of the classifier. To this end, we exploit geometric patterns that result from the fact that crops are usually sowed in rows. Within a field of row crops (such as sugar beets or corn), the plants share a similar lattice distance along the row, whereas weeds appear more randomly. In contrast to the visual cues, this geometric signal is much less affected by changes in the visual appearance. Thus, we propose an approach to exploit this geometric information as an additional signal by analyzing image sequences that cover a local strip of the field surface in order to improve the classification performance.

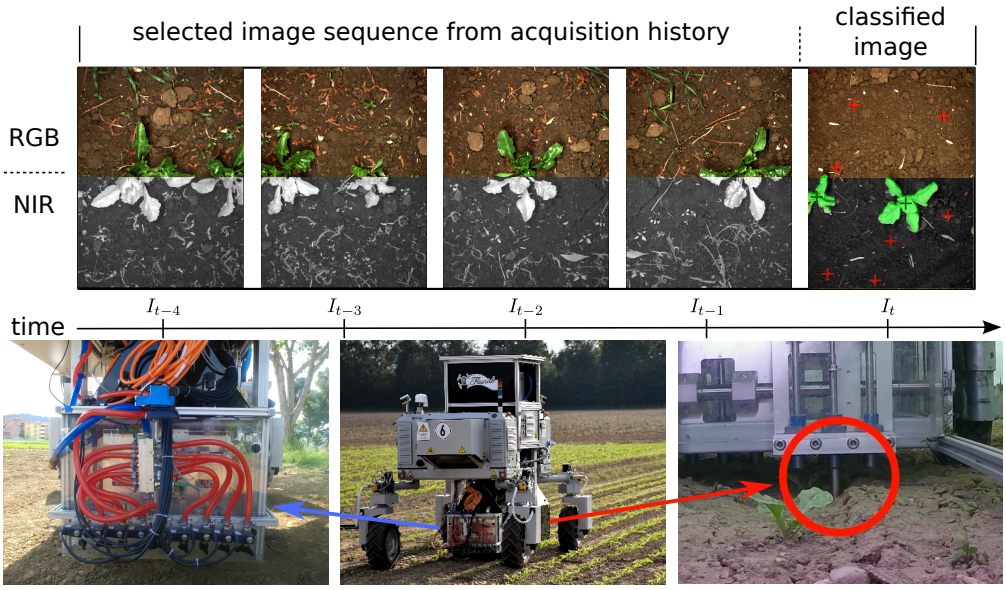

Figure 1: Given a sequence of images, $I_{t-4}, \ldots, I_t$, at current timestamp t, our approach determines jointly a segmentation map (green for crop, red for weed, blue for grass) and stem positions (crosses), as shown for I_t. With this information about weed coverage and stem positions, the agricultural robot selects the appropriate treatment, such as spraying for grass (shown on the left) and stamping (shown on the right) for weed with stems.

Fig. 1 depicts the farming robot employed during operation in the field and an example of an image sequence consisting of 4-channel images, i.e., conventional red, green, and blue (RGB) channels as well as an additional near infra-red (NIR) channel. Also shown is the desired output for the current image at timestamp t, which is given by a segmentation mask and stem positions.

The main contribution of this paper is an end-to-end trainable pipeline for joint plant stem detection and pixel-wise plant segmentation that operates on image sequences and through this incorporates information about the plant arrangement into the classification process. We employ a fully convolutional neural network (FCN) architecture sharing the visual code, i.e., the encoded representation of the image content, for the specific tasks such as the semantic segmentation of crops, dicot-weeds, grasses, and soil as well as the stem detection of the individual crops and dicot-weeds used for mechanical removal. More specifically, we jointly estimate the pixel-wise segmentation into the classes (1) crop, (2) dicot-weed, (3) grass-weed, and (4) background, i.e., mostly soil, and estimate the stem locations of crops and dicot-weeds at the same time. We extend our architecture by adding a sequential module enabling the usage of image sequences to implicitly encode the local geometry. This combination leads to better generalization performance even if the visual appearance or the growth stage of the plants changes between training and test time. Our system is trained end-to-end and relies neither on pre-segmentation of the vegetation nor on any kind of handcrafted

features.

As an extension to our previous work (Lottes et al., 2018a) on which this article is based upon and inspired by our previous work (Lottes et al., 2018b), we exploit additional sequential information for the joint crop-weed classification and stem detection. In addition to the evaluation in prior work, we furthermore perform an extensive evaluation of the system on a larger dataset containing image sequences from three countries and explicitly evaluate here the generalization capabilities of the proposed system to unseen fields.

In sum, we make the following three claims: Our approach is able to (i) accurately determine the stem positions of crop and weed stems enabling weeding systems for precise interventions, (ii) segments the images into the classes crop, dicot-weed, grass-weed, and soil allowing for class-specific treatments, and (iii) it generalizes well in most of evaluated cases of the data acquired from previously unseen fields and classifies sugar beets at different growth stages without the need for retraining. We perform extensive experiments on data captured on different fields in Germany, Switzerland, and Italy to show the generalization capabilities of our approach.

2 Related Work

Estimating semantic information from sensor data is a relevant topic in robotics (Milioto and Stachniss, 2018) and computer vision (Leibe et al., 2008), since more than two decades (Papageorgiou et al., 1988). Various approaches have been proposed to analyze the environment around a robot in indoor (Stachniss et al., 2005) as well as outdoor for optimizing mapping, traversability analysis (Bogoslavskyi et al., 2013; Wurm et al., 2013), and navigation (Kümmerle et al., 2013), for pedestrian detection (Leibe et al., 2005), for autonomous driving (Behley et al., 2013), for face detection (Viola and Jones, 2001), and for various other applications. In the past, a variety of classification techniques such as Boosting methods (Freund and Schapire, 1997), support vector machines (SVM) (Boser et al., 1992), or random forests (Breiman, 2001) have been applied. Within the last 6 years however, deep learning has revolutionized the semantic interpretation of image or laser range data in a large number of domains, including the domain of precision agriculture.

While there has recently been significant progress towards robust vision-based crop-weed classification, many systems rely on handcrafted features (Haug et al., 2014b; Lottes et al., 2017a,b). These hand-crafted features are usually geared towards the crop and weed present in the specific application and usually involve tweaking of parameters to adapt them to a different situation. A classifier using handcrafted knowledge will always

be limited by the features employed and the information extracted by these features. Therefore, much of research focused on the development of more complicated non-linear classifiers, like the aforementioned SVMs, random forests, or boosting, to overcome the limitations of the features employed.

However, the advent of end-to-end trainable convolution neural networks (CNN) (Krizhevsky et al., 2012) also spurred interest in end-to-end learnable crop-weed classification pipelines (Cicco et al., 2017; McCool et al., 2017; Milioto et al., 2017, 2018; Mortensen et al., 2016; Potena et al., 2016) to overcome the earlier described limitations of handcrafted pipelines, since they allow to learn feature representations directly from the training data using backpropagation (Rumelhart et al., 1986). This richer feature representation aggregated over multiple layers of convolutions, pooling operations, and non-linearities enable the CNN to get away with simple linear classifiers on top of these more complex features as compared to the aforementioned simpler hand-crafted features.

For semantic crop-weed segmentation, CNNs are often applied in pixel-wise fashion operating on image patches provided by a sliding window approach. Using this principle, Potena et al. (2016) use a cascade of CNNs for crop-weed classification, where the first CNN detects vegetation and then only the vegetation pixels are classified by a deeper crop-weed CNN. McCool et al. (2017) fine-tune a very deep CNN (Szegedy et al., 2016) and attain practical processing times by compression of the fine-tuned network using a mixture of small, but fast networks, without sacrificing too much classification accuracy.

These pixel-wise approaches operating on small patches extracted from the image are limited to use only very local information present inside the patch. This limits the receptive field of the employed convolutions and therefore also the amount of context incorporated into the classifier. Our network architecture, inspired by SegNet (Badrinarayanan et al., 2017a), instead uses the whole image and therefore uses potentially also information from the whole image in higher layers. The work by Milioto et al. (2018) combines an effective end-to-end semantic segmentation also based on fully convolutional network (FCN) architecture with plant features, which are comprised of low-level image features, like a vegetation index. They also show that the network can be fine-tuned to novel data using only very few labeled images.

Several works have focused on identifying the stem locations of the plants. Most of these approaches are also based on hand-crafted heuristics targeted towards specific applications. Kiani and Jafari (2012) use hand-crafted shape features selected through a discriminant analysis to differentiate corn plants from weeds and identify stem positions of the plants as the centroid of the detected vegetation. This leads to sub-optimal results particularly when the plant shapes are not symmetric or multiple plants are overlapping. Midtiby

et al. (2012) present an approach tailored for sugar beet plants by detecting individual leaves and use the contours of the leaves for finding the stem locations. However, such approaches usually fail to locate the stems in the presence of occluded leaves or overlapping plants.

Moving in the direction of a data driven approach, Haug et al. (2014a) propose a system to detect plant stems using keypoint-based random forests. They use a sliding window based classifier to predict stem regions by using several hand-crafted geometric and statistical features. Their evaluation shows that the approach often misses several stems for overlapping plants or generates false positives for leaf areas which locally appear to be stem regions. Kraemer et al. (2017) aim at addressing this issue by increasing the field of view of the classifier using a fully convolutional networks (FCN) (Long et al., 2015). The goal of their work is to identify plant stems over a temporal period allowing them to use the stem locations as landmarks for localization.

In our earlier work (Lottes and Stachniss, 2017), we also employed geometrical features and the plant arrangement for more robust crop/weed classification. However, we explicitly modeled the plant arrangement prior. In our follow-up approach (Lottes et al., 2018b), we integrated sequential information using a so-called sequential module that computes sequence features from an image sequence. Here, a 3D convolution aggregates information from the image sequence, which is generated by an encoder network. We show that this learned representation enables our approach to generalize well to unseen fields and also provide insights into the sequential information learned using simulation.

Our work overcomes many of the limitations by taking a holistic approach by jointly detecting stems and estimating a pixel-wise segmentation of the plants based on FCNs. Moreover, we explicitly distinguish crop and dicot stems, since it enables plant-specific treatment, for example fertilizing a crop or destroying a weed mechanically.

3 Approach

Our approach provides a semantic segmentation into the classes crop, dicot weed, grass weed, and soil as well as the stems positions for dicot weeds and crops, even in cases when the visual appearance of the plants, but also soil has changed substantially. The estimated pixel-wise labels of the semantic segmentation are important to determine the area for weed removal using selective spraying and the stem positions are a prerequisite for selective, high precision weed removal actions like mechanical stamping or laser-based weeding.

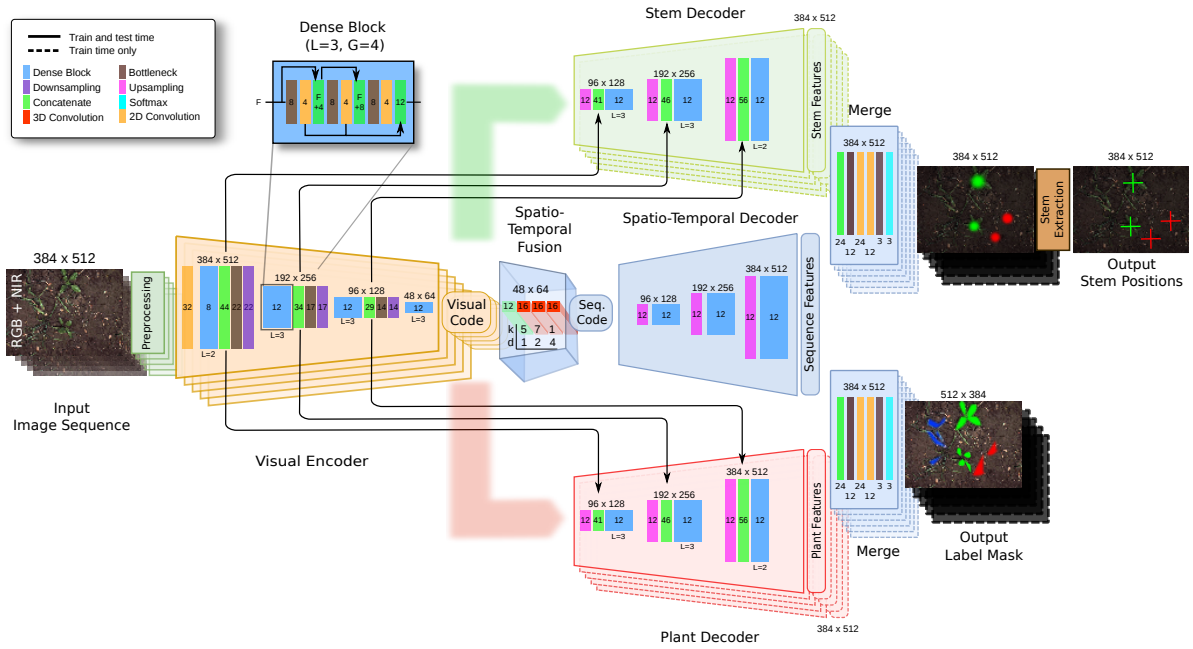

Figure 2: Architecture for the joint sequential stem and crop/weed classification network. Separate pathways for crop-weed classification and stem detection use only a single image. The resulting feature maps are combined with sequence features produces by the sequential module, which takes a sequence of images into account.

We propose an approach for this task based on fully convolutional networks (FCN), see also Fig. 2 for an overview of the network architecture. The proposed task-specific decoder networks for detecting stem regions and performing pixel-wise segmentation segmentation share a single encoder. This enables the network to learn better features in the shared encoder compared to naively learning separate encoders for both respective tasks. Sharing the backbone for different tasks on the same images showed better results also in other approaches (He et al., 2017).

To improve the generalization capabilities, we exploit information about the regular spatial plant arrangement induced by the planting process, where seeds are sown at regular distances by using specialized machines. We let the classification model learn this information from image sequences, which represents parts of the crop row in the field. We then fuse the arrangement information with the visual features to obtain a better classification and stem localization performance. By using this combination of sequential and visual features, we can improve the performance and generalization capabilities of our joint stem detection and segmentation pipeline. (Lottes et al., 2018b) have shown, that the geometric signal of the plant arrangement can be exploited by integrating sequential information to improve the generalization capabilities of an FCN for semantic segmentation in crop rows.

We now describe the proposed network architecture and the key elements of our approach as well as the strategy for the network training.

3.1 Architecture

Fig. 2 shows the network architecture and the information flow from the input to the output for a sequence $\mathcal{I}$ of length $S = 5$. Conceptually, we have S encoder-decoder networks that take only a single image as input and produce a single segment mask (either pixel-wise class labels or stem locations). To integrate sequence information, we use the so-called sequential module (Lottes et al., 2018b) that takes S encoder feature volumes and produces sequence features. The sequence features are then merged with the outputs of the decoders for plant segmentation and stem detection.

In the following discussion, we will not distinguish between the plant and stem decoders, since they are architecturally the same. But note that the decoders for stems and plant segmentation do not share weights. We will use the term visual decoder to denote both decoders.

The visual encoder of the FCN shares its weights over different timesteps such that we reuse it as a global feature encoder for each image from the sequence respectively. Thus, we compute S visual codes that are a compressed, but highly informative representation of the input images. We now use the visual code for three different paths within the architecture. First, each visual code is passed to the decoder resulting in S decoded visual features volumes. Note that the decoders for one branch, i.e, plant or stem, also share the weights internally. Therefore, we can treat them as respective encoder-decoder FCNs which are applied to each image respectively. Second, we pass all visual code volumes to the so-called sequential module. The sequential module now aggregates the S visual codes using 3D convolutions and outputs a single sequence code, which contains information about the sequential content. The sequence code is then upsampled by a spatio-temporal decoder to the match the (image) resolution of the visual features. The visual feature maps of the aforementioned visual decoders and the sequential feature maps are then merged and classified to obtain the desired label mask output.

In contrast to our prior work (Lottes et al., 2018b), we found that an aggregation of S sequential codes resulting from the 3D convolutions leads to same or even better results as compared to taking S independent sequential codes. By this aggregation. we only need a single sequence decoder and thus effectively reduce the size of the model on the GPU. Note that the number of parameters stays the same, as these parameters are shared. We explicitly compare the aggregated approach with the original approach (Lottes et al., 2018b)

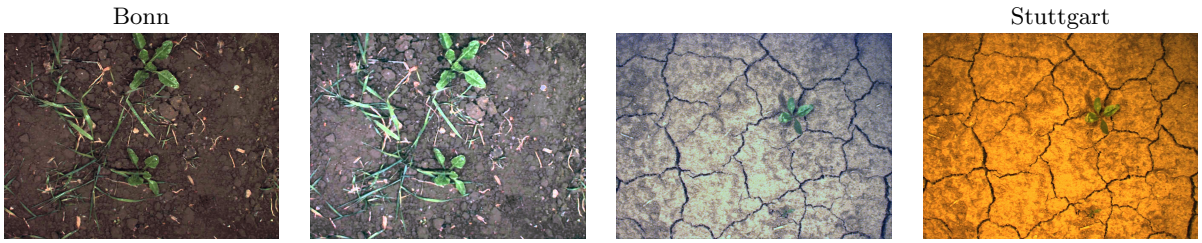

Figure 3: Left: RGB image from Bonn dataset. Right: RGB image from Stuttgart dataset. Center: respective RGB images after preprocessing.

within our experimental section to back this statement empirically.

The input to our network is given by a sequence of S RGB images with optional near infra-red information. The output consists of two label masks, where each pixel corresponds to the probability of the respective class labels. More specifically, the first output is the plant mask reflecting the pixel-wise semantic segmentation with classes crop, dicot weed, grass weed, and soil. The second output is the stem mask segmenting regions within the image corresponding to crop stems and weed stems. Note, that we try to predict the area of the stem instead of regressing the stem location. This is the key for using the same architecture for learning plant classification and stem locations. Finally, we extract pixel-accurate stem positions from the stem mask using a post-processing step.

3.2 Preprocessing

Preprocessing the input can help to improve the generalization capabilities of a machine learning approach by transforming the test data distribution such that it is more similar to the distribution of the training data. The objective of our preprocessing is to minimize the influence of changes in the environment on the inputs as much as possible. We perform the preprocessing independently for each image and moreover separately on all channels, i.e. red, green, blue, and near infra-red. For each channel, we (i) remove noise by performing a Gaussian blur using a $[5 \times 5]$ kernel given by the standard normal distribution, i.e., $\mu = 0$ and $\sigma^2 = 1$, (ii) standardize the channels by subtracting the mean of channel values and dividing by standard deviation of the channel values, and (iii) normalize and zero-center the channel values to the interval $[-1, 1]$ as it is commonly done when training neural network. Fig. 3 shows qualitatively the effect for the exemplary images from our used datasets, which were captured in different lighting condition using varying sensor setups.

3.3 Encoder-Decoder FCN

FCNs for semantic segmentation tasks (Long et al., 2015) achieve high performance for a variety of tasks (Badrinarayanan et al., 2017b; Paszke et al., 2016; Ronneberger et al., 2015; Huang et al., 2017). Commonly, FCN architectures employ the so-called "hourglass" structure referring to a downsampling in the encoder followed by a complementary upsampling in the decoder to regain the full resolution of the input image for pixel-wise segmentation. We design our network architecture with the aim to run the classification near real-time such that an actuator can directly act upon the classification result. Additionally, we use a comparably small number of learnable parameters compared to the state-of-the-art networks to obtain a model capacity which is sufficient for the four class prediction problem. Therefore, our network is more lightweight than other pipelines.

A basic building block in our encoder-decoder FCN is the so-called Fully Convolutional DenseNet (FC-DenseNet) proposed by (Jégou et al., 2017). The FC-DenseNet combines the recently proposed densely connected CNNs organized as dense blocks (Huang et al., 2017) with fully convolutional networks. The key idea is to enable a dense connectivity pattern which iteratively concatenates all computed feature maps of subsequent convolutional layers with features maps from before. These "dense" connections enable deeper layers to reuse features produced by earlier layers.

We define our 2D convolutional layer as a composition of the following components: (1) 2D convolution, (2) rectified linear unit (ReLU) as non-linear activation, (3) batch normalization (Szegedy et al., 2016) and (4) dropout (Srivastava et al., 2014). We repeatedly apply bottleneck layers and thus keep the number of feature maps small while achieving a deep architecture. Our bottleneck layer is a 2D convolutional layer applying the convolutions with a $[1 \times 1]$ kernel (Lin et al., 2014).

A dense block is given by a stack of L consecutive 2D convolutional layers convolving feature maps with the same spatial size. Fig. 2 conceptually shows a dense block and how the information is propagated. The input of the l^{th} 2D convolutional layer is given by a concatenation of all feature maps produced by the previous layers $l-1, ..., 0$ inside the dense block, whereas the output feature volume is given by the concatenation of the newly computed feature maps. The number of the resulting feature maps is commonly called the growth rate G of a dense block (Huang et al., 2017). Consequently, we use $2G$ kernels in the bottleneck layers to reduce the computational cost in the subsequent 2D convolutional layers.

Fig. 2 illustrates the data flow through the FCN. The first layer in the encoder is a 2D convolutional layer

augmenting the 4-channel images using 32 $[5 \times 5]$ kernels. The following operations in the encoder are given by a recurring composition of dense blocks, bottleneck layers and downsampling operations, where we concatenate the input of a dense block with its output feature maps. We downsample the features maps with a strided convolution employing 2D convolutional layers with an $[5 \times 5]$ kernel and a stride of 2. All bottleneck layers between dense blocks compress the feature volumes by halving the feature volume along the feature axis in a learnable way. In the decoder, we revert the downsampling using strided transposed convolution (Dumoulin and Visin, 2018) with a $[2 \times 2]$ kernel and a stride of 2.

To facilitate the recovery of spatial information, we use skip connections and concatenate feature maps produced by the dense blocks in the encoder with the corresponding feature maps produced by the upsampling in the decoder and feed both feature volumes into a bottleneck layer to fuse them. In contrast to the encoder, we reduce the increase of the number of feature maps within the decoder by omitting the concatenation of a dense blocks input with its respective output.

For learning, we use a multi-task loss L combining the loss for the plant segmentation L_{plant} and for the stem region segmentation L_{stem}, i.e.,

$$L = (1 - \alpha) \cdot L_{\text{stem}} + \alpha \cdot L_{\text{plant}}, \tag{1}$$

where we use $\alpha = 0.5$ as it showed the best results on the validation set. Here, L_{plant} is a weighted cross entropy loss, where we penalize errors differently using a weight depending on count over the class labels. L_{stem} is a loss based on an approximation of the intersection over union (IoU) metric as it is more stable with imbalanced class labels (Rahman and Wang, 2016), which is the case in our problem with under-represented stems as compared to the amount of soil. The multi-task loss enables to share information for learning the encoder, since it can use the loss information from all decoders in the backward pass of the backpropagation.

3.4 Sequential Module

Fig. 4 shows how the sampling of the $S = 5$ images is performed, in order to build the sequence $\mathcal{I} = \{I_t, \ldots, I_{t-4}\}$ which our pipeline uses as an input. It also shows an overlay of each image with the predictions from our approach. In order to maximize the spatial information that we input to the network while minimizing the computational cost of running the approach, we subsample the images from the recorded data stream along the transversal trajectory. We maximize the area covered by the images in object-space while using only a minimum number of images. To achieve this, we use the odometry information from the

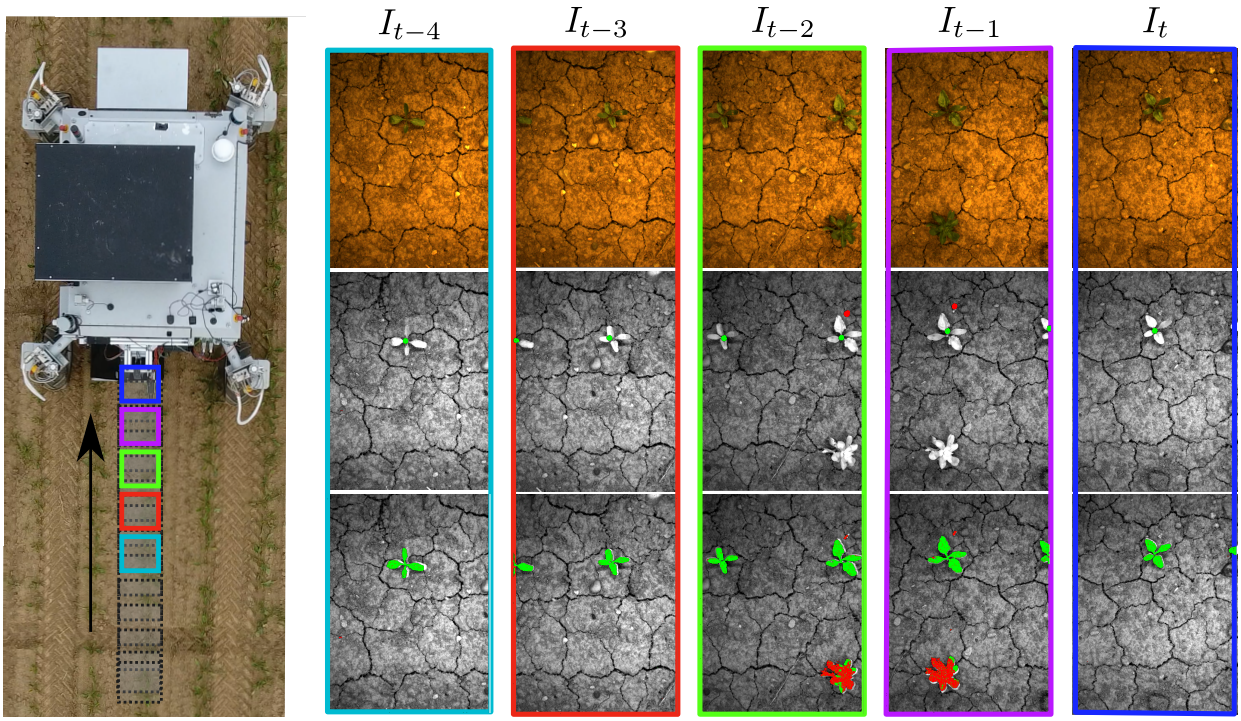

Figure 4: Left: BoniRob capturing images while driving along a crop row. Our approach exploits an image sequence by selecting those images from the history that maximize the spatial coverage in object-space while using a minimum number of images. Right: Exemplary prediction of crop vs. weed and stems for an image sequence captured on a field near Stuttgart where the classification model was solely trained on data acquired in Bonn; top row: RGB images; middle row: predicted label mask projected on the image (crop in green, weed in red, background transparent in the upper part) and stem locations (lower part); bottom row: ground truth, where stem locations are given by filled circles.

robot, along with the calibration parameters of the camera, and an estimation of the height above ground plane given by the rigid position of the camera on the platform, which is a good-enough approximation to calculate the candidates for the image sequence.

Our approach exploits the crop-weed arrangement along the crop-row caused by the regular seed placement during the sowing process. To exploit this information, our network needs to be able to observe and exploit the whole sequence corresponding to a crop row. Thus, our main architectural design contribution is the sequential module, which enables this type of learning through the exploitation of sequential information. This module represents an additional pathway for this type of spatial information to flow in the network, and consists of three subsequent parts, i.e. the (i) spatio-temporal fusion, the (ii) spatio-temporal decoder and the (iii) merge layer.

The core module of the sequential processing is the spatio-temporal fusion step. First, we extract the visual code feature volumes for each image in the sequence, and concatenate them along an additional time dimension. Second, we combine them in the spatio-temporal feature volume, i.e. the sequence code, by processing

the stack of visual features with a set of 3D convolutional layers. We define the 3D convolutional layer analogously to the 2D convolutional layer, i.e. as a composition of convolution, ReLu, batch normalization and dropout. In each 3D convolutional layer, we use 16 3D kernels with a size of $[5 \times 5 \times S]$ to allow the network to learn weight updates under consideration of the whole input sequence. We apply the batch normalization to all feature maps jointly regardless of their position in the sequence.

Another design choice which aids the usage of the spatial information is increasing the receptive field of the subsequent 3D convolutional layers in spatial domain. This allows the network to exploit longer range dependencies in the crop-row structure. In order to do this, we increase the kernel size k and the dilation rate d of the 3D kernels for subsequent 3D convolutional layers. Note that we only increase k and d only for the spatial domain of the convolutional operation, i.e. $[k \times k \times S]$ with $k = \{5, 7, 1\}$ and $[d \times d \times 1]$ with $d = \{1, 2, 4\}$. The usage of this dilated 3D convolutions in the time domain translates into a larger receptive field of the spatio-temporal fusion in space, given the motion of the robot and the sampling of the images. This allows the model to consider the whole encoded content of all the images along the sequence. In our experiments, we show that the model gains performance by increasing this dilation in the time domain within the spatio-temporal fusion.

The second part of the sequential module is the spatio-temporal decoder, which upsamples the sequence code to the desired resolution matching it to the output of the other decoders, and the input. Analogous to the visual decoder, the decoding is performed by recurrently applying upsampling and bottleneck layers, followed by a dense block until it reaches the required spatial resolution. This process generates the pixel-wise sequence feature map. In order to increase the capacity of the model, we neither share weights between both pathways nor connect them via skip connections with the encoder of the FCN.

The last building block of the sequential module is the merge layer. Its main objectives are to merge the visual features with the sequence features and to compute the label mask as the output of the system. First, we concatenate the input feature volumes along their feature axis and pass the result to a bottleneck layer using 12 kernels, where the actual merge takes place. Then we pass the resulting feature volume through a stack of two 2D convolutional layers. Finally, we convolve the feature volume into the label mask using a bottleneck layer with 3 kernels for respective class labels and perform a pixel-wise softmax along the feature axis. Unlike previous work (Lottes et al. (2018b)), which generates a spatio-temporal feature volume for each image in the sequence, this architecture condenses all sequential information into one feature volume, which is then shared for all the images in the sequence. Our experiments show that this allows for more robust encoding of the crop structure, resulting in better generalization results, as shown in our ablation

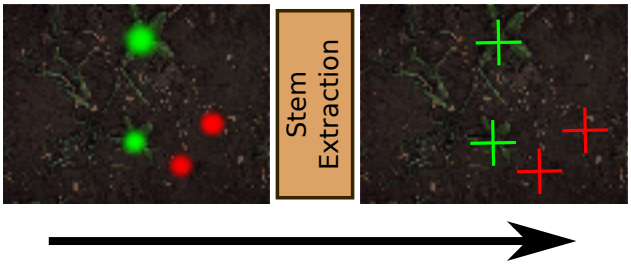

Figure 5: We extract the pixel-wise stem locations by computing a weighted center of mass of the predicted stem regions by the FCN.

study.

For more in-depth implementation details refer to Fig. 2, which contains the details of all the layers, parameters, and sizes used throughout the architecture. Complementary to this, in Sec. 4, we provide an evaluation of each architectural design choice.

3.5 Stem Extraction

Once the pixel-wise stem mask is obtained from its decoder in the FCN (see Fig. 5), i.e., $P(y|\mathbf{x})$ with $y \in \{\text{soil}, \text{crop}, \text{dicot weed}\}$ for each pixel $\mathbf{x}$, we need to extract an accurate stem location for both the crops and the dicot weeds. To achieve this, we first calculate the class with highest label probability for each pixel, i.e., $y^* = \text{argmax}_y P(y|x)$. Next, we determine the connected components $\mathcal{X}_j^c$ for each class c and compute the weighted mean $\bar{\mathbf{x}}_j^c$ of the pixel locations by

$$\bar{\mathbf{x}}_j^c = \frac{\sum_{\mathbf{x} \in \mathcal{X}_j^c} P(y = c|\mathbf{x}) \cdot \mathbf{x}}{\sum_{\mathbf{x} \in \mathcal{X}_j^c} P(y = c|\mathbf{x})}. \tag{2}$$

The weighted means $\bar{\mathbf{x}}_j^c$ for class c are then the stem detections reported by our approach.

4 Experiments

Our experiments are designed to support three main claims: (i) Our approach is able to detect the stem locations of crops and dicot-weeds, and simultaneously (ii) segment the images into crop, dicot-weed, grass-weed and soil classes, and (iii) generalize well to new fields despite the large change in visual appearance of the plants and the soil without the need for re-training the model.

Figure 6: Different versions of the BOSCH Deepfield BoniRob used for data acquisition and deployment. Left: BoniRob V2 used for data acquisition in Stuttgart. Middle: BoniRob V3 used for data acquisition in Bonn. The acquired data is published in (Chebrolu et al., 2017). Right: BoniRob V3 equipped with the weeding module for selective spraying and stamping.

In the following sub-sections, we first describe the various agricultural robots and the camera setup used for the data acquisition including the image data which is used as input to our classification system. Furthermore, we describe the metrics used for the performance evaluation and introduce the various datasets used in our experiments.

4.1 Field Robot and Camera System

Our experiments have been conducted with different generations of the BoniRob platform, shown in Fig. 6. BoniRob was built by BOSCH DeepField Robotics as a multi-purpose field robot for research and development applications in precision agriculture such as weed control, plant phenotyping and soil monitoring. In order to equip the robot with different tools for these tasks, the platform provides an empty installation space. BoniRob can be easily adapted to navigate in different fields as it is equipped with four independently steerable wheels and a mechanism to adapt the track width to the crop-row distance on the field. More details of the BoniRob are given in (Chebrolu et al., 2017).

We acquired the image data using a 4-channel JAI AD-130 GE camera pointing downwards on the field approximately $70 - 85$ cm above soil (depending on the version of the robot used). This camera system allows the acquisition of time synchronized and optically aligned images through a prism based mapping of the incoming light to the CCD arrays, one for RGB and NIR respectively. The RGB+NIR images of the JAI camera were captured with a resolution of 1296×966 pixels yielding a ground resolution of approximatively $3\,\frac{px}{mm}$. The field of view varies over the datasets from $20 - 30$ cm in driving direction and $35 - 50$ cm cross to it.

For all experiments in this paper, the camera was mounted on the robot under a shaded area in order to

Table 1: Information about the datasets. Soil *on* refers to the computation including the soil class and *off* vice versa. The number of potential sequences is given by #images$-(S-1)$

Parameter	Bonn	Ancona	Stuttgart	Eschikon
#images	2359	192	1686	65
Crop pixels (soil on/off)	2.7% / 68.7%	1.0% / 66.1%	0.8% / 71.7%	0.8% / 54.4%
Dicot pixels (soil on/off)	0.8% / 19.9%	0.2% / 13.1%	0.3% / 28.3%	0.6% / 44.0%
Grass pixels (soil on/off)	0.4% / 11.4%	0.3% / 20.3%	-	0.02% / 1.6 %
#Crop stems	3083	542	2117	41
#Dicot stems	19911	1019	2279	1130
Approx. crop size	2-10 cm^2	2-6 cm^2	6-8 cm^2	4-8 cm^2
Approx. dicot size	0.5-4 cm^2	0.5-4 cm^2	0.5-10 cm^2	0.5-5 cm^2
Approx. grass size	0.5-6 cm^2	1-6 cm^2	-	0.5-4 cm^2
light setup	LED	LED	Halogen	LED

acquire images independent of the natural light sources. The artificial light setup inside the shaded area, however, changed for the different versions of the BoniRob. In the initial version of the BoniRob, the artificial lighting is provided by a series of halogen bulbs which resulted in a "spotty" illumination of the scene, whereas in the later versions, a LED-tube based system consisting of diodes in the red, green, blue, and infra-red spectrum was used which provided a more uniform illumination. The images were captured with a frequency of 1 Hz while the robot was moving over the field with a speed of approximately $0,3\,\frac{m}{sec}$. Example images obtained from these different setups are shown in Fig. 7.

4.2 Datasets

In order to explicitly evaluate the generalization capabilities of the classification system to unseen fields, we gathered data from different fields located in different cities in different countries such as (1) **Bonn**, Germany, (2) **Ancona**, Italy, (3), **Stuttgart**, Germany, and (4) **Eschikon**, Switzerland. All datasets consist of sugar beet plants (crop) and dicot-weeds. Additionally, datasets from Bonn and Ancona also have grass-weeds. Note that no grass-weeds are present in the Stuttgart dataset and only a very small number is present in the Eschikon dataset.

Overall, the datasets represent challenging conditions for a vision-based classification system as they contain different dicot-weed and grass-weed types with varying sizes as well as different soil conditions. Furthermore, the image data differs also in color, brightness, and contrast, due to the changing light setups of the field robot.

The Bonn dataset forms the basis for our training data. It contains the most samples and holds a substantial amount of plants from all the considered classes, i.e crops, dicot-weeds and grass-weeds. From this dataset,

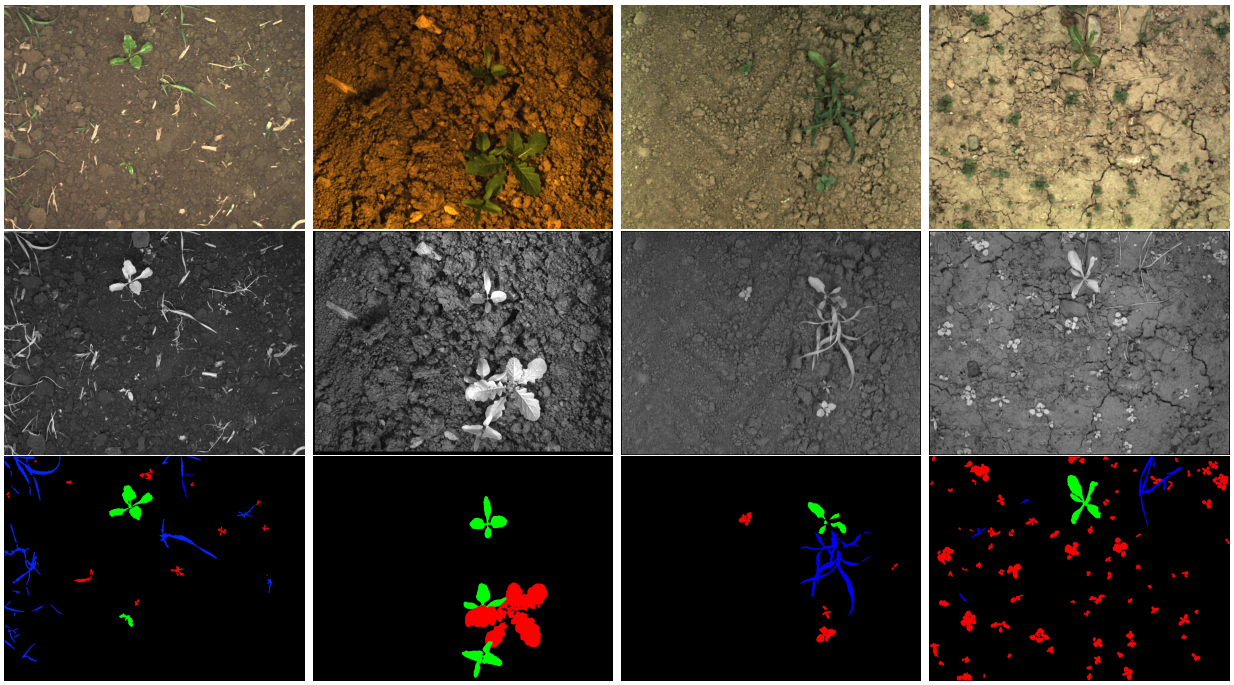

Figure 7: Example RGB+NIR images and the corresponding ground truth information for the segmentation task. From left to right: Bonn, Stuttgart, Ancona, and Eschikon. The dataset differ from each other in terms of environmental changes including varying weed pressure, various weed types, different growth stages of crops and weeds, changing visual appearance of the plants and soil, and illumination conditions.

we split 10% for the validation, e.g. hyperparameter search, and 10% for testing of our approach. All other datasets are entirely used as test datasets. Additionally, we annotated all datasets with pixel-wise class labels and stem locations.

Fig. 7 shows examples from each dataset and Tab. 1 summarizes their key statistics. We observe from the dataset statistics that the soil class represents 96% of the total pixels whereas all the vegetation classes put together only contribute to about 4%, which causes class imbalance problems for the learning algorithm. Moreover, the weed classes mostly contain comparably small objects making the segmentation task even more challenging as they are represented by only a few pixels in the image. In this paper, we only consider vegetation objects which are $>= 0.5\,\mathrm{cm}^2$ i the object-space.

4.3 Performance Metrics

We rely on different metrics to analyze the performance of our approach and to compare it to other approaches. We use the mean average precision (mAP) over the per-class average precisions (AP) (Everingham et al., 2010) as metric for our evaluation. The mAP represents the area under the interpolated precision-

recall curve. As noted by Everingham et al. (2010), a method must have a high precision at all levels of recall to achieve a high score with this metric. Thus, this is a preferred metric to be used to compare different approaches with a single number. A drawback of the mAP is that it is not intuitively interpretable in terms of the practical requirements of weeding systems. Therefore, we provide also the precision (P) and recall (R) values to give a performance measure reporting the actual percentage of correctly and/or wrongly classified crops and weeds in the fields.

For the stem detection task, a predicted stem is considered to be a positive detection if its Euclidean distance to the nearest unassigned ground truth stem is below a threshold $\theta = 10\,\text{mm}$. This threshold has been chosen keeping in mind the size of the mechanical stamping tool of the BoniRob. Furthermore, we compute the mean average distance (MAD) in object-space [mm] for all true positives to show the spatial precision of our approach.

For the segmentation task, we evaluate the performance in a (1) pixel-wise manner such that it corresponds to the coverage of the crops and weeds in the scene and in an (2) object-wise manner to report a measure on plant level. In the second case, we compare the predicted label mask with the crop and weed objects given by the class-wise connected components from the corresponding ground truth segments.

4.4 Comparison to Other Approaches

In order to understand the effects of various components in our approach, we compare the performance of our approach against several other methods. We refer to our proposed approach as *stem-seg-S*, which stands for stem-segmentation-sequential. First, we compare *stem-seg-S* against its non-sequential version as proposed in Lottes et al. (2018a) (see details in Sec. 3). We refer to this approach with *stem-seg*. We do this comparison in order to understand the gain in performance due to the sequential module as *stem-seg* has the same architecture without the sequential module. Both these approaches have the ability to jointly estimate the plant stems and perform semantic segmentation.

We also compare our current approach *stem-seg-S* against our previous approach proposed in (Lottes et al., 2018b), called *seg-S*, where we also exploit sequential information but did not compute the stem locations. Here want to observe, if the additional information induced by the stem detection task, helps to improve the semantic segmentation of the crops and weeds.

In a third comparison, we introduce an architectural design change, i.e., the aggregation of the features in

the sequential module (described in Sec. 3.4). We refer to this approach with *seg-S-mod*. In order to show that our modification leads to the same or even better results by requiring less computational resources, we explicitly evaluate its impact on the performance on the semantic segmentation and compare it to the original architecture *seg-S* proposed in (Lottes et al., 2018b).

Finally, we we report the performance when using a single image FCN for plant segmentation (*seg-only*) given by the visual encoder + plant decoder and a single image FCN for stem detection (*stem-only*) given by the visual encoder + stem decoder. We compare the performance of our approach against these two and show that sharing the encoder for stem detection and plant segmentation enables it to learn better features.

4.5 Parameters

As input to the network, we use images with a resolution of $W = 512$ and $H = 384$, which yields a ground resolution of around $1\,\frac{\text{px}}{\text{mm}}$. Based on our hyperparameter search, we selected the values for S, G, B for our architecture as mentioned in Sec. 3 and initialize the weights as proposed by He et al. (2015). In case of approaches considering images sequences as their input, we choose a sequence length $S = 5$ and use the RMSPROP optimizer with a mini-batch size of $B = 1$ leading to 5 images per mini-batch. For a fair comparison in terms of the batch-norm statistics, we choose a batch size of 5 for all non-sequential approaches. We use a weighted cross-entropy loss, where we penalize prediction errors for the crops and grass-weeds by a factor of 10 and for dicot-weeds by a factor of 20 to cope with the under-representation of those classes in the training data. We use dropout with a rate of one-third and set the initial learning rate to 0.01 and divide it by 10 after 10 and 25 epochs respectively. We stop the training for all approaches after 75 epochs in order to have the same training duration. For our proposed approach *stem-seg-S*, one training step takes around 0.7 seconds leading to a total training duration of around 39 hours. We implemented our approach using Tensorflow and it provides classification results with a processing rate of approximately 5 Hz (200 ms) on a NVIDIA Geforce GTX 1080 Ti GPU at inference time (without any optimizations).

4.6 Stem Detection Performance

In this first experiment, we show that our approach is able to accurately detect the stem locations of crops and dicot-weeds and provides state-of-the art performance in terms of generalization to new fields having a different visual appearance of the plants and the soil without the need for adapting the model through retraining. The main purpose of this experiment is to evaluate the performance of a crop-weed classification system in a real-world scenario where it is trained on a particular field but deployed on new unseen fileds.

Table 2: Stem Detection Performance. MAD in mm, precision (P) and recall (R) in percent.

Approach	Mean				Crop				Dicot			
	mAP	mP	mR	mMAD	AP	P	R	MAD	AP	P	R	MAD
Bonn (10%-split of the trainset)												
stem-seg-S	**85.4**	**90.2**	**91.6**	**1.6**	80.2	86.3	89.9	1.7	90.5	94.0	93.3	1.4
stem-seg	74.4	72.0	90.2	4.0	71.6	73.8	86.7	3.6	77.2	70.1	93.6	3.6
stem-only	53.3	67.4	80.8	3.8	52.1	69.1	76.4	3.9	54.4	65.7	85.2	4.3
Stuttgart												
stem-seg-S	**66.9**	**73.0**	**87.0**	**2.4**	64.7	75.2	84.0	1.9	69.0	70.8	90.0	2.8
stem-seg	46.5	53.1	76.7	4.5	51.4	54.8	82.6	3.6	41.6	51.3	70.7	5.3
stem-only	29.6	37.0	78.6	4.7	31.1	42.3	71.1	5.1	28.0	31.6	86.0	4.3
Ancona												
stem-seg-S	**42.9**	**66.1**	**60.0**	**1.9**	30.4	46.2	57.5	1.9	55.4	85.9	62.5	1.8
stem-seg	19.2	40.0	42.9	5.0	13.8	24.3	47.2	6.1	24.5	55.6	38.5	3.3
stem-only	12.1	26.5	39.1	4.0	15.6	37.5	35.2	4.7	8.5	15.5	42.9	3.8
Eschikon												
stem-seg-S	**50.1**	**72.0**	63.1	**2.7**	52.0	94.9	53.6	1.9	48.1	50.0	72.5	3.4
stem-seg	33.7	49.9	57.1	5.6	42.4	79.8	46.0	3.6	24.9	20.0	68.2	3.0
stem-only	26.3	47.7	**70.9**	4.6	15.7	13.0	97.6	6.2	36.8	82.3	44.2	6.1

In our experimental setup, we use the data gathered in Bonn as the training dataset and evaluate the performance on a 10%-split of the Bonn dataset as well as other datasets that are entirely used for testing only. Tab. 2 summarizes quantitatively the obtained results.

In all datasets, we see that our approach outperforms the competing approaches in terms of the mAP. The difference in mAP is induced by a better AP for both crops and dicot weeds. The performance gain on the Bonn data is around 10%, whereas the gain on the other test sets is around 20%. This clearly indicates that additional sequential information aids the stem detection performance as well as the generalization capabilities to new field with different underlying data distributions.

In terms of the recall, we obtain 90% for dicots and 84% for crops in the Stuttgart dataset. Similar results are obtained for Bonn dataset with 93% (dicot) and 90% (crop), where most of the stems are in the dataset are detected. The drop in precision of about 10% on the Stuttgart data is mostly due to the false detections of stems which are predicted in soil regions where no plant actually exists. The datasets from Ancona and Eschikon are more challenging where the overall performance drops around 20%. In comparison to other non-sequential approaches, our approach still performs substantially better.

With regards to the MAD for stems, we report an improvement of at least two times as compared to other approaches. In the worst case our approach (*stem-seg-S*) improves the mMAD from 4.6 mm to 2.7 mm (Eschikon dataset). Averaging over all datasets, we obtain a mMAD of 2.2 mm which is sufficient for precise mechanical treatments, but

also, for the even more precise laser-based weeding application. The sequential information aids the MAD directly by exploiting the (strong) geometric signal in the data and indirectly by improving the recall and precision for the stem detection.

In Fig. 8, we illustrate qualitative results of our approach for all datasets respectively. We see that most of the stems both for crops and for dicot weeds are detected correctly. Moreover, the results show that our approach is able to detect very small dicot weeds which are of a size of around $0.5\,\mathrm{cm}^2$ and are only represented by a few pixels in the image. However, in the Eschikon dataset a non-negligible amount of weeds are not detected reliably.

4.7 Segmentation Performance

The second experiment is designed to show that our approach provides a pixel-wise semantic segmentation of the scene into crops, dicot-weeds, grass-weeds, and provides state-of-the-art performance in terms of its generalization capabilities to new field environments. The training and test data setup is the same as for the first experiment. Tab. 3 illustrates the performance achieved for the semantic segmentation task.

Except for the Eschikon dataset, our approach outperforms the other methods in terms of the mAP and P as well has similar recall values as the individual best approaches on the respective datasets. These results clearly show that the methods using the sequential information have better generalization capabilities than the non-sequential ones. In sum, our approach gains on average around 13% in mAP, 10% in mP, 29% in mR compared to the non-sequential *stem-seg* approach in the cross dataset evaluation.

When comparing our approach *stem-seg-S* against our previous approach proposed in (Lottes et al., 2018b), called *seg-S*, we obtain a better segmentation performance in three out of four datasets (Bonn, Stuttgart, Ancona). Even on fourth dataset (Eschikon), the performance is similar. This indicates that the additional information induced by the stem detection task helps to improve the semantic segmentation of the crops and weeds.

To evaluate the effect of our architectural design change, i.e., the aggregation of the features in the sequential module (described in Sec. 3.4), we compare the results of *seg-S-mod* against *seg-S*. We can observe that in most cases, except for the Eschikon data, *seg-S-mod* outperforms *seg-S*. We see an average improvement of around 4% in terms of the mAP for the Bonn, Stuttgart, and Ancona datasets. This indicates the advantage of our architectural design choice as we obtain on average better performance while computing only 20% of the features in the spatio-temporal decoder.

In Fig. 8, we illustrate qualitative results of our approach on sample images from all datasets. Overall, we observe that the semantic segmentation for all classes is visually good. However, there are small areas particularly on crop plants and grass weeds, which are classified falsely as dicot. Since the total area of dicot weeds is small compared to the crops, even small error regions in the crop leads to a substantial drop in the segmentation metrics for dicot weeds.

Table 3: Segmentation Performance. mAP, precision (P) and recall (R) in percent.

Approach	Overall			Crop			Dicot-Weed			Grass-Weed		
	mAP	mP	mR	AP	P	R	AP	P	R	AP	P	R
Bonn (10%-split of the trainset)												
stem-seg-S	**69.7**	**76.7**	89.9	91.0	92.5	95.6	69.7	75.3	87.4	48.4	62.3	86.8
stem-seg	57.6	67.1	79.9	91.9	93.5	97.9	41.9	53.9	74.1	38.9	54.0	67.8
seg-S	64.6	69.8	**93.1**	91.8	92.4	96.8	58.9	64.6	92.0	43.2	52.5	90.6
seg-S-mod	69.1	74.7	86.6	91.6	92.4	93.7	60.9	66.9	93.5	54.7	64.7	72.5
seg-only	61.0	68.7	85.4	89.1	90.3	94.6	44.6	55.2	83.9	49.4	60.6	77.7
Stuttgart												
stem-seg-S	**58.9**	**68.8**	**72.2**	77.7	84.4	85.5	40.1	53.2	58.8			
stem-seg	40.3	53.7	43.0	74.9	85.7	17.7	5.7	21.6	68.2			
seg-S	54.3	66.0	68.8	75.6	83.2	82.5	33.0	48.7	55.0			
seg-S-mod	57.9	59.8	51.6	86.8	91.5	50.4	29.1	28.0	52.7			
seg-only	37.5	50.6	46.4	70.3	79.7	23.1	4.7	21.4	69.7			
Ancona												
stem-seg-S	**52.9**	**61.9**	**70.0**	84.5	88.5	71.2	12.5	26.1	70.3	61.7	71.2	68.5
stem-seg	45.4	55.3	66.4	83.9	90.2	61.0	9.3	21.5	70.2	43.0	54.1	67.9
seg-S	47.7	57.5	64.4	86.3	89.1	53.4	5.5	21.3	66.8	51.3	62.2	72.9
seg-S-mod	52.0	60.3	68.8	85.5	88.8	60.3	9.4	21.7	77.4	61.2	70.3	68.8
seg-only	46.6	57.3	62.0	86.4	89.4	52.4	3.9	18.1	73.1	49.5	64.3	60.4
Eschikon												
stem-seg-S	40.1	49.7	44.8	86.7	92.2	49.0	32.2	47.4	73.7	1.3	9.5	11.8
stem-seg	27.6	39.4	27.9	63.2	78.6	42.7	18.9	38.9	39.6	0.6	0.7	1.3
seg-S	**44.2**	**57.7**	**48.8**	80.1	86.2	67.9	29.3	52.9	58.9	23.2	33.9	19.7
seg-S-mod	36.1	47.1	42.3	85.3	91.0	48.7	21.8	44.7	77.2	1.1	5.6	1.0
seg-only	33.0	48.2	38.1	62.1	75.4	39.1	21.5	45.9	63.7	15.3	23.3	11.6

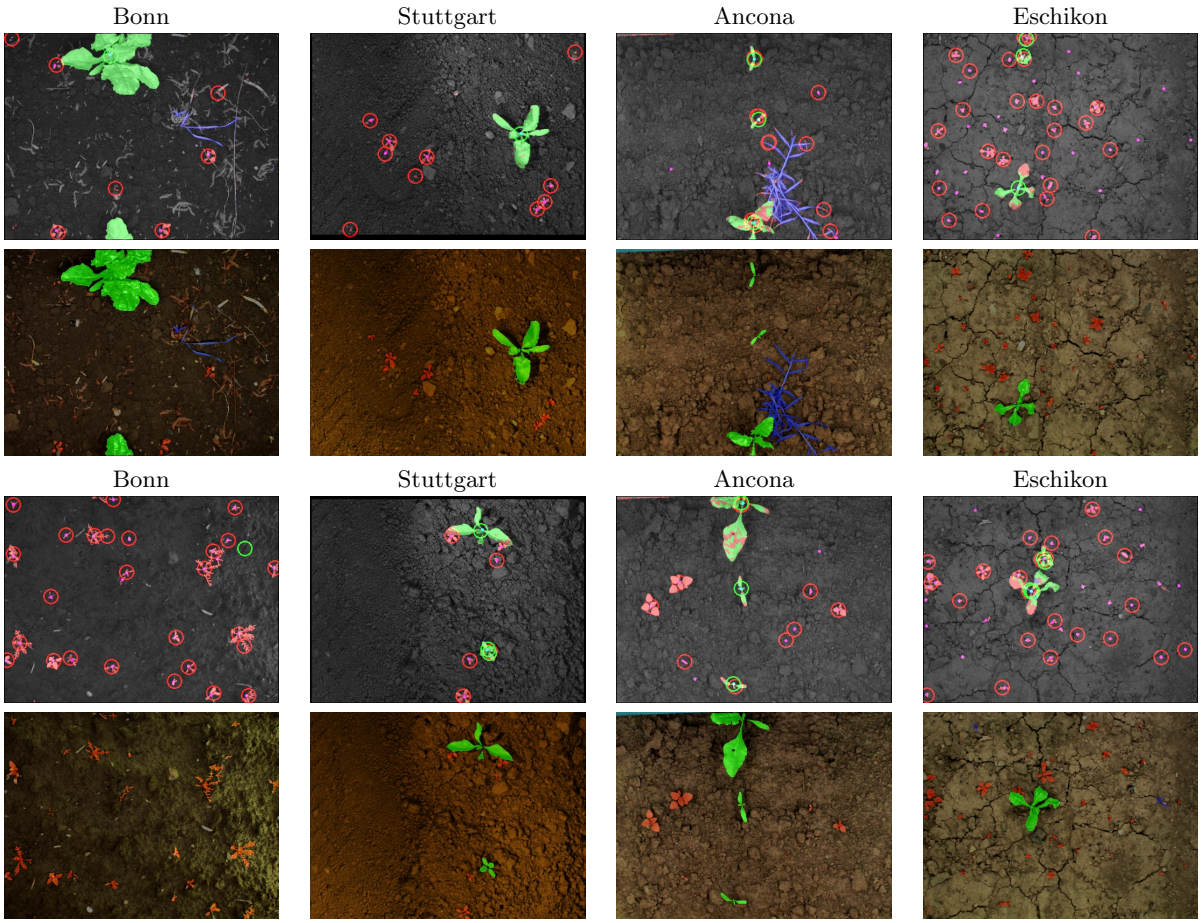

Figure 8: Qualitative results of our approach (*stem-seg-S*). We show two representative examples per dataset. The respective top rows represent an overlay of the NIR image with the prediction, where crops (green), dicot weeds (red), and grass weeds (blue) represent the semantic segmentation. The predicted stems are illustrated by red (dicot) and green (dicot) cycles, whereas the ground truth stems are illustrated by smaller filled circles. The bottom rows illustrate the ground truth of the semantic segmentation.

Note that the output used for the evaluation is the raw prediction by our approach and no further post-processing such as spatial smoothing is performed. By performing this post processing, we could improve the performance substantially due the aforementioned error source.

We also provide quantitative numbers on plant-level using the object-wise metric for the segmentation task. The same metric is also used in Lottes et al. (2018b) and can be used for comparison of the results. Tab. 4 summarizes the object-wise performance for our approach (*stem-seg-S*). On the Ancona and Stuttgart dataset our approach provides a good generalization performance across all classes.

Table 4: Object-wise performance of our approach *stem-seg-S*

Approach	Overall		Crop		Dicot-Weed		Grass-Weed	
	mP	mR	P	R	P	R	P	R
Validation	91.3	96.3	95.5	98.0	99.1	95.7	79.2	95.3
Stuttgart	87.2	84.8	79.8	90.2	94.6	79.4		
Italy	75.1	68.6	92.7	59.4	47.4	80.0	85.2	66.3
Eschikon	92.8	71.1	94.3	61.1	91.3	81.1		

5 Conclusion

In this paper, we presented a novel approach for joint stem detection and crop-weed segmentation using a FCN integrating sequential information. Our proposed architecture enables a sharing of feature computations in the encoder, while using two distinct task-specific decoder networks for stem detection and pixel-wise semantic segmentation of the input images. Furthermore, we encode the spatial arrangement of plants in a row by using 3D convolutions over an image sequence and share the resulting spatio-temporal feature maps with the ones produced by the task specific decoders. Our thorough experimental evaluation using real-world data demonstrates that our system is able to detect the stem positions of crops and dicot weeds as well as provides the semantic segmentation into crops, dicot weeds, grass weeds, and soil enabling robotic weeding system for precise and plant-specific treatments.

Acknowledgments

This work has partly been supported by the European Commission under the grant number H2020-ICT-644227-FLOURISH. We thank R. Pude and his team from the Campus Klein Altendorf for their great support as well as F. Langer, and J. Weyler, D. Gogoll, and J. Kirchdorf for labeling the datasets.

References

Badrinarayanan, V., Kendall, A., and Cipolla, R. (2017a). Segnet: A deep convolutional encoder-decoder architecture for image segmentation. *IEEE Trans. on Pattern Analalysis and Machine Intelligence (TPAMI)*.

Badrinarayanan, V., Kendall, A., and Cipolla, R. (2017b). SegNet: A Deep Convolutional Encoder-Decoder Architecture for Image Segmentation. *IEEE Trans. on Pattern Analalysis and Machine Intelligence (TPAMI)*, 39(12):2481–2495.

Behley, J., Steinhage, V., and Cremers, A. (2013). Laser-based Segment Classification Using a Mixture of Bag-of-Words. In *Proc. of the IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*.

Bogoslavskyi, I., Vysotska, O., Serafin, J., Grisetti, G., and Stachniss, C. (2013). Efficient Traversability Analysis

for Mobile Robots using the Kinect Sensor. In *Proc. of the Europ. Conf. on Mobile Robotics (ECMR)*, Barcelona, Spain.

Boser, B., Guyon, I., and Vapnik, V. (1992). A training algorithm for optimal margin classifiers. In *Proc. of the workshop on computational learning theory (COLT)*.

Breiman, L. (2001). Random forests. *Machine Learning*, 45:5–32.

Chebrolu, N., Lottes, P., Schaefer, A., Winterhalter, W., Burgard, W., and Stachniss, C. (2017). Agricultural robot dataset for plant classification, localization and mapping on sugar beet fields. *Intl. Journal of Robotics Research (IJRR)*.

Cicco, M., Potena, C., Grisetti, G., and Pretto, A. (2017). Automatic Model Based Dataset Generation for Fast and Accurate Crop and Weeds Detection. In *Proc. of the IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*.

Dumoulin, V. and Visin, F. (2018). A guide to convolution arithmetic for deep learning. *arXiv preprint*, abs/1603.07285.

Everingham, M., Van Gool, L., Williams, C., Winn, J., and Zisserman, A. (2010). The Pascal Visual Object Classes (VOC) Challenge. *Intl. Journal of Computer Vision (IJCV)*, 88(2):303–338.

Freund, Y. and Schapire, R. (1997). A Decision-Theoretic Generalization of On-Line Learning and an Application to Boosting. volume 55, pages 119–139.

Haug, S., Biber, P., Michaels, A., and Ostermann, J. (2014a). Plant stem detection and position estimation using machine vision. In *Workshop Proc. of Conf. on Intelligent Autonomous Systems (IAS)*, pages 483–490.

Haug, S., Michaels, A., Biber, P., and Ostermann, J. (2014b). Plant Classification System for Crop / Weed Discrimination without Segmentation. In *IEEE Winter Conf. on Appl. of Computer Vision (WACV)*.

He, K., Gkioxari, G., Dollar, P., and Girshick, R. (2017). Mask R-CNN. In *Proc. of the IEEE Intl. Conf. on Computer Vision (ICCV)*.

He, K., Zhang, X., Ren, S., and Sun, J. (2015). Delving Deep into Rectifiers: Surpassing Human-Level Performance on ImageNet Classification. In *Proc. of the IEEE Intl. Conf. on Computer Vision (ICCV)*.

Huang, G., Liu, Z., Maaten, L., and Weinberger, K. Q. (2017). Densely Connected Convolutional Networks. In *Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*.

Jégou, S., Drozdzal, M., Vázquez, D., Romero, A., and Bengio, Y. (2017). The One Hundred Layers Tiramisu: Fully Convolutional DenseNets for Semantic Segmentation. *arXiv preprint*, abs/1611.09326.

Kiani, S. and Jafari, A. (2012). Crop detection and positioning in the field using discriminant analysis and neural networks based on shape features. *Journal of Agricultural Science and Technology*, 14:755–765.

Kraemer, F., Schaefer, A., Eitel, A., Vertens, J., and Burgard, W. (2017). From Plants to Landmarks: Time-invariant Plant Localization that uses Deep Pose Regression in Agricultural Fields. In *IROS Workshop on Agri-Food Robotics*.

Krizhevsky, A., Sutskever, I., and Hinton, G. (2012). ImageNet Classification with Deep Convolutional Neural Networks. In *Proc. of the Advances in Neural Information Processing Systems (NIPS)*.

Kümmerle, R., Ruhnke, M., Steder, B., Stachniss, C., and Burgard, W. (2013). A Navigation System for Robots Operating in Crowded Urban Environments. In *Proc. of the IEEE Intl. Conf. on Robotics & Automation (ICRA)*, Karlsruhe, Germany.

Leibe, B., Leonardis, A., and Schiele, B. (2008). Robust object detection with interleaved categorization and segmentation. *Intl. Journal of Computer Vision (IJCV)*.

Leibe, B., Seemann, E., and Schiele, B. (2005). Pedestrian detection in crowded scenes. In *Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*.

Lin, M., Chen, Q., and Yan, S. (2014). Network In Network. In *Proc. of the International Conference on Learning Representations (ICLR)*.

Long, J., Shelhamer, E., and Darrell, T. (2015). Fully Convolutional Networks for Semantic Segmentation. In *Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*.

Lottes, P., Behley, J., Chebrolu, N., Milioto, A., and Stachniss, C. (2018a). Joint Stem Detection and Crop-Weed Classification for Plant-specific Treatment in Precision Farming. In *Proc. of the IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*.

Lottes, P., Behley, J., Milioto, A., and Stachniss, C. (2018b). Fully convolutional networks with sequential information for robust crop and weed detection in precision farming. *IEEE Robotics and Automation Letters (RA-L)*, 3:3097–3104.

Lottes, P., Höferlin, M., Sander, S., and Stachniss, C. (2017a). Effective Vision-based Classification for Separating Sugar Beets and Weeds for Precision Farming. *Journal of Field Robotics (JFR)*, 34:1160–1178.

Lottes, P., Khanna, R., Pfeifer, J., Siegwart, R., and Stachniss, C. (2017b). UAV-Based Crop and Weed Classification for Smart Farming. In *Proc. of the IEEE Intl. Conf. on Robotics & Automation (ICRA)*.

Lottes, P. and Stachniss, C. (2017). Semi-supervised online visual crop and weed classification in precision farming exploiting plant arrangement. In *Proc. of the IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*.

McCool, C., Perez, T., and Upcroft, B. (2017). Mixtures of Lightweight Deep Convolutional Neural Networks: Applied to Agricultural Robotics. *IEEE Robotics and Automation Letters (RA-L)*.

Midtiby, H., Giselsson, T., and Joergensen, R. (2012). Estimating the plant stem emerging points (pseps) of sugar beets at early growth stages. *Biosystems Engineering*, 111(1):83 – 90.

Milioto, A., Lottes, P., and Stachniss, C. (2017). Real-time Blob-wise Sugar Beets vs Weeds Classification for Monitoring Fields using Convolutional Neural Networks. In *Proc. of the Intl. Conf. on Unmanned Aerial Vehicles in Geomatics*.

Milioto, A., Lottes, P., and Stachniss, C. (2018). Real-time Semantic Segmentation of Crop and Weed for Precision Agriculture Robots Leveraging Background Knowledge in CNNs. In *Proc. of the IEEE Intl. Conf. on Robotics & Automation (ICRA)*.

Milioto, A. and Stachniss, C. (2018). Bonnet: An Open-Source Training and Deployment Framework for Semantic Segmentation in Robotics using CNNs. *Worshop on Perception, Inference, and Learning for Joint Semantic, Geometric, and Physical Understanding, IEEE Int. Conf. on Robotics & Automation (ICRA)*.

Mortensen, A. K., Dyrmann, M., Karstoft, H., Jörgensen, R. N., and Gislum, R. (2016). Semantic Segmentation of Mixed Crops using Deep Convolutional Neural Network. In *Proc. of the International Conf. of Agricultural Engineering (CIGR)*.

Papageorgiou, C., Oren, M., and Poggio, T. (1988). A general framework for object detection. In *Proc. of the IEEE Intl. Conf. on Computer Vision (ICCV)*.

Paszke, A., Chaurasia, A., Kim, S., and Culurciello, E. (2016). ENet: Deep Neural Network Architecture for Real-Time Semantic Segmentation. *arXiv preprint*, abs/1606.02147.

Potena, C., Nardi, D., and Pretto, A. (2016). Fast and accurate crop and weed identification with summarized train sets for precision agriculture. In *Proc. of Int. Conf. on Intelligent Autonomous Systems (IAS)*.

Rahman, M. A. and Wang, Y. (2016). Optimizing Intersection-Over-Union in Deep Neural Networks for Image Segmentation. In *Int. Symp. on Visual Computing*.

Ronneberger, O., P.Fischer, and Brox, T. (2015). U-Net: Convolutional Networks for Biomedical Image Segmentation. In *Medical Image Computing and Computer-Assisted Intervention*, volume 9351 of *LNCS*, pages 234–241. Springer.

Rumelhart, D., Hinton, G., and Williams, R. (1986). Learning representations by back-propagating errors. *Nature*, 323:533–536.

Slaughter, D., Giles, D., and Downey, D. (2008). Autonomous robotic weed control systems: A review. *Computers and Electronics in Agriculture*, 61(1):63 – 78.

Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., and Salakhutdinov, R. (2014). Dropout: A Simple Way to Prevent Neural Networks from Overfitting. *Journal of Machine Learning Research*, 15:1929–1958.

Stachniss, C., Martínez-Mozos, O., Rottmann, A., and Burgard, W. (2005). Semantic Labeling of Places. In *Proc. of the Intl. Symposium on Robotic Research (ISRR)*, San Francisco, CA, USA.

Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., and Wojna, Z. (2016). Rethinking the Inception Architecture for Computer Vision. In *Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, pages 2818–2826.

Viola, P. and Jones, M. (2001). Robust real-time object detection.

Wurm, K., Kretzschmar, H., Kümmerle, R., Stachniss, C., and Burgard, W. (2013). Identifying Vegetation from Laser Data in Structured Outdoor Environments. *Journal on Robotics and Autonomous Systems (RAS)*.