# Of Plants and Pixels:
# Leveraging Biological Priors for Ecological Data Analysis When Generic Methods Fall Short

Cumulative Dissertation

submitted for the academic degree of

Doktoringenieur (Dr.-Ing.)

Faculty of Computer Science and Electrical Engineering
University of Rostock, Germany

Alexander Gillert

2023

https://doi.org/10.18453/rosdok_id00004597

**Reviewers:**

Prof. Dr.-Ing. Uwe Freiherr von Lukas
University of Rostock, Fraunhofer IGD

Prof. Dr.-Ing. Oliver Staadt
University of Rostock

Prof. Dr.-Ing. Reinhard Koch
Christian-Albrecht University of Kiel

**Year of submission:** 2023
**Year of oral defense:** 2023

# Acknowledgements

I want to express my gratitude to the Fraunhofer IGD for the support and resources and to my colleagues for the discussions, feedback and constructive criticism.

Fraunhofer IGD

Universität Rostock Traditio et Innovatio

A special thanks goes to my fellow PhD students at the University of Greifswald who worked with me in this research project for the productive collaborations and knowledge exchange.

DIG-IT!

UNIVERSITÄT GREIFSWALD
Wissen lockt. Seit 1456

This dissertation was completed as part of the research project "Dig-IT!" and financed by funds from the European Social Fund (ESF). This work is part of the qualification program "Promoting young researchers in an excellent research network", an excellence research program of the state of Mecklenburg-West Pomerania (ESF/14-BM-A55-0015/19).

EUROPÄISCHE UNION
Europäischer Sozialfonds

Europäische Fonds EFRE, ESF und ELER
in Mecklenburg-Vorpommern 2014-2020

# Abstract

As ecological research continues to expand across temporal and spatial scales, it generates immense amounts of data. The constantly growing volume and complexity of this data poses a major bottleneck in the scientific process.

The primary objective of this interdisciplinary PhD project was to accelerate several data analysis tasks in ecological research that are too time-consuming to perform manually. In a first step, the feasibility of achieving this goal through the use of existing, generic methods was evaluated. For some specific tasks, such as instance segmentation of tree rings or root turnover estimation, the performance of these generic methods failed to meet the desired level of accuracy because of unique requirements that are not adequately addressed by generic computer vision research.

This led to the development of new algorithms addressing these new challenges which constitute the main part of this thesis. Incorporating domain-specific prior knowledge of the underlying biological systems and processes has shown to be essential in guiding the development of these algorithms.

The presented algorithms are packaged as user-friendly analysis tools that do not require a high level of technical expertise. They have already proven their usability within several ecological studies and led to new follow-up projects.

# Zusammenfassung

Da sich die ökologische Forschung weiterhin über zeitliche und räumliche Skalen ausdehnt, erzeugt sie immense Datenmengen. Die ständig wachsende Menge und Komplexität dieser Daten stellt einen großen Engpass im wissenschaftlichen Prozess dar.

Das Hauptziel dieser interdisziplinären Arbeit war es, mehrere Datenanalyseaufgaben in der ökologischen Forschung zu beschleunigen, die zu zeitaufwändig sind, um manuell durchgeführt zu werden. In einem ersten Schritt wurde die Realisierbarkeit dieses Ziels durch den Einsatz bestehender, generischer Methoden evaluiert. Für einige spezifische Aufgaben, wie z. B. die Instanzsegmentierung von Baumringen oder die Schätzung des Wurzelumsatzes, erfüllte die Erkennungsleistung dieser generischen Methoden nicht das gewünschte Genauigkeitsniveau, aufgrund einzigartiger Anforderungen, die in der generischen Bildverarbeitungs-Forschung nicht angemessen adressiert werden.

Dies führte zur Entwicklung neuer Algorithmen zur Bewältigung dieser neuen Herausforderungen, die den Hauptteil dieser Arbeit ausmachen. Die Einbeziehung von domänenspezifischem Vorwissen über die zugrunde liegenden biologischen Systeme und Prozesse hat sich als wesentlich für die Entwicklung dieser Algorithmen erwiesen.

Die vorgestellten Algorithmen sind als benutzerfreundliche Analysewerkzeuge verpackt, die kein hohes Maß an technischem Fachwissen erfordern. Sie haben ihre Verwendbarkeit bereits in mehreren ökologischen Studien bewiesen und zu neuen Folgeprojekten geführt.

# Contents

# Part I

# Synopsis

# 1. Introduction

## 1.1 Motivation

The pace of global climate change has accelerated in recent decades, posing a far-reaching threat to ecosystem stability and biodiversity. Extreme weather events such as heat waves, droughts or floods are becoming more frequent and more intense as global temperatures continue to rise unabated [Seneviratne et al., 2012]. In this period of rapid change, ecologists are increasingly being tasked with answering significant, pressing questions that have impact on the entire world [McCrea et al., 2023].

Ecology is an interdisciplinary academic field of research that focuses on how living organisms interact with each other and their physical and biological environment. Ecological research involves the collection and analysis of data which can range from observations at a microscopic level to large-scale global surveys. Regardless of the scope, ecological datasets are constantly increasing in their volume and complexity, enabling unprecedented scientific insights but also posing practical challenges for ecological inference. With the recent innovations and increasing availability in observation technology, such as digital cameras, sound recorders or remote sensing with satellites or drones, the bottleneck in the scientific research process has shifted away from the acquisition of these datasets towards gaining meaningful knowledge from them. Manually performing this task can be time consuming, tiresome and prone to errors [Christin et al., 2019]. Researchers require new data analysis methods to meet the demands of modern ecological research and Artificial Intelligence (AI) is one of the most promising and important technologies for this, as AI has opened up new possibilities of solving complex problems that traditional computing has not been able to handle.

AI is a branch of computer science that aims to create intelligent ma-

chines or tools that are capable of performing tasks that normally require human intelligence. Machine Learning (ML) is a subfield of AI that focuses on developing systems that are capable of learning from data or making decisions without being explicitly programmed. Deep Learning (DL) is a subset of ML that uses artificial neural networks motivated by the learning process of the human brain [LeCun et al., 2015]. 2012 is commonly seen as the start of the DL era with the introduction of the AlexNet [Krizhevsky et al., 2012] architecture which outperformed previous paradigms on the ImageNet [Deng et al., 2009] benchmark. Since then, DL algorithms have been used to solve complex problems, such as medical image analysis [Litjens et al., 2017], natural language processing [Young et al., 2017] and autonomous driving [Grigorescu et al., 2019]. DL has surpassed human-level performance on some of these tasks [He et al., 2015].

Unfortunately, the field of ecology has received comparatively little attention from the DL community so far, although close collaborations between ecologists and computer scientists could provide innovative solutions to globally relevant issues and lead to progress that benefits all involved parties [McCrea et al., 2023].

The benefits for ecology are obvious: AI-based automated analysis methods provide fast processing speed and high accuracy of the results. Besides, there are other aspects that are of utmost importance. Analyzing complex data manually by hand can lead to inconsistent results depending on the experience level of the person [Peters et al., 2023]. AI-based analysis methods on the other hand can be used to ensure that results are reproducible and more coherent. All these advantages enable scaling scientific studies to sizes that could not have been feasible previously.

Computer science can also benefit from this collaboration, because as this thesis shows, simply applying existing algorithms from other domains is often insufficient. Computer vision research can benefit from unique problems in ecology that have not been addressed so far.

Despite the potential benefits of collaboration between ecological and computer science researchers, there are several obstacles that can impede progress. As identified in [Goodwin et al., 2022] these mainly consist of three barriers: Firstly, knowledge barriers and field-specific jargon can hinder communication and understanding between ecologists and computer scientists, making it difficult to transfer necessary information and ideas between the two groups. Secondly, ecologists who lack familiarity with

AI may miss out on potential solutions to their research problems. They may need to know about the possibilities and limitations of AI for their specific task, how to prepare and annotate datasets, and what information to provide to the computer scientist to identify the best AI method for the job. Conversely, computer scientists may struggle to understand the underlying ecological question, the data and its inherent variability or noise, its classification, and the required level of accuracy before offering advice on potential AI solutions. Both parties must be willing to invest significant effort in the interdisciplinary partnership, to overcome these barriers and achieve a common understanding.

This interdisciplinary PhD project has aimed at bridging exactly this gap between ecology and computer science. Its focus was on application of existing methods as well as development of new AI-based computer vision algorithms for analysis of ecological image data from various acquisition sources such as microscopes, camera traps and minirhizotrons to assist ecologists addressing scientific questions about ecosystem functioning and future reactions to changes. The methods presented here are mostly specialized to solve a specific problem as the variety of ecological tasks is too large to be covered by a single generalized method. The analysis tools containing these methods are already in active use by ecologists today.

## 1.2 Thesis Outline

This doctoral thesis is written in form of a cumulative dissertation. Chapter 2 covers the ecological background information about the topics of this project. An overview of the research methodology is provided in Chapter 3 including the main research questions from a computer science point of view in 3.1 and an extensive literature review which includes newer publications and without the page limitations of the conference papers in section 3.2. Chapter 4 summarizes the outcomes of this research, including an overview of the main publications in 4.1 and a description of the developed analysis tools in 4.2. This part is completed with an outline of the already achieved impact and perspectives for future research in Chapter 5.

In Part II, the main part of this thesis, the following works are included as published in their original, unmodified versions:

- Page 53: ”Iterative Next Boundary Detection for Instance Segmentation of Tree Rings in Microscopy Images of Shrub Cross Sections” [Gillert et al., 2023b]
- Page 67: ”Tracking Growth and Decay of Plant Roots in Minirhizotron Images” [Gillert et al., 2023a]
- Page 83: ”Identification and Measurement of Individual Roots in Minirhizotron Images of Dense Root Systems” [Gillert et al., 2021]
- Page 93: ”Towards Combined Open Set Recognition and Out-of-Distribution Detection for Fine-grained Classification” [Gillert and von Lukas, 2021]

Furthermore, the following works that were published in collaboration with ecology researchers from the University of Greifswald were part of this PhD research, but their contributions to the field of computer science are limited and therefore published in ecology journals. They are not fully included and only briefly covered in section 4.1.

- ”Mask, Train, Repeat! Artificial Intelligence for Quantitative Wood Anatomy” [Resente et al., 2021]
- ”Rewetting prolongs root growing season in minerotrophic peatlands and mitigates negative drought effects” [Schwieger et al., 2022]
- ”As good as human experts in detecting plant roots in minirhizotron images but efficient and reproducible: the convolutional neural network “RootDetector”” [Peters et al., 2023]
- ”BatNet: a deep learning-based tool for automated bat species identification from camera trap images” [Krivek et al., 2023a]

# 2. Ecological Background

## 2.1 Wood Anatomy

During the course of the their lifetime plants react to various environmental events and store this information expressed in cell structures, tree ring widths or chemical composition. This is the basis of *dendroecology* which examines these growth patterns in relation to ecological drivers [Schweingruber et al., 1996]. It can be used to reconstruct past climate conditions and helps to understand future reactions of ecosystems to climate change [Charney et al., 2016].

Traditionally, the most important parameter for climate reconstruction has been tree ring width. This parameter is relatively easy to measure manually but only summarizes all possible growth processes during a year into a single value. It is thus of limited use for estimating plant reactions to ever more frequently occurring extreme events such as droughts, late frost [Diffenbaugh et al., 2017] or insect outbreaks [Wilmking et al., 2018]. *Wood anatomical* investigations within the rings can be more suitable, for one due to the higher temporal resolution, for the other because parameters like cell dimensions are more closely intertwined with physiological processes and more sensitive to climate fluctuations than simple ring widths. Additional parameters such as cell wall thickness, vessel density, ray abundance and others, contain further valuable ecological information because they often react to different environmental signals [Fonti et al., 2010].

Dendroecology and quantitative wood anatomy are not limited to the study of trees. Shrubs are another important source of ecological information and of special interest for environments with harsh climatic conditions such as the Arctic regions [Power et al., 2022]. They dominate this landscape and are often the only woody plant that can provide a record of

environmental events in these ecosystems. In particular, dendroecology of shrubs has been employed in reconstructing climate [Rayback et al., 2012], measuring responses to changing climatic conditions on a landscape level [Forbes et al., 2010] or assessing human impacts in tundra ecosystems [Rixen et al., 2004].

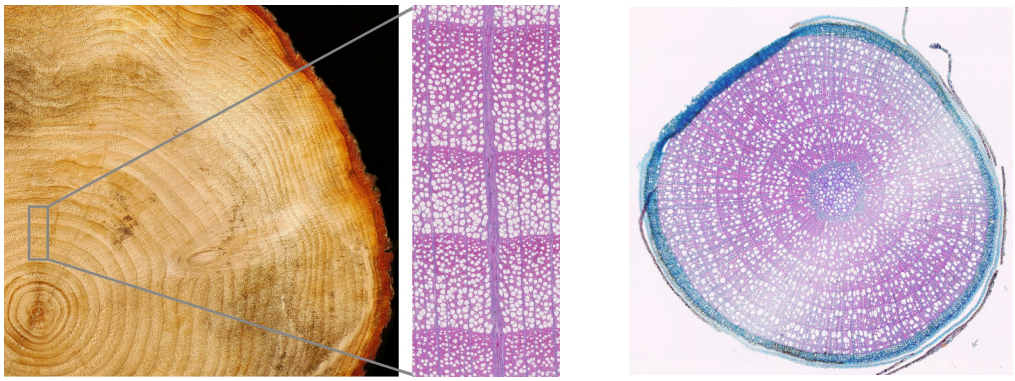


**Figure 2.1:** Illustration of tree rings from a tree (left) and a shrub branch (right) as seen under a microscope. (Images provided by the University of Greifswald (left) and CREAF (right))

Wood anatomical analyses are usually performed on thin sections of tree cores or cross sections of branches or roots of 10-20 $\mu m$ thickness, cut with a sledge or rotary microtome and stained with a coloring agent to increase contrast of the cell walls and ring boundaries. They are then observed under a microscope with a high image resolution to capture cellular information with examples illustrated in Figure 2.1. This high resolution, the abundance of cells in each image and the large number of different properties that can be extracted from them, makes manual data analysis extremely time-consuming, which in turn makes it difficult to upscale ecological studies.

## 2.2 Root Ecology

Plants consume $CO_2$ from the atmosphere and store it via photosynthesis as carbon in their biomass. When plants decompose, the carbon in the aboveground part of their biomass is released back into the air. Belowground on the other hand, i.e. in the root system biomass, carbon is re-

tained for a longer period in the ground, especially in wetland landscapes [Bridgham et al., 2006]. With the ground containing more than twice as much carbon as is contained in the atmosphere, this makes roots a crucial component in the effort to offset human-made greenhouse gas emissions.

The primary storage pathway of carbon into the ground (carbon sequestration) is happening via *root turnover*, the periodic process or root growth and decomposition [Matamala et al., 2003, Norby et al., 2004]. This process is directly connected to the seasonal activity of the plants (*phenology*), such as the start and duration of the growing season. Whereas aboveground phenology is relatively easy to monitor (e.g. via remote sensing) and intensively researched, knowledge about the activity of roots is highly incomplete [Laliberté, 2017]. Yet, further investigations are needed in this area, as activity patterns below- and aboveground can differ significantly [Blume-Werry et al., 2016] prohibiting inference based solely on aboveground observations. With changing environmental and climatic conditions which affect both the phenology and lifespan of roots this could have implications on ecosystem functioning and the carbon cycle at a global level. [Eissenstat et al., 2000].

Plant root research is traditionally very labor-intensive since roots are hidden belowground. The most often used techniques such as digging and washing out only capture around 60% of the biomass [Robinson, 2004] losing the important fine roots (<2mm), which not only are responsible for the water and nutrient uptake and thus for the growth of the plant but also constitute the major part of carbon sequestration [Norby et al., 2004]. Moreover, these techniques capture only a single moment in time and do not allow for acquisition of time series data, since the plant is destroyed in the process. Time series are needed to estimate the turnover and lifespans of roots, an important parameter for estimation of the amount of carbon sequestration especially for the ephemeral fine roots which live only for days or weeks.

Minirhizotrons (Figure 2.2) are transparent tubes that are buried into the ground, so that roots can grow around them. With a camera or a scanner, roots can be monitored over longer periods from inside the tube, without disturbance. They have been used to understand root functioning and to quantify fine-root dynamics in forests and wetlands [Hendrick and Pregitzer, 1996, Iversen et al., 2011]. Analyzing these time series manually is not only time-consuming but can also lead to incorrect results, especially from beginners. AI-based methods show more consis-

**Figure 2.2:** Minirhizotrons in a mesocosm experiment (left) and a crop of the resulting minirhizotron image data (right).
(Images provided by the University of Greifswald, slight modifications added)

tent and much faster performance [Peters et al., 2023].

Ecological studies with minirhizotrons are usually conducted as mesocosm or field experiments. Mesocosms are an intermediate form between fully controlled laboratory experiments and the very realistic in-situ field tests. They provide a semi-controlled environment, ensuring some factors for reproducibility such as soil type and plant species while being exposed to realistic weather conditions outdoors. For field experiments, minirhizotrons are installed in target environments such as forests, bogs, grasslands and others to monitor real-world ecological processes over longer periods, in this case root dynamics.

As field experiments are more realistic, they are more valuable from an ecological perspective but also more difficult to analyze. The image data often contains ambiguous situations, such as whether or not a root is alive or dead, close visual resemblance of roots and mycorrhiza (fungi), posing challenges to both human experts and machines.

The average manual processing time for a minirhizotron image amounts to around 3 hours. Given that a high acquisition frequency on a weekly or even daily basis is needed to capture fine-root dynamics, this puts limits on the scale of experiments and the insights that can be gained from them.

## 2.3 Bat Conservation

Bats are present in nearly every terrestrial environment across the globe, providing crucial ecosystem services such a insect control, pollination and seed dispersal [Kunz et al., 2011]. However, they also face a variety of threats, from the destruction and degradation of their habitats to the spread of infectious diseases and the impacts of climate change, resulting in worldwide declining populations [Frick et al., 2020]. Accurate estimates of their population trends are required to assess the effectiveness of conservation strategies, assist policy makers in making wildlife management decisions and thus ensure the aforementioned ecosystem services are sustained.

**Figure 2.3:** Camera trap setup triggered by infrared light barriers (left) and a crop of a typical camera trap image, recording a bat flying through the opening. (Images provided by the University of Greifswald)

Traditional methods of collecting bat population data include field surveys, which involve researchers visiting bat winter hibernacula (e.g. caves or old buildings) to manually count the bats. These methods are not only

time-consuming and costly but can also be invasive and fail to accurately estimate the population data, as they do not take into account individuals that may not be visible (e.g. hidden in small crevices), thus underestimating their true numbers [Krivek et al., 2023b]. More recently, newer technologies such as acoustic monitoring with microphones that detect echolocation calls have been used [Thomas and Davison, 2022]. They provide more accurate estimates but lack the ability to provide a more fine-grained identification by bat species.

Camera traps allow for both, accurate population estimates and the identification of different species upon visual inspection. In contrast to often-used motion-triggered camera traps used to monitor terrestrial mammals, cameras for bat monitoring can be triggered by bats passing through an infrared light barrier at the entrance of their winter hibernacula (Figure 2.3), allowing for more comprehensive population size estimates. Additionally considering their low impact on bat behavior, this makes them an ideal monitoring method [Krivek et al., 2022].

A single camera trap setup can generate around 30,000 images per year for an average hibernaculum hosting around 600 bats [Krivek et al., 2023a]. Taking into account that much larger sites exist, an average processing time of half a minute per image and that species identification requires highly trained experts, this large amount of data makes manual bat counting and species identification nearly infeasible for large-scale monitoring projects.

## 2.4 Paleopalynology

Palynology is the study of pollen and spores, a term coined by [Hyde and Williams, 1944]. It finds various applications in diverse domains such as pollen forecasts in health care [Bastl et al., 2017], pollination ecology [Bertrand et al., 2019], and honey quality monitoring [von der Ohe et al., 2004].

The field of *paleo*palynology is a branch of palynology that uses fossilized pollen to draw insights about past climates, several thousands or even millions years ago [Birks and Berglund, 2018]. Pollen grains are released by flowering plants and transported by the wind and remain comparatively well-preserved in sediments. Therefore, they can provide a reliable proxy of a region's vegetation history. By examining changes in the

composition of pollen or spore species at different sediment depths, palynologists can infer which plants were present at the corresponding time point, drawing conclusions about past climates, environments, ecosystems and human land use of a local area. Lennart von Post was a pioneering figure in this field in the early 20th century, presenting the first *pollen diagrams*, standardized charts that show a comprehensive vegetation history of a local area [Birks and Berglund, 2018].

Samples are usually obtained by drilling a sediment core from lake sediments or peatlands. The individual sedimentary layers are then observed under a microscope, where the pollen species are classified and their relative abundance is quantified (Illustrated in Figure 2.4).

Analyzing data manually requires skilled experts and is a time-inefficient process due to the large number of grains per sample, bringing with it the need to compromise between the quantity of samples to analyze and the number of analyzed pollen grains per sample. This puts limitations on the study size and can lead to an increase in the level of uncertainty in quantitative estimates [Bertrand et al., 2019, Olsson et al., 2021]. Given that several thousands of microscopy images per sediment core depth sample can be generated, creating a full pollen diagram by hand at the present is enough work for a PhD thesis [Stebich, 1999].

**Figure 2.4:** Drilling of a sediment core (left), the resulting core (center) and microscopy image data (right). (Images provided by the University of Greifswald)

Depending on several factors such as the age of the sediment layer, the pollen grains may either be well preserved in their original form or may be degraded by biological decomposition or chemical and physical processes, such as dehydration or oxidation. These processes can signif-

icantly change the visual appearance and make species identification extremely challenging, for both, human experts as well as AI systems. In fact, many objects cannot be identified at all.

# 3. Research Design

## 3.1 Research Questions

While the overall objective of this research project is to accelerate analyses of ecological image data, this can be broken down into several steps. The steps are formulated here as a set of research questions from a computer science perspective, guiding the reading through this thesis and serving as a documentation of the conducted research process. These research questions can also be utilized as a framework for future endeavors in other ecological research areas or even in other scientific fields.

First, one has to consider that the development of new Deep Learning based data analysis methods entails a considerable investment of resources, including time, energy (due to the potentially long and repeated use of GPUs for neural network training), and therefore funds, all of which is highly undesirable. To mitigate costs and efforts one has to make the most of available tools and methods, invented by other researchers that have been successfully employed in other domains. Consequently, the first research question aims to identify which ecological problems can be effectively addressed using readily available computer vision methods to avoid reinventing the wheel. By addressing this question, specialized methods need to be developed only when needed.

**RQ1** *Which ecological data analysis problems can be sufficiently well addressed using readily available computer vision methods?*

RQ1 is answered via a comprehensive literature review which is presented in section 3.2 as well as own experiments and evaluations from an ecological perspective which were published in ecology journals [Resente et al., 2021, Peters et al., 2023, Krivek et al., 2023a] for best outreach to the ecological scientific community.

As soon as this has been clarified, the corresponding complementary question arises. Generic computer vision algorithms were not designed to handle the complex structures and irregular patterns that ecological data frequently exhibit. This type of data is usually acquired not from controlled laboratory experiments but from natural environments which makes it often highly variable and subject to noise, negatively affecting the accuracy and reliability of the results. The aim of the second research question is to identify the limitations and deficiencies of generic methods, and investigate the underlying reasons for these limitations. The insights gained from this analysis can guide the development of targeted solutions that effectively address these problems.

**RQ2** *Which problems are unique to the field of ecology, and not adequately served by methods from other domains and why is this the case?*

The answers to this question are of course not exhaustive but rather focus on the specific research interests of the ecology working groups that have collaborated in this project.

Having identified those problems for which no adequate solution exists yet, one can take a closer look at the data in question to draw inspiration from biology to tackle these problems. Whereas the recent trends in computer vision research strive to reduce the use of priors (e.g. via Vision Transformers [Dosovitskiy et al., 2020]) to maximize generalizability, this comes at the cost of neglecting specialized problems that do not benefit from this.

To address these problems, it can be beneficial to impose constraints on algorithms by incorporating priors, i.e. leveraging knowledge about biological processes and systems to guide the design of computer vision algorithms. Priors can help to reduce the search space for the algorithm making it more accurate, lead to better generalization on out-of-distribution data and enhance interpretability of the results. Answering this third research question necessitates interdisciplinary discourse and knowledge exchange.

**RQ3** *In what way can biological priors be incorporated into computer vision algorithms to automate yet unsolved data analysis tasks?*

Finally, the practical implementation raises a number of concerns from the end user perspective that need to be addressed. Deep Learning algo-

rithms are known to be very data hungry and usually require user guidance in form of manually created annotations. This becomes particularly problematic and time-consuming with ecological image data which is often acquired at high resolutions. Therefore, the last research question is focused on finding effective ways to mitigate this issue. Specifically, answers to this research question can draw inspiration and apply techniques from the unsupervised or self-supervised machine learning line of research.

**RQ4** *How can the required amount of data and annotation effort be reduced effectively?*

Questions RQ2-4 have been addressed in peer-reviewed computer science publications that are included in Part II of this thesis. A brief overview of these publications, their methods and the main findings are presented in section 4.1.

## 3.2 Literature Review

There have been numerous applications of AI to ecological issues, especially since the start of the deep learning era in 2012 [Christin et al., 2019]. These focus mostly on subjects such as wild animal monitoring with camera traps [Nguyen et al., 2017, Willi et al., 2018, Norouzzadeh et al., 2017] or large-scale observation of human impact on ecosystems with satellite imagery [Brovelli et al., 2020].

The subsequent subsections provide a detailed literature review of the current state of applied AI and computer vision technology with a specific focus on publications directly related to the narrower topics of this PhD project. It is aimed at answering research questions RQ1 and RQ2 by identifying solved problems and the remaining research gaps that still need to be addressed.

### 3.2.1 Generic Computer Vision

As RQ1 is about reusing previous methods for new ecological domains, an overview of relevant and well established generic computer vision methods is provided in this subsection.

U-Net [Ronneberger et al., 2015] is a convolutional neural network (CNN) architecture that was originally designed for biomedical image seg-

mentation tasks but since been successfully applied to a wide variety of application domains. It is composed of an encoder network, which performs a series of down-sampling operations aimed at converting the raw pixel-space into a more meaningful latent feature-space, connected to a decoder network, which restores the spatial resolution with series of up-sampling operations. The architecture also includes skip connections that connect the encoder and decoder networks to enable the propagation of spatial information across different scales and preserving the image resolution. U-Net and several of its variations have become a popular choice for various segmentation tasks outside of the original biomedical domain, due to its simplicity, high degree of generalizability, pixel-level precision and comparatively small number of parameters.

Faster R-CNN [Ren et al., 2015] is a region-based object detection model that was introduced in 2015. It is composed of two main components: a Region Proposal Network (RPN) and a Fast R-CNN [Girshick, 2015] network. The RPN generates region proposals, which are areas in the image that potentially contain objects, and the Fast R-CNN network classifies each proposed region and refines the bounding box coordinates of the object within the region. It is usually classified as a two-stage approach although the RPN and Fast-R-CNN share a common backbone. An often used backbone is the ResNet [He et al., 2016] with a feature pyramid network (FPN) [Lin et al., 2016] for detection of objects at multiple scales.

Mask R-CNN [He et al., 2017] is a natural extension of Faster R-CNN that adds an additional head to the network for predicting object masks in addition to bounding boxes, designed for the general task of instance segmentation. The mask branch predicts a binary mask for each object in the proposed regions, enabling pixel-level segmentation. This extra information that the segmentation mask provides, enables measurement of various object properties such as area or diameter. As with the previously mentioned architectures, Mask R-CNN is among the most widely used neural networks as it achieves robust segmentation performance across a variety of practical applications.

Although numerous enhanced architectures have been proposed since their inception, the above mentioned methods continue to be widely used in various computer vision applications, due to their strong detection performance, versatility, speed and code availability. In real-world applications these latest models typically offer only marginal benefits.

Various ecological domains have repurposed the above methods for their specific use cases, albeit sometimes with slight adaptations and modifications. Examples are listed in the following subsections. All of these methods are based on Deep Learning which has become the standard paradigm in computer vision. It offers many benefits such as generalizability and ease of use, automatically learning a mapping from the input to a desired output without requiring a high level of domain knowledge. However, although generalizable methods can cover a wide range of problems with sufficient performance, they often struggle when addressing highly specialized problems. Additionally, neural networks act as black boxes, posing challenges for the interpretability of the outputs.

### 3.2.2 Microscopic Wood Image Analysis

The most often used analysis tool for wood anatomical image analysis is ROXAS [von Arx and Dietz, 2005, von Arx and Carrer, 2014]. It is however based on traditional computer vision techniques rather than machine learning. As a consequence, it is sensitive to perturbations in the input data and requires domain knowledge and manual species-specific tuning of parameters. Machine learning approaches are more flexible and easier to adapt to new species or acquisition devices, without technical or domain knowledge by the end user.

Application of Deep Learning in this field so far has focused on detecting cells. For instance, [García-Pedrero et al., 2018, García-Pedrero et al., 2019] have interpreted this as a semantic segmentation task and have used a U-Net architecture for binary classification of each pixel as cell or not cell. They however only evaluate with pixel-level metrics (e.g. how many pixels were correctly detected), neglecting instance statistics, (e.g. how many cells were correctly detected).

Publications on the detection of tree rings have mostly dealt with photography images or scans of wood rather than microscopy images [Fabijańska et al., 2017, Fabijańska and Danek, 2018]. This restricts the possible downstream processing use cases, e.g. the resolution is too low and does not capture cellular information. These papers only use partial images or crops rather than full cross sections and employ heuristics to compute the chronological order of tree rings. Detecting tree rings in cross sections on the other hand is more challenging because the objects are concentric, i.e. have (almost) the same center. Additional challenges

include high precision requirements and tree-ring specific issues such as wedging rings. No publication, neither from the computer vision community, nor in the dendrochronology community has addressed this specific problem prior to the work that is included in Chapter 6 of this thesis [Gillert et al., 2023b]. The only few marginally similar works have significant differences: [Martinez-Garcia et al., 2022] use X-rays to detect concentric tree rings in tree stems, a method not applicable for such fine objects such as branches of shrubs and not revealing intra-annual information. [Cerda et al., 2007] use traditional computer vision techniques such as Canny edge detection to detect tree ring boundaries in tree stems which fails on microscopy images. Moreover, their method cannot deal with wedging rings and shows poor performance on rings that are close to the center.

### 3.2.3 Minirhizotron Image Analysis

The majority of published work on automated minirhizotron data analysis puts focus on agricultural rather than ecological applications. PRMI [Xu et al., 2021] for example, the currently largest publicly available dataset of annotated minirhizotron image data, contains mainly recordings of crop plants such as peanuts, cotton or sunflowers.

Differences between agricultural and ecological use cases can be substantial: For one, the analyzed plant species can have different root systems for which generic algorithms exhibit performance issues and adaptations should be made for optimal results. Next, the desired plant traits that need to be estimated are also not the same: root turnover is an important parameter for ecological studies as it corresponds to the amount of carbon deposited into the ground. For agriculture, this parameter is of much smaller interest, because most crop plants are annuals, i.e. they live only for one season and die off after harvest. Thus no publication has addressed time series analysis of minirhizotron data so far. All of these aspects can change the requirements for the automatic analysis method and methods developed for agricultural use cases may not be optimal for ecological applications.

In most publications, the basic procedure for automatic analysis of minirhizotron images is to classify each pixel as either ”root” or ”not root”, a task commonly known as semantic segmentation. The specific network architectures used are either SegNet

[Badrinarayanan et al., 2015] as used with slight modifications by SegRoot [Wang et al., 2019] or U-Net [Ronneberger et al., 2015] as used by [Smith et al., 2019, Narisetti et al., 2021]. Further developments usually aim at improving the detection performance such as with data augmentations via grid deformations in [Smith et al., 2019].

In [Yu et al., 2019, Yu et al., 2020] the authors use multiple instance learning (MIL) to reduce the amount of annotation effort needed to train root detection networks. MIL enables weakly supervised training from image-level labels only containing information whether an image contains roots or not. The drawback here is that negative images (i.e. those not containing any roots) are needed which are usually not acquired in ecological studies.

One of the few minirhizotron analysis methods to take the temporal component into account is rhizoTrak [Moeller et al., 2019], a Fiji plugin for annotation of time-series. However, this tool offers only manual annotation for tracking roots and provides no capabilities for automated processing.

Of lesser use for ecology applications are works that employ other root observation methods or ex situ experimental setups. These include X-ray computed tomography systems such as in [Soltaninejad et al., 2019]. This method, although also non-destructive and in situ, has the crucial drawbacks that it only works well in specific types of soils and only coarse roots can be captured well, whereas the important fine roots remain mostly undetected [Phalempin et al., 2020].

ChronoRoot [Gaggion et al., 2020] uses time series to analyze growth patterns of roots. They use images from ex situ experiments with plants growing in agarized medium and acquired with high temporal resolution. This method depends on these requirements and is thus not transferrable to the real-world experiments in forests or bogs that are of interest for ecology research.

A different direction is explored by RootNav 2.0 [Yasrab et al., 2019] which uses the A* root path finding algorithm [Hart et al., 1968] to reconstruct the root system architecture from the surface level to first and second order root tips. However here too, the authors use an ex situ plant phenotyping system in which all roots are easily recognizable. In in situ minirhizotron image data, roots often get obstructed by soil and it is rarely possible to follow a path longer than a few centimeters or to reconstruct a full root system.

### 3.2.4 Wildlife Monitoring with Camera Traps

Automatic monitoring of animals with camera traps has been subject to a number of publications but has focused mainly on terrestrial mammals in various habitats such as forests, deserts or savannas [Wilber et al., 2013, Swanson et al., 2015, Beery et al., 2021] or fish in rivers [Tödtmann et al., 2020] or coral reefs [Villon et al., 2018] among others. Most of these monitoring applications are focused on recognizing animals that are completely unrelated to each other, e.g. gazelles, lions, elephants. Classifying animals into finer grained species level is much more challenging and rarely subject in the published literature.

BatNet [Krivek et al., 2023a] is the first project to address automated bat monitoring with camera traps. The recognition is performed on a species level, which despite visual similarity of many bat species results in good performance.

Automated wildlife monitoring with camera traps often faces a common constraint related to transferability, where a model trained to identify species in one particular area may not perform as effectively for the same species in different geographic locations [Tabak et al., 2020]. Besides different location-dependent distributions of species, [Miao et al., 2019] identified the image background as the main reason for this issue. As the cameras are static, the background hardly changes and neural networks are prone to shortcut learning [Geirhos et al., 2020], i.e. associating certain background features with a certain type of animal. If not accounted for, this can result in a bias towards the dominant animals and to poor detection performance at newly set-up locations on which the neural network was not trained on.

### 3.2.5 Pollen Counting

There have been several publications on automated pollen counting in microscopy images with deep neural networks starting with [Khanzhina et al., 2018, de Geus et al., 2019, Menad et al., 2019]. The used images are mostly from clean samples from aerial pollen traps or reference image collections and thus contain none or minimal amounts of dirt or decomposed pollen [Olsson et al., 2021] and in many cases even manually cropped [Kaya et al., 2013, Sevillano and Aznarte, 2018, Sevillano et al., 2020], thus omitting the crucial pollen detection step and

resulting in minimal time-saving benefits over fully manual pollen analysis. For lake sediment samples, differentiating between pollen and other objects can be the most challenging task, with decomposed or broken pieces of pollen grains making this decision extremely hard, for both human experts and machines. An additional challenge is the long-tailed imbalanced distribution of classes, i.e. while some species are abundant, for the majority of species only a handful of grains are present in a sample. This leads to bias in the training process if not taken care of.

One of the few publications that address a more realistic setting with most of the issues mentioned above is [Punyasena et al., 2022]. The authors use a custom object detection method, based on a U-Net, with each pixel predicting whether it belongs to a pollen or not and additionally estimating a vector to the center of mass for each pollen grain. They achieve decent results, yet again as their application area is not paleopalynology their data is acquired from pollen traps rather than from lake sediments, i.e. it contains much less dirt and debris.

### 3.2.6 Fine-Grained Open Set Recognition

Neural networks are known to be prone to producing overconfident predictions even if the results are incorrect [Guo et al., 2017, Li and Hoiem, 2020]. Recognizing unknown classes with neural networks, i.e. ones that were not present in the training data is even harder. Even more so if the classes are on a fine-grained level, i.e. (sub-)species. At the same time this is a highly relevant task and occurs in many object counting applications such as pollen: species that are too rare to train on, still should be recognized as unknown or at least report a warning to the end user that the confidence of the result is low and special care is needed.

Prior to the publication that is included in Chapter 9 [Gillert and von Lukas, 2021], no previous work has dealt with recognizing unknown fine-grained categories. Only afterwards (or possibly concurrently) further methods have been developed. This includes [Jezequel et al., 2021] who use an ensemble of networks, each trained on slightly different tasks such as solving a jigsaw puzzle or estimating the rotation of an input image. Their outputs are then aggregated into a final anomaly score. In [Cheng and Vasconcelos, 2021], a regularization constraint during training is introduced to enforce Gaussian class-conditional distributions.

Similarly to [Gillert and von Lukas, 2021], in [Linderman et al., 2022] different degrees of unknown classes are considered. This work uses an inference mechanism that predicts at different levels of granularity, at each level comparing how similar the input is to the corresponding set of known classes.

The authors of [Sun et al., 2022] address this issue from the perspective of vision transformers. They argue that unknown fine-grained classes can be easier recognized by inspecting the features during their propagation within the network. To this end, they propose a spatial-temporal attention network, which combines the depth-wise sequence of spatial features with a LSTM [Hochreiter and Schmidhuber, 1997] module.

Finally, the publication [Zhang et al., 2023] shows that Outlier Exposure [Hendrycks et al., 2018], a common technique with coarse-grained classes in which additional training data is used, is insufficient for fine-grained classes. It introduces a variation of this technique, improving the performance.

# 4. Research Outcomes

## 4.1 Publications

An overview of the publications derived from this research project is presented in Table 4.1 along with the main contributions they made to the computer vision or ecology scientific communities.

The first three publications address the research question RQ1 within the context of wood anatomy, root ecology and bat monitoring. They mostly apply existing generic algorithms, such as U-Net [Ronneberger et al., 2015], Faster-R-CNN [Ren et al., 2015] and Mask-R-CNN [He et al., 2017] to new problems in ecology, albeit with some modifications that improve the performance for each specific problem. The contributions to the field of computer science are marginal and therefore published in ecology journals and not fully discussed or included in this thesis. The main contribution of these publications is the evaluation from an ecological perspective and introduction of AI-based data analysis methods to the corresponding sub-fields of ecology. The results obtained for each of the addressed problems were deemed satisfactory and in some cases reaching the level of human experts, thereby adequately answering RQ1.

The remaining four publications are the primary content of this thesis and are included as published, including supplementary material, in part II as individual chapters. All of these works deal with yet unexplored problems in computer vision that are mostly unique to the field of ecology thus addressing RQ2. RQ3 and RQ4 also addressed to different degrees as mentioned below. In all of these publications, the author of this dissertation has contributed to the definition of the problem from a computer science point of view, the main algorithm development, data annotation and evaluation.

| Publication | | |
|---|---|---|
| Ch. | Addressed Questions | Main Findings |
| **Mask, Train, Repeat! Artificial Intelligence for Quantitative Wood Anatomy [Resente et al., 2021]** | | |
| - | RQ1 | • Cells in wood thin sections can be efficiently detected and measured with instance segmentation methods |
| **As good as human experts in detecting plant roots in minirhizotron images but efficient and reproducible: the convolutional neural network "RootDetector" [Peters et al., 2023]** | | |
| - | RQ1 | • Generic semantic segmentation methods provide consistent root length measurements on expert level |
| **BatNet: a deep learning-based tool for automated bat species identification from camera trap images [Krivek et al., 2023a]** | | |
| - | RQ1 | • Camera trap data for bat monitoring can be efficiently analyzed with generic object detection methods |
| **Iterative Next Boundary Detection for Instance Segmentation of Tree Rings in Microscopy Images of Shrub Cross Sections [Gillert et al., 2023b]** | | |
| Ch. 6 | RQ2, RQ3 | • Tree rings cannot be adequately detected with generic instance segmentation methods<br>• Polar coordinates can help modelling the natural growth of shrub branches |
| **Tracking Growth and Decay of Plant Roots in Minirhizotron Images [Gillert et al., 2023a]** | | |
| Ch. 7 | RQ2, RQ3, RQ4 | • Estimation of root turnover requires estimation of non-linear displacement maps due to moving roots and soil<br>• Similar problems from the medical domain are not exactly the same and methods have some issues<br>• Visual root similarity can be learned without additional annotations |
| **Identification and Measurement of Individual Roots in Minirhizotron Images of Dense Root Systems [Gillert et al., 2021]** | | |
| Ch. 8 | RQ2, RQ3 | • Dense root systems are a problem unique to minirhizotron imagery in the ecological context<br>• Root width can be accurately estimated via regression |
| **Towards Combined Open Set Recognition and Out-of-Distribution Detection for Fine-grained Classification [Gillert and von Lukas, 2021]** | | |
| Ch. 9 | RQ2, RQ4 | • Combining body or object parts helps to recognize unknown fine-grained classes<br>• Part detection can be learned without additional annotations |

**Table 4.1:** Overview of the main publications of this PhD project

*Chapter 6* is based on the publication at the CVPR 2023 conference [Gillert et al., 2023b], the largest and most prestigious computer vision venue. It deals with the detection and measurement of tree rings in microscopy images of shrub cross sections, which from a computer vision perspective can be regarded as a special case of the instance segmentation task. Generic instance segmentation methods fail to produce satisfactory results due to the unique concentric shape and topology of the objects and other difficulties.

A new iterative pipeline termed INBD is introduced which models the natural growth direction of the shrub branches, starting from the center (pith) and detecting the next growth ring boundary in each iteration step. To this end, polar coordinates are employed, which impose a prior (RQ3), ensuring a coherent, biologically plausible shape of the rings. Besides, this publication releases a new, publicly accessible and fully annotated dataset of high resolution images for interested computer scientists who might want to continue this line of research by developing better algorithms.

*Chapter 7* has been published at the WACV 2023 conference [Gillert et al., 2023a]. WACV is the largest computer vision conference that puts focus on applications. It describes the first method that considers a time series of minirhizotron images rather than a single image at a time independently of each other. This allows for a more comprehensive estimation of root turnover which would not be possible with previous state of the art and extremely time-consuming to perform by hand.

The algorithm matches roots from two images with the help of a similarity network which gives an estimate of the visual similarity of two roots from two images. Importantly, no additional annotations are used to train this similarity network (RQ4). Instead, pixel-wise augmentations of the same root are used to simulate different environmental conditions, whereas geometric augmentations simulate different roots. This idea behind this is based on the environmental prior, that the movement of roots is usually very restricted to the local area (RQ3).

There are no generic algorithms for this problem, although a similar task exists in the medical domain: deformable image registration, in which observed deformed objects (such as organs from CT scans) are aligned onto a model object. The corresponding methods show performance issues when trying to align roots, because they assume that the objects in the compared images are the same and do not take into account that new objects might grow or old ones disappear, as is the case with roots.

In *Chapter 8*, the context is again minirhizotron imagery. This chapter was published at the CVPPA (Computer Vision for Plant Phenotyping and Agriculture) workshop at the ICCV 2021 conference [Gillert et al., 2021]. It handles a special case of plants with very dense root systems which are prone to get oversegmented with generic methods, skewing the root length and width measurements. These plants belong mostly to the graminoid family and of little interest for agricultural use cases. Therefore this problem has not been tackled previously.

This algorithm is inspired by traditional image processing from before the deep learning era [Steger, 1998] and combines it with modern neural networks. A learned distance transform imposes a prior, isolating closely packed curvilinear structures (RQ3). The resulting improvements are particularly useful in combination with the root tracking algorithm (Chapter 7 [Gillert et al., 2023a]) as they help to find the correct keypoints for the matching of roots.

The final publication in *Chapter 9* is concerned with recognizing unknown classes and was presented at the VISAPP 2021 conference [Gillert and von Lukas, 2021]. Other than previous literature on this topic, fine-grained classes such as animal species are considered. In addition, a distinction is made between *invalid* and *unknown* (but still valid) object classes.

The method asserts that unknown fine-grained species can be easier detected by examining different object parts. Manual annotation is not necessarily needed for this (RQ4). Although the evaluation was performed on generic publicly available datasets for best reproducibility, this problem is highly relevant for a variety of areas in ecology. In the case of this PhD project it is relevant for recognition of rare or unexpected pollen or bat species for which too little or no data is available to train on.

## 4.2 Developed Analysis Tools

The majority of nowadays' AI research is published only as Python scripts or libraries. This can pose a problem, as scripts require a non-trivial level of technical knowledge to understand and use, often offering little to no option for adaptation to new circumstances such as new plant species or new camera trap locations [Tabak et al., 2020]. This is particularly the case when it comes to mitigating their limitations. Obtaining this technical

knowledge has a steep learning curve and requires a significant amount of time, rendering this technology inaccessible to many ecologists. Consequently, a lot of this AI research is not used by the scientists who could benefit from it.

An important part of this PhD project was to develop intuitive tools with a graphical user interface which researchers can use out of the box to perform data analysis without having to spend time on setup and configuration. These tools include capabilities for retraining as well as correcting results and annotating new data. All of them were published as open-source software on the platform GitHub.

All tools share a common software foundation for basic capabilities that are shared among all of them. This base is then modified and extended in the individual sub-projects for new or customized features. The user interface client runs in every modern web browser and was developed in HTML and Javascript communicating with the PyTorch [Paszke et al., 2019] processing backend. This server-client architecture was chosen for one, because browser-based technologies enable the most flexibility in the design of user interfaces and for the other with a potential extension of the tools to online versions in mind.

A version that runs fully in the browser (i.e. without the Python backend) is in the planning for the future and would enable full cross-platform capabilities, making it more accessible to a wider range of users. Modern web standards such as WebGPU [W3C, 2023b] and WebNN [W3C, 2023a] could enable built-in hardware acceleration. Several technical challenges would need to be addressed, especially the retraining functionalities would be difficult to implement with present technologies.

The individual tools are described in the following:

- **CARROT** - **C**ell **A**nd **R**ing **R**ecogniti**O**n **T**ool
  `https://github.com/alexander-g/CARROT`
  This tool serves the quantitative anatomical analysis of wood thin sections in microscopy images. It is currently specialized for samples from tree cores. The cell recognition module is based on the Mask-RCNN [He et al., 2017] architecture and is described and evaluated from an ecological perspective in the publication [Resente et al., 2021]. An additional unpublished tree ring detection algorithm is included. Analysis outputs consist of basic wood anatomical parameters such as cell lumen sizes, grouped by calendar

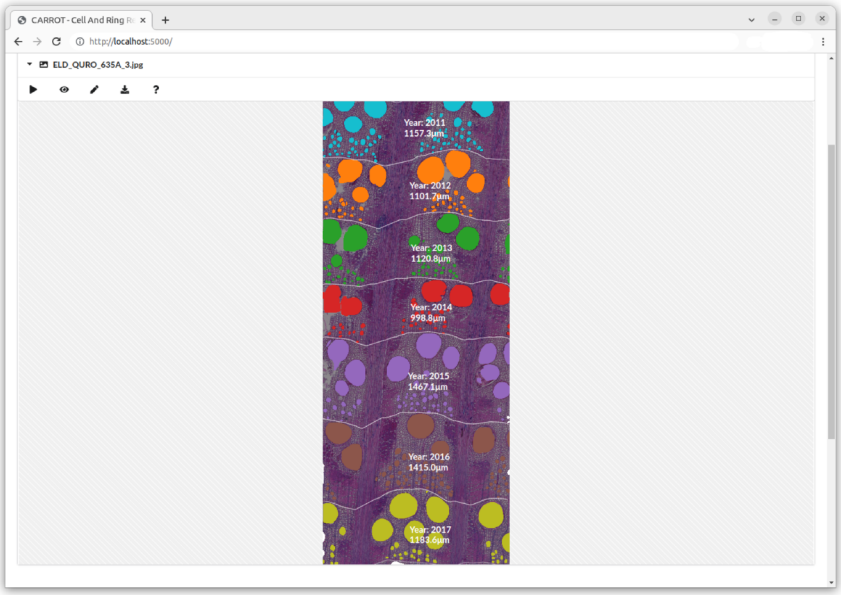



**Figure 4.1:** A screenshot of CARROT, showing detection of cells and tree rings in a microscopic wood thin section

year as well as the corresponding tree ring widths. Four tree species are supported by default and capabilities to adapt to new ones via retraining.

At the time of this writing not yet included are capabilities for analysis of shrub cross sections as they differ significantly from tree cores. The tree ring detection algorithm for this type of samples is part of the main publications of this thesis and described in section 6 [Gillert et al., 2023b]. An extension of this tool to includes this algorithm is planned for future work.

- **Root-Detector**
  `https://github.com/ExPlEcoGreifswald/Root-Detector`
  This analysis tool is designed to process minirhizotron images for research on plant roots dynamics. Its functionalities include basic detection of roots with an optimized U-Net [Ronneberger et al., 2015] architecture resulting in the estimation of total root biomass. Additional features include detection of foreign objects such as tape

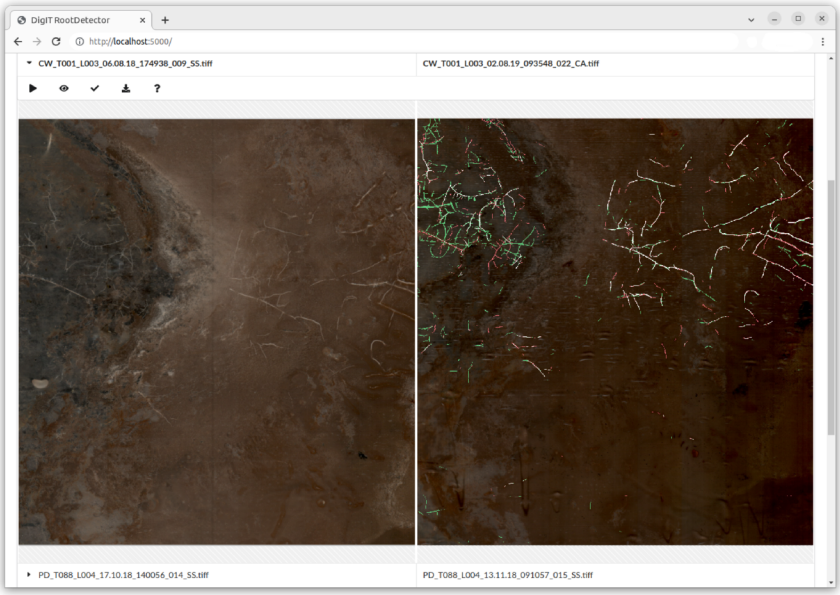



**Figure 4.2:** A screenshot of Root-Detector, showing estimation of root turnover from a pair of minirhizotron images

and morphological postprocessing with the skeletonization procedure [Zhang and Suen, 1984] enables measurements of the more important total root length trait. These measurements are further refined with a more advanced mathematical formula as introduced in [Kimura et al., 1999] for enhanced accuracy. The full system has been evaluated in [Peters et al., 2023], showing performance on the level of human experts.

A more advanced algorithm that is described in Chapter 7 [Gillert et al., 2023a] compares pairs of images from the same minirhizotron experiment, taken at different points in time. This allows for estimation of root turnover, i.e. how much growth and decay occurred over a certain time span. This parameter is of high importance for plant ecology but so far has been only indirectly estimated by hand. To date, there have been no existing automated tools for measuring root turnover, making this algorithm a valuable contribution to this field.

- **BatNet**
  `https://github.com/GabiK-bat/BatNet`
  BatNet is a tool for automated detection and identification of bat

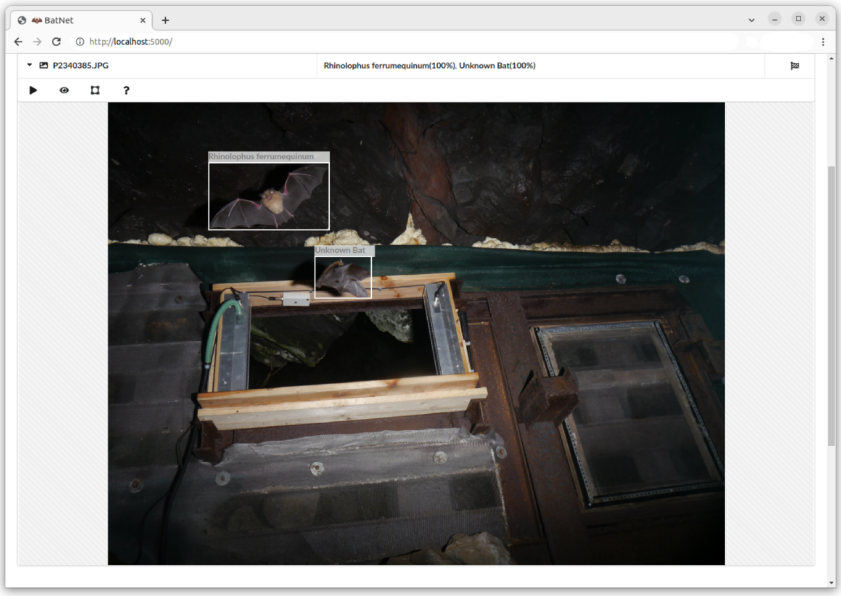



**Figure 4.3:** A screenshot of BatNet, showing detection and classification of bats in camera trap images

species from camera trap images. It is meant to be used by conservation ecologists as well interested amateurs. The main detection pipeline is based on a Faster-RCNN [Ren et al., 2015] object detection architecture, followed by background removal and species identification with an ensemble of 3 classifiers. As many winter hibernacula are dominated by a single bat species, this background removal helps to prevent bias. Neural networks are prone to shortcut learning [Geirhos et al., 2020], i.e. in this case they tend to classify by background rather than learning the identifying features of the bat species. This has been also observed in other camera trap applications such as in [Miao et al., 2019] An ensemble of classifiers helps to produce realistic confidence estimates, as neural networks are known to be overconfident [Guo et al., 2017]. These confidence

estimates are useful for manual verification and correction of the results by the end user.

A more detailed technical description is published in [Krivek et al., 2023a] including an extensive evaluation from a conservationist perspective showing classification high accuracies of 96-98% depending on the site. Conditions under which accuracy suffers such as poor illumination, abnormal camera angle or distance to the infrared light barrier were shown to be mitigable via retraining adaptation.

This tool ships with a pretrained base model that was trained on 13 bat species, that are commonly found in north-eastern Germany. Additional species can be added by retraining on new data if used in different regions. Such a use case has been evaluated in [Krivek et al., 2023a] as well, showing that a small number of new image is usually sufficient.

- **Tofsi-POST** - **T**rainable **O**bject **F**inder, **S**elector and **I**dentifier for **Po**llen, **S**pores and Other **T**hings
  `https://github.com/alexander-g/Tofsi-POST`

  Named after Selma 'Tofsy' von Post, the wife of the pioneer of pollen analysis Lennard von Post, mentioned in subsection 2.4. She did most of the manual pollen counting for her husband, without acknowledgement [Birks and Berglund, 2018]. This tool is intended for use by paleoecologists or palynologists for working on microscopy images from lake sediment samples.

  Microscopy images are often acquired as z-stacks, i.e. with multiple focus settings in order to identify details that may not be visible from a single point of focus. This is of particular importance for species identification of pollen grains which often exhibit very fine-grained surface patterns. For detection and localization of the pollen grains, a focus stacking algorithm merges the individual z-stack layers. This algorithm is based on the exposure fusion algorithm introduced in [Mertens et al., 2009] for High Dynamic Range (HDR) imaging. The classification of the pollen species is then performed on individual z-stack layers to avoid loss of visual details.

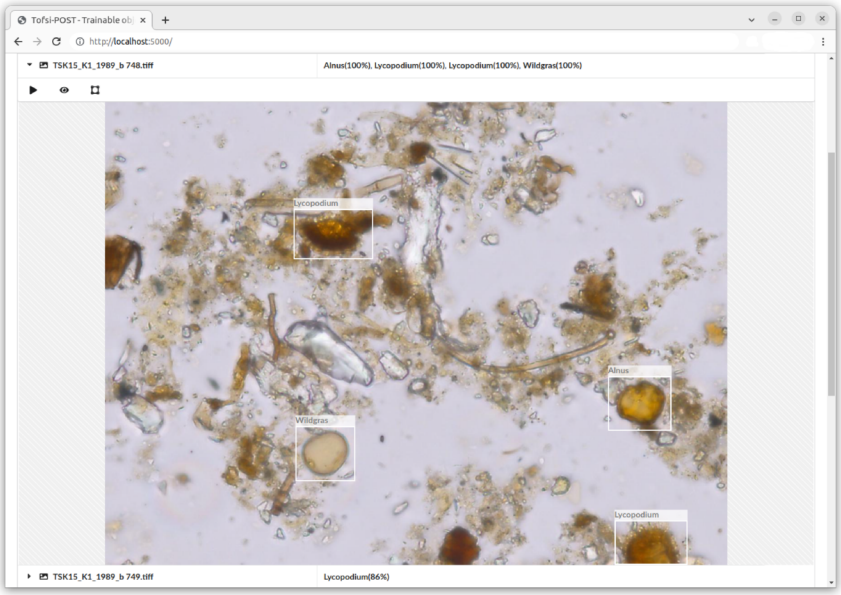



**Figure 4.4:** A screenshot of Tofsi-POST, showing detection and classification of pollen in microscopy images

Here too, a pretrained model that is able to detect 9 pollen species is provided by default. The users can retrain on their own data to adapt to additional species.

# 5. Conclusions and Outlook

The term *hardware lottery* [Hooker, 2020] refers to the trend in current machine learning and computer vision research that new algorithms are developed based on the capabilities of existing hardware. Modern GPUs and TPUs incorporate special computing units to optimize convolutions, or more recently, self-attention, leading to a "free lunch" for algorithms that exploit these mechanisms (CNNs and Vision Transformers). This results in the neglect of other algorithmic techniques that might be more effective for certain tasks but are not explored due to the dominance of established approaches and lack of efficient hardware support. For instance, evolutionary algorithms possess interesting properties that could provide advantages over neural networks, but their slow performance on readily available off-the-shelf processors and accelerators makes them less appealing compared to algorithms that have the support of modern hardware and thus little research is done in this direction.

Similarly, in other scientific domains such as ecology, research is often conducted based on the availability of tools. Studies are kept small in scale because the volumes of data to process manually is too large to handle, potentially missing out on novel insights. This thesis has shown that AI-based automated analysis tools can help overcome these limitations and expand the scale of studies and potentially lead to more comprehensive, accurate and faster findings.

The new tools and algorithms that were developed during this project are already being used for ecological studies. For instance in [Schwieger et al., 2022] the *Root-Detector* minirhizotron analysis tool was used to analyze the effects of rewetting of drained peatlands on the growing season of roots. This tool has also led to new experiments and expansion of existing ones: a total of 227 new minirhizotrons were installed in 4 newly acquired, publicly funded projects with one additional project in

preparation. Moreover, an expansion of the camera trap experiments to a Germany-wide bat monitoring network is currently in preparation, thanks to the automatic processing capabilities of *BatNet*.

Despite its contributions, this thesis only scratches at the surface of what is achievable for ecological sciences. It can only be seen as an introduction to new problems that have not yet been addressed. For instance, the root tracking algorithm in Chapter 7 [Gillert et al., 2023a] only considers two images at a time. A straightforward extension to an arbitrary time series length would allow for measurements of life spans of individual roots and thus more accurate root turnover estimates, but several technical and algorithmic challenges would need to be overcome in order to accomplish this.

In a similar vein, several enhancements for the shrub tree ring detection algorithm INBD from Chapter 6 [Gillert et al., 2023b] are possible. It was tested only on a selected dataset of comparatively simple images, whereas real world ecological studies, often deal with more complex individuals containing much larger numbers of tree rings, sometimes exceeding one hundred. In addition, samples from harsh climatic conditions such as the high Arctic, contain many ambiguities that even experts find challenging to interpret. To achieve higher degrees of automation, a reliable confidence measure would be needed.

All of this highlights the existence of several unsolved problems in the field of ecology yet to be explored. Addressing them goes beyond the scope of this thesis and thus left for future work.

At the time of this writing, large generic *foundation models* such Segment Anything (SAM) [Kirillov et al., 2023] that contain billions of parameters and are trained on huge datasets and hardware inaccessible to most researchers, are generating news headlines, even in popular media outlets. They are designed to work across a wide range of applications and indeed show impressive performance on a variety of tasks. Yet, they struggle on highly specialized problems. For instance, SAM shows very limited segmentation performance on use cases such as polyp detection inside the gastrointestinal tract, road network extraction from remote sensing, anomaly detection, nighttime driving and many more [Ji et al., 2023].

This thesis argues that generic methods have inherent limitations in specialized problems in various fields, such as in ecology. Specialization is required to solve them and domain knowledge in form of priors can be the key to achieve this.

# Part II

# Peer-Reviewed Scientific Publications

# 6. Iterative Next Boundary Detection for Instance Segmentation of Tree Rings in Microscopy Images of Shrub Cross Sections

| | | |
|---|---|---|
| Title | Iterative Next Boundary Detection for Instance Segmentation of Tree Rings in Microscopy Images of Shrub Cross Sections | |
| Authors | Alexander Gillert, Giulia Resente, Alba Anadon-Rosell, Martin Wilmking, Uwe Freiherr von Lukas | |
| Publication Venue | IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) 2023, Vancouver, Canada | |
| Status | Accepted | |
| DOI | 10.48550/arXiv.2212.03022 (preprint) | |
| Venue Ratings as of 2023-03-05 | CORE2021[1] | A* |
| | Research.com[2] | 63.10 (Impact Score)<br>1st among all computer science venues |
| | Google Scholar[3] | 389 (h5-Index)<br>1st among all computer science venues<br>4th among all scientific venues |

[1] http://portal.core.edu.au/conf-ranks/604/

[2] https://web.archive.org/web/20230305125228/https://research.com/conference-rankings/computer-science/computer-vision

[3] https://web.archive.org/web/20230305124904/https://scholar.google.com/citations?view_op=top_venues&hl=en

# Iterative Next Boundary Detection for Instance Segmentation of Tree Rings in Microscopy Images of Shrub Cross Sections

Alexander Gillert[1] Giulia Resente[2] Alba Anadon-Rosell[3]
Martin Wilmking[2] Uwe Freiherr von Lukas[1,4]

[1]Fraunhofer Institute for Computer Graphics Research (IGD), Rostock
[2]Institute of Botany and Landscape Ecology, Ernst Moritz Arndt University, Greifswald
[3]Centre for Research on Ecology and Forestry Applications (CREAF), Barcelona
[4]Institute for Visual & Analytic Computing, University of Rostock
{alexander.gillert, uwe.freiherr.von.lukas}@igd-r.fraunhofer.de

## Abstract

*We address the problem of detecting tree rings in microscopy images of shrub cross sections. This can be regarded as a special case of the instance segmentation task with several unique challenges such as the concentric circular ring shape of the objects and high precision requirements that result in inadequate performance of existing methods.*

*We propose a new iterative method which we term Iterative Next Boundary Detection (INBD). It intuitively models the natural growth direction, starting from the center of the shrub cross section and detecting the next ring boundary in each iteration step. In our experiments, INBD shows superior performance to generic instance segmentation methods and is the only one with a built-in notion of chronological order.*

*Our dataset and source code are available at http://github.com/alexander-g/INBD .*

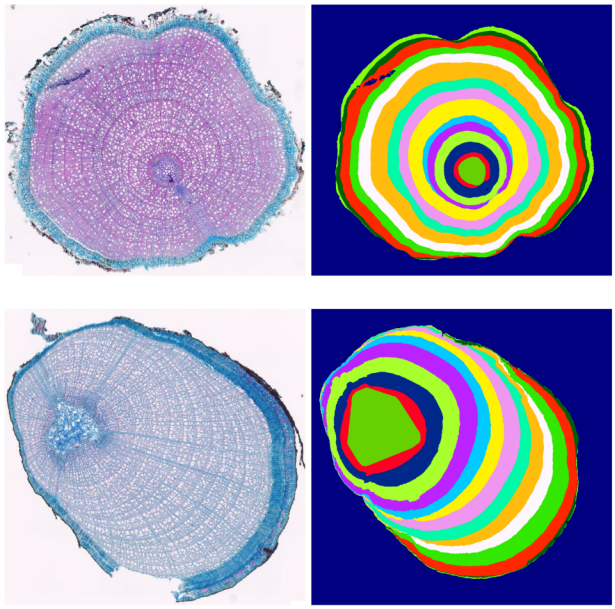


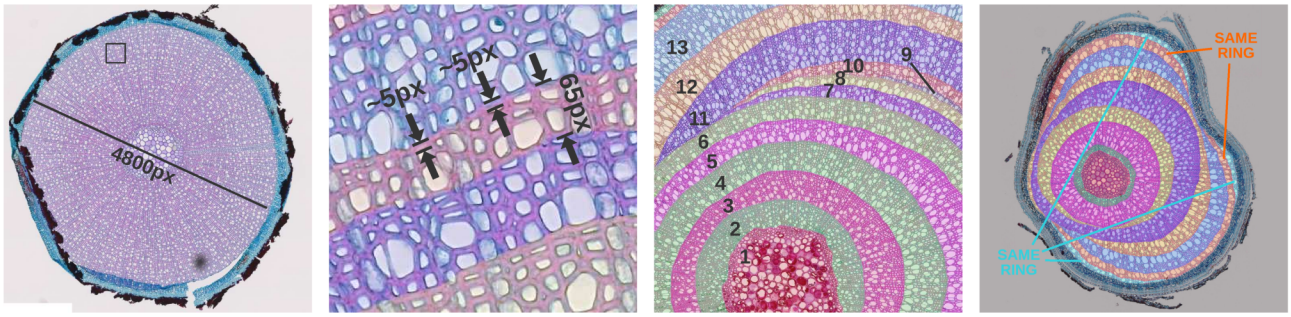
Figure 1. Example microscopy images (left) of shrub cross sections from our new dataset and the outputs (right) of our proposed method INBD for instance segmentation of tree rings

## 1. Introduction

Dendrochronology is the science that provides methodologies to date tree rings [4], i.e. measuring and assigning calendar years to the growth rings present in a wood stem. By analyzing anatomical properties like ring widths or the cell sizes within the rings, dendrochronology can be applied to dating archaeological manufactures, tracking timber sources or reconstructing past climate conditions [11].

For climate reconstruction in the Arctic, shrubs constitute the most important source of dendrochronological information, since they are the only woody plants able to thrive there [23]. As temperature is a limiting factor for shrub growth in the Arctic, it shows a strong relationship with climate, making these plants a reliable proxy to reconstruct past climate events [24]. Dendrochronological analyses on shrubs are usually performed on thin cross sections of branches or roots and observed under the microscope with a magnification that allows ring identification at a cellular level. As of now, ecological studies are limited in size by the amount of manual analysis work due to the lack of automatic tree ring detection methods.

With this paper we want to introduce this problem to the




(a) High resolution and faint boundaries (b) High precision requirements (c) Wedging rings (d) Disconnected rings

Figure 2. Some of the challenges encountered in this task:
(a) Boundaries inbetween tree rings are often hard to recognize. For example, this cross section contains 14 rings.
(b) Crop of the previous image (indicated by the square) with overlayed annotation. A tree ring is only 65 pixels wide or ca 1.4% of the full cross section diameter. The cell wall that divides late summer cells and the next year's early summer cells is only 5 pixels wide or 0.1%.
(c) Wedging rings can complicate finding the chronologically correct next year ring.
(d) Rings can grow in multiple disconnected parts from different sides.

computer vision community and enhance the capabilities for ecological sciences. We release a new dataset containing high resolution microscopy images of shrub cross sections and propose a specialized method for growth ring identification. Example images from our dataset and corresponding outputs of our method are shown in Figure 1. From a computer vision point of view, this can be regarded as a special case of the instance segmentation task, however it differs from previous generic datasets in several ways which makes existing methods underperform.

Figure 2 illustrates these differences. For one, the concentric ring shape of the instances can pose a significant obstacle, particularly for top-down methods because the objects have almost identical bounding boxes. This gets complicated by the fact that year rings can also form incomplete circles (wedging rings) and grow from only one side, or even in multiple disconnected parts from different sides (2d). Depending on the species, plant part and climatic conditions the amount of wedging rings can range from zero to being the majority. Assigning the correct order to wedging rings can be an issue where rings of more than 2 years touch each other (2c). Bottom-up methods on the other hand struggle with faint ring boundaries (2a) as the presence of the boundary pattern is not always constant throughout the whole stem circumference. They are prone to merging rings where no boundary can be detected or splitting them where the ring width is narrow. Next, the images are acquired at a high resolution (2a) to capture cellular information, yet a high degree of precision is required for the downstream task of assigning individual cells to the correct year. The thickness of a cell wall that is dividing the cells from one ring to another can be as low as 0.01% of the whole object (2b). Finally, as the preparation of samples and annotation of the images is very costly, training has to be performed in a low data regime.

We argue that a specialized approach can help to overcome those challenges and propose a new iterative method which we term Iterative Next Boundary Detection (INBD). In the first step, it performs semantic segmentation to detect basic features such as the background, center and the ring boundary pixels. From this starting point, it iteratively detects the next year ring's boundaries, following the natural growth of the plant. This process is augmented with a recurrent wedging ring detection module to counteract issues with incomplete rings. We compare our method with both top-down and bottom-up generic instance segmentation in our experiments in which it shows better results. Moreover, it is the first method that automatically assigns a chronological order to the detected objects.

The contributions of this paper can summarized as follows:

- Publication of a new challenging dataset for a special case of instance segmentation.
- Development of the specialized method INBD for tree ring instance segmentation.
- Evaluation of previous generic instance segmentation methods and comparison with INBD

## 2. Related Work

Instance segmentation is a widely studied problem in computer vision, commonly benchmarked on a variety of standard generic datasets such as COCO [12] which contains photographs of everyday objects or the the more specialized CREMI 2016 [6] challenge for cell segmentation in

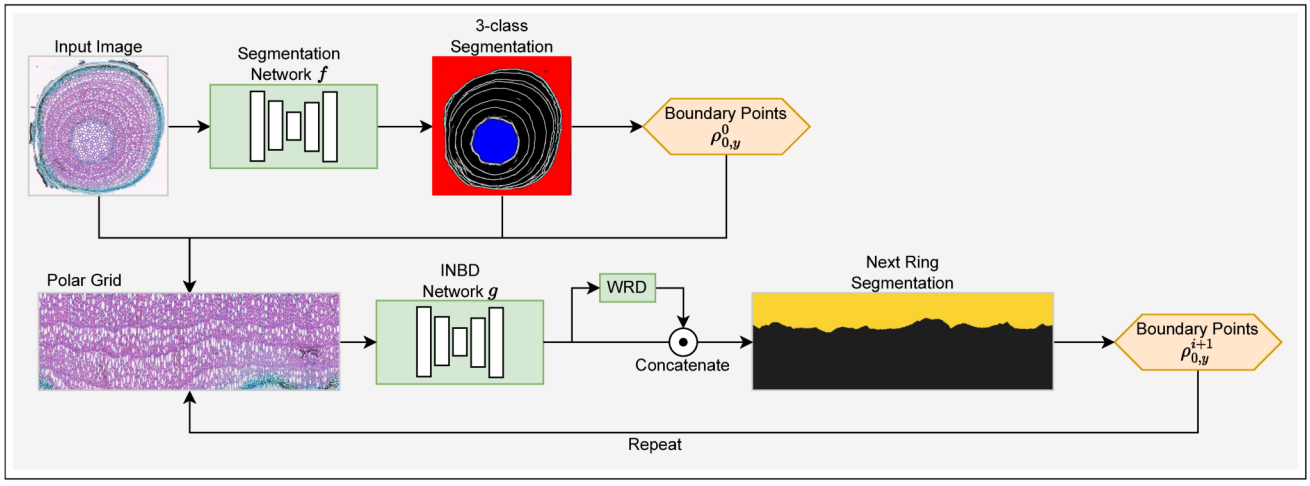



Figure 3. Overview of the INBD pipeline. An input image is first passed through a generic semantic segmentation network that detects 3 classes: background, ring boundaries and the center ring. A polar grid is sampled starting from the the detected center ring and passed to the main INBD network that detects the next ring. This process is repeated until the background is encountered.

electron microscopy images. No publicly available dataset is known to us that contains concentric ring shaped and ordered objects.

Methods can be categorized into *top-down* and *bottom-up* procedures. Mask-R-CNN [8] is the most widely used architecture and belongs to the top-down group. It relies on an object detector to first detect bounding boxes of objects which are then segmented. This fails on overlapping or as in our case concentric objects due to non-maximum suppression. Moreover, it can only generate low resolution masks. Contour methods such as Deep Snake [17] or DANCE [13] can generate masks with higher precision but still require an upstream object detector.

Bottom-up methods for instance segmentation methods work by first computing object boundaries or affinities and then clustering the resulting superpixel graph into whole objects via the multicut objective. Finding the optimal solution for this is known to be NP-hard [10], therefore several approximate solvers such as GASP [2] have been developed. These methods perform significantly better on our dataset but still show deficits in cases where object boundaries are hard to recognize and they cannot handle disconnected rings (such as in Fig. 2d).

None of the above methods has a built-in notion of sequence order of the detected objects that would be needed to assign a tree ring to a year.

Application of deep learning methods to ecological purposes is nowadays an established procedure [3] due to the complexity related to ecological investigations and the use of increasingly larger datasets. Specifically for quantitative wood anatomy (QWA), deep learning research has so far focused mostly on detection and measurement of cells such as in [7,19]. Tree ring detection was subject in [5,18], however only on scans or photographs of mature wood core samples rather than full cross sections as in our case.

ROXAS [21,22] is the most commonly used analysis tool in QWA, however it is based on traditional image processing methods and not on deep learning which makes it sensitive to sample processing and image quality. It also contains tree ring detection functionality which works by line-following early summer cells but requires domain knowledge for manual tuning of many species-specific parameters like cell shape and size.

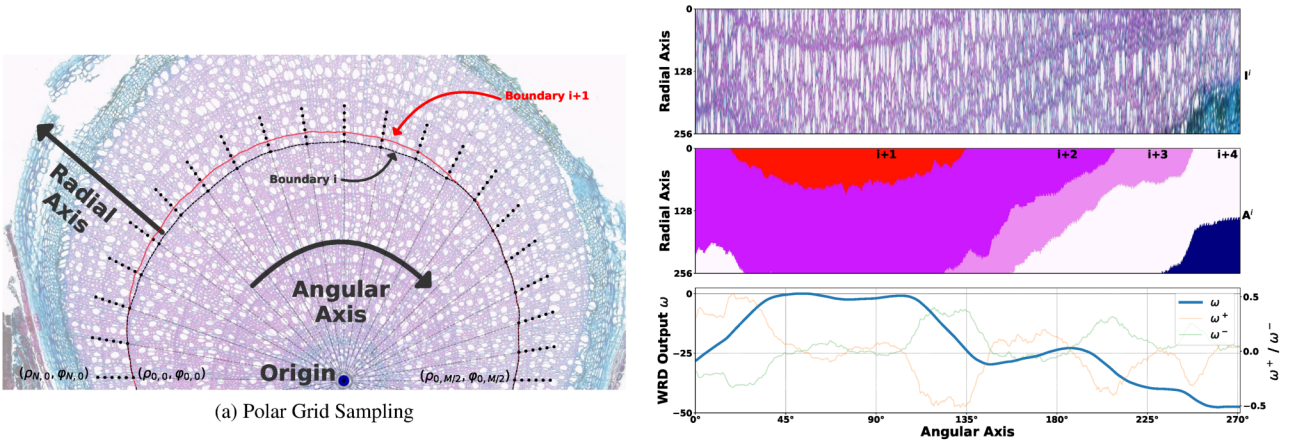
## 3. Method

On a high level, INBD simply modifies and extends the various contour based methods like Deep Snake [17] or PolarMask [26] with an iterative inference procedure. In reality, this requires several important design choices to make this perform well. The influence of the individual design choices is analyzed in an ablation study in subsection 5.2. An architectural overview of the INBD pipeline can be found in Figure 3.

### 3.1. Network Architecture

The INBD pipeline is composed of two neural networks. The first network is a simple semantic segmentation network that is trained to detect three classes: background, ring boundaries and the center ring (or pith). We denote this network and its output with $f(I) = (y^{bg}, y^{bd}, y^{ct})$ when applied on image $I$. We select an architecture based on U-Net [20] with a pretrained backbone. The three classes are trained with a combination of cross-entropy loss and the Dice loss [16]:

$$L_f = \lambda_1 L_{CE}^{background} + \lambda_2 L_{Dice}^{boundaries} + \lambda_3 L_{CE}^{center} \quad (1)$$




(a) Polar Grid Sampling

(b) Wedging Ring Detection

Figure 4. Visual explanation of the concepts. (a) We sample on a polar grid starting from the previous detected ring boundary. The number of points is reduced for better visualization. (b) shows the resulting input to the network $g$ (top), the corresponding annotation (center) and the output signal $\omega$ of the wedging ring detection module (bottom). $\omega$ accumulates along the angular axis, rising on start and falling on end of wedging rings.

with $\lambda = (0.01,\ 1.0,\ 0.1)$ balancing coefficients to account for class imbalances. Due to the large size of the images and for a larger field of view, $f$ operates on $\times 0.25$ of the original resolution.

Our main network, which we denote with $g$, is another 2D convolutional segmentation network that classifies each pixel as belonging to the next ring or not. By choosing a 2D network as opposed to a 1D one, as in many contour methods such as Deep Snake [17], we can leverage transfer learning since we are working in a low data regime and in addition to that we can reject and interpolate ambiguous predictions (see below, Eq. 4). This second network has mostly the same architecture as the first one, except that we replace the normal 2D convolutions with circular convolutions to wrap around the full circle, as also used in Deep Snake [17]. The circularity is only applied to the angular axis (see below).

### 3.2. Polar Grid

The network $g$ operates on "unrolled" rings $I^i \in \mathbb{R}^{[C \times N \times M]}$ sampled on a polar grid $P^i \in \mathbb{R}^{[N \times M]}$ with $N = 256$ a fixed resolution in the radial dimension and $M$ an adaptive resolution in the angular dimension. The polar grid origin is computed from the center of mass of the center ring $y^{ct}$ as detected by $f$. Polar coordinates impose a prior, ensuring a coherent (quasi-)convex shape, contrary to Cartesian coordinates.

We express the sampling points for ring $i$ as polar coordinates $(\rho^i_{xy}\ ,\ \varphi^i_{xy})$, with $x \in [0, N]$, $y \in [0, M]$ indices within the grid. The boundary point radii for the second ring $\rho^1_{0,y}$ are inferred directly from the detected center ring $y^{ct}$.

Estimating the extent of the grid in the radial dimension (i.e. $\rho_{N,y}$) is crucial: if too short, the next ring will be cut off, if too long, the next ring might get skipped and not detected at all. For this, we compute the distances to the closest positive value in $y^{bd}$ for each angle $\varphi$ and set the extent to $1.5\times$ 95%-th percentile of these distances, to make sure that most points are included but also to filter outliers. This was empirically verified to cover all rings in our dataset. The remaining radial values $\rho$ are then uniformly distributed along this range: $\rho_{x,y} = \frac{1}{N}(\rho_{N,y} - \rho_{0,y})x + \rho_{0,y}$

The angular resolution $M$ is computed so that the angles $\varphi$ have an approximately uniform euclidean distance to each other across rings: since the outer rings have a larger circumference than the inner ones they should be sampled at a higher angular resolution $M$. The value $M^i$ for ring $i$ is computed from the previous ring's average radii:

$$M^i = \alpha \frac{1}{M^{i-1}} \sum_{y}^{M^{i-1}} \rho^{i-1}_{0,y} \tag{2}$$

with $\alpha$ a hyperparameter that controls the general density of $M^i$ which we set to $2\pi$ where not otherwise mentioned. The angles $\varphi$ are spaced uniformly along the full circle: $\varphi_{x,y} = \frac{1}{M} 2\pi y$

The channel dimension $C = 7$ is composed of the RGB channels of the input image, the detected "background" and the "boundaries" outputs $y^{bg}$ and $y^{bd}$ from $f$, normalized radii $\rho$ and the output of the wedging ring detection module (see below) concatenated together.

The main loss $L^{cls}_{CE}$ for network $g$ is the standard cross-entropy loss to classify each pixel in the polar grid as belonging to the next ring or not, according to the annotation $A^i$, sampled on the same polar grid.

### 3.3. Inference

To perform inference of the next ring's boundary points $\rho_{0,y}^{i+1}$, we select last positive point in the output $g(I^i)$ column-wise, where it is unambiguous:

$$\tilde{X}_y = \{x, \text{ where } g(I^i)_{x,y} = 1\} \tag{3}$$

$$\rho_{0,y}^{i+1} = \begin{cases} \rho^i_{\max \tilde{X}_y, y}\,, & \text{if } \max \tilde{X}_y = \min \overline{\tilde{X}_y} - 1 \\ \text{undefined}\,, & \text{otherwise} \end{cases} \tag{4}$$

Ambiguous values linearly interpolated. Importantly, the interpolation should be performed on polar coordinates and not on Cartesian ones and wrap around the circle. This detection process is repeated iteratively with the new predicted ring boundary points $i+1$ as the starting point to detect the ring $i+2$ until the background $y^{bg}$ that was detected by the segmentation network $f$ is reached.

### 3.4. Wedging Ring Detection

The method as described so far is able to detect full tree rings sufficiently well but struggles with wedging rings. More specifically, it is prone to skipping a ring boundary in locations where the wedging ring is far away and outside the field of view, e.g. as in Figure 2c when trying to detect the next boundary after ring 6. To counteract this issue we insert a wedging ring detection (WRD) module before the final classification layer.

This module consists of 3 additional convolutional layers with two output channels. The two channels are averaged along the radial axis into 1-dimensional signals $\omega^+$ and $\omega^- \in \mathbb{R}^M$ and combined via a recurrent mechanism:

$$\omega'_0 = \beta \tag{5}$$

$$\omega'_\varphi = \sigma(\omega^+_{\varphi-1}) - \sigma(\omega^-_{\varphi-1}) \tag{6}$$

$$\omega_\varphi = \omega'_\varphi - \max \omega' \tag{7}$$

where $\sigma$ is the sigmoid function and $\beta$ is a starting point constant. Intuitively, $\omega^+$ is responsible for detecting the start of a wedging ring and increases the output signal $\omega$, whereas $\omega^-$ detects the end and decreases it. $\omega$ is then forwarded to the final classification layer by concatenating it to the features along the channel dimension.

During inference, the choice of $\beta$ does not matter because of the normalization by subtracting the maximum (Eq. 7). This ensures a standardized representation of the signal to the following downstream classification layer, irrespective of the starting point $\beta$. High values close to zero indicate valid locations (next ring or $i+1$), whereas low values are invalid locations (next but one or $i+2$). This functionality is illustrated graphically in Figure 4b.

Although in theory the network could derive useful information from this module by itself, we have found that in practice it is highly beneficial to add an explicit training signal. Again, we use the cross entropy loss, but modified for the single dimension and applied on the unnormalized signal $\omega'$:

$$L^{wrd} = A^{wrd}_\varphi \; log\; \sigma(\omega'_\varphi) + (1 - A^{wrd}_\varphi)\; log\; 1 - \sigma(\omega'_\varphi) \tag{8}$$

$$A^{wrd}_\varphi = \begin{cases} 1, & \text{where } A^i_{0,\varphi} = i+1 \\ 0, & \text{otherwise} \end{cases} \tag{9}$$

During training $\beta$ is set so that $\sigma(\omega') = 0$ if the ground truth at angle $\varphi = 0$ is low, or so that $\sigma(\omega') = 1$ if it is high, to avoid incorrect training signals (we choose $\beta = \pm 15$).

An example where this module helps to catch an error is shown in Figure 5.

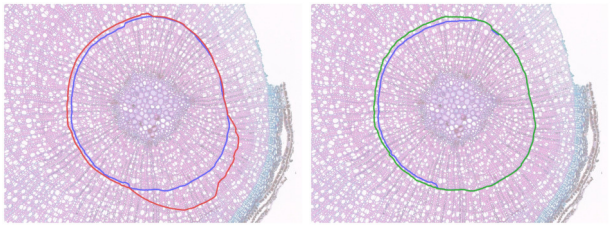


Figure 5. INBD can be prone to skipping boundaries. In this example, the wedging ring detection module helps to catch mistakes like this. (Left: without WRD, right: with WRD)

The final loss for network $g$ is defined as $L_g = L^{cls}_{CE} + \lambda_4 L^{wrd}$ with $\lambda_4 = 0.01$. We have found higher values to have a negative impact on the main classification loss.

### 3.5. Training Procedure

Since INBD is an iterative procedure, errors caused by an earlier ring get easily propagated onto the later rings. It can however also recover from previous mistakes if trained with an *iterative training* procedure: rather than using only the (near-perfect) boundary points from the annotation, the training loop should incorporate previous (possibly faulty) predictions as the starting point for polar grids. Listing 1 shows the high-level pseudo code for one training epoch.

Listing 1. Pseudo-code for one training epoch

```
for (image I, annotation A, ring i) in dataset:
  L_g = 0
  ρ^i_{0,y} = boundary_from_annotation(A, i)
  loop i = i..i+n:
    ρ̂^i_{0,y} = augment(ρ^i_{0,y})
    I^i = sample_polar_grid(I, (ρ̂^i, φ^i))
    y^i = g(I^i)
    L_g += compute_loss(y^i, A^i)
    ρ^{i+1}_{0,y} = compute_boundary(y^i)
  backpropagate( L_g / n )
```

| Subset | Species | Training Images | Test Images | Number of rings | Average diameter | Example Images |
|---|---|---|---|---|---|---|
| DO | Dryas octopetala | 22 | 42 | 544 | 3700px | Figs. 1 (bottom), 7 (bottom) |
| EH | Empetrum hermaphroditum | 24 | 58 | 949 | 3260px | Figs. 1 (top), 2c, 2d |
| VM | Vaccinium myrtillus | 22 | 45 | 494 | 3979px | Figs. 2a, 2b, 7 (top) |

Table 1. Overview of our dataset

Where not otherwise mentioned we use $n = 3$ iterations per epoch.

Besides the conventional data augmentations such as the pixel-wise color jitter operations we employ additional augmentations specific to polar grids such as varying the boundary points:

$$\hat{\rho}_{0,y} = \rho_{0,y} + \cos(\varphi_{0,y} + X_0)\gamma_0 + X_1\gamma_1 \qquad (10)$$

with $X \sim \mathcal{U}(-1, 1)$ random variables and $\gamma$ hyperparameters.

Both networks are trained separately with the AdamW [15] optimizer for 100 epochs, 1e-3 base learning rate and cosine annealing [14] learning rate schedule.

## 4. Experimental Setup

### 4.1. Dataset

Our dataset consists of overall 213 high-resolution images. It is split into 3 subsets according to the plant species. An overview is provided in Table 1. The amount of images is rather low due to the high cost of sample preparation as well as annotation: a single image containing a large amount of rings can take up to 6 hours to annotate by hand. The dataset and annotations are publicly available at http://github.com/alexander-g/INBD .

The shrub samples were collected at subalpine, alpine and subarctic sites across the Pyrenees, Southern Norway and Northern Sweden. Aboveground shoots (ramets) were clipped at the stem base, above the soil surface. In the lab, the samples were cut into 15-20 $\mu m$ cross-sections with a rotary microtome, stained with a mixture of 1:1 safranin and astrablue, rinsed with ethanol solutions, embedded in Euparal, dried and finally scanned in a slide scanner to obtain high resolution images.

### 4.2. Compared Methods

As there are no specialized methods for tree ring detection in shrub cross sections yet, we compare our method with generic instance segmentation methods. From the top-down category we compare with Mask-R-CNN [8] and Deep Snake [17]. Mask-R-CNN is trained in two modes: in the *hollow* (h) mode, objects are defined as single calendar years and are donut-shaped (with a hole), whereas in the *filled* (f) mode, objects consist of multiple years (and have no holes). We use the implementation from the torchvision (v0.11) framework. The non-maximum suppression is increased to 0.7 to reduce the filtering of overlapping detections and the images are downscaled to accommodate for GPU memory limits. For Deep Snake only the filled mode is used because it cannot model hollow objects.

In the bottom-up group we select Multicut [10] and GASP [2] for comparison. We use the implementation from the PlantSeg [25] source code which was developed in part by the original GASP algorithm authors. For a fair comparison, the detected boundaries from the same segmentation network $f$ as for INBD are used. We have found bottom-up methods require species-specific tuning of hyperparameters. We have tested several combinations and report only the best ones here. More information can be found in the supplement.

### 4.3. Metrics

Our main evaluation metric is the mean Average Recall (mAR) as defined in [9] averaged at IoU=.50:.05:.95 intervals. We do not use the mean Average Precision (mAP) that is often used in generic instance segmentation literature, as we regard instance recall as more important than precision: it is easier for the end user to delete false positive objects on manual inspection than adding new ones.

We additionally report the Adapted Rand errors (ARAND) as defined in [1] because this metric is more commonly used in the bottom-up literature. It can be interpreted as the harmonic mean of the pixelwise precision and recall values.

## 5. Results

### 5.1. Method Comparison

The main results of the compared methods are presented in Table 2. For all metrics we observe consistently better performance of INBD over the compared methods.

Top-down methods show very unsatisfactory performance. The filled mode gives a small performance boost but the results are still too inaccurate to be useful, particularly missing many thin rings. Deep Snake struggles remarkably, often detecting only one or two rings at most.

| | mAR↑ | | | ARAND↓ | | |
|---|---|---|---|---|---|---|
| Method | DO | EH | VM | DO | EH | VM |
| Mask-R-CNN (h) | .106 (.008) | .144 (.003) | .185 (.008) | .644 (.007) | .694 (.002) | .532 (.004) |
| Mask-R-CNN (f) | .210 (.006) | .176 (.004) | .218 (.002) | .441 (.002) | .499 (.001) | .425 (.007) |
| Deep Snake (f) | .061 (.011) | .015 (.001) | .019 (.008) | .524 (.024) | .620 (.003) | .584 (.027) |
| GASP | .374 (.002) | .667 (.004) | .576 (.014) | .313 (.003) | .144 (.003) | .168 (.010) |
| Multicut | .387 (.008) | .688 (.005) | .596 (.006) | .301 (.001) | .132 (.004) | .154 (.005) |
| INBD (ours) | **.553** (.011) | **.738** (.018) | **.704** (.014) | **.196** (.009) | **.113** (.010) | **.112** (.007) |

Table 2. Method comparison. Values are averaged over 3 full training runs with the standard deviation provided in parentheses. (h) refers to the hollow mode, (f) to the filled mode. ↑ denotes higher is better, ↓ lower is better.

We attribute this to its base detector CenterNet [27] which inherently fails with concentric objects.

The bottom-up methods can compete with INBD on EH thanks to relatively well recognizable ring boundaries in this subset. The VM and especially DO subsets on the other hand have much less pronounced and sometimes ambiguous boundaries which often cannot be detected at all. This is particularly a problem for the bottom-up methods which are then prone to incorrectly merging two rings. INBD on the other hand can interpolate ambiguous locations (Eq. 3). The results of GASP and Multicut are very similar to each other, as also noted in [25].

In general, we observe that INBD is better at detecting difficult rings. This observation is confirmed in the more fine-grained analysis in Figure 6 which shows the recall values for the individual IoU thresholds. INBD scores only slightly better on the high threshold recalls such as AR90 or AR95 which are usually the easily recognizable rings. The real benefits come from detecting harder examples.

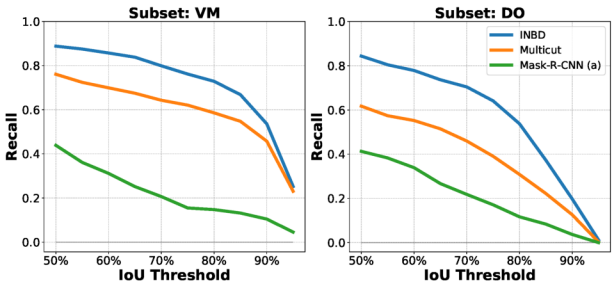



Figure 6. Recall over IoU for the compared methods

## 5.2. Ablation Study

In Table 3 we show how the individual design choices as proposed in section 3 affect the detection performance of INBD. Two baselines of INBD without the adaptations are evaluated, one with Cartesian and another with polar coordinates. We note that those two implementations are not fully comparable, more details in the supplement.

Our experiments show that increasing the angular resolution (Eq. 2) in order to keep the Cartesian resolution roughly constant across rings yields almost a 3 mAR percentage points improvement. Interpolating ambiguous boundary points (Eq. 4) is highly important and improves the mAR by more than 6 points. Iterative training (subsection 3.5), i.e. training with previously predicted boundary points (as apposed to only using the annotation) gives an additional performance boost of more than 3 mAR. Finally, the WRD module (subsection 3.4) helps with wedging rings. As wedging rings are comparatively few in numbers, the performance gain is relatively moderate but consistent among training runs.

| Configuration | mAR↑ | ARAND↓ |
|---|---|---|
| Cartesian coordinates baseline | .498 | .237 |
| Polar coordinates baseline | .601 | .218 |
| + adaptive angular resolution M | .629 | .190 |
| + ambiguous boundary interpolation | .691 | .146 |
| + iterative training | .722 | .126 |
| + wedging ring detection | .738 | .113 |

Table 3. Influence of design choices on the performance. All values refer to the EH subset.

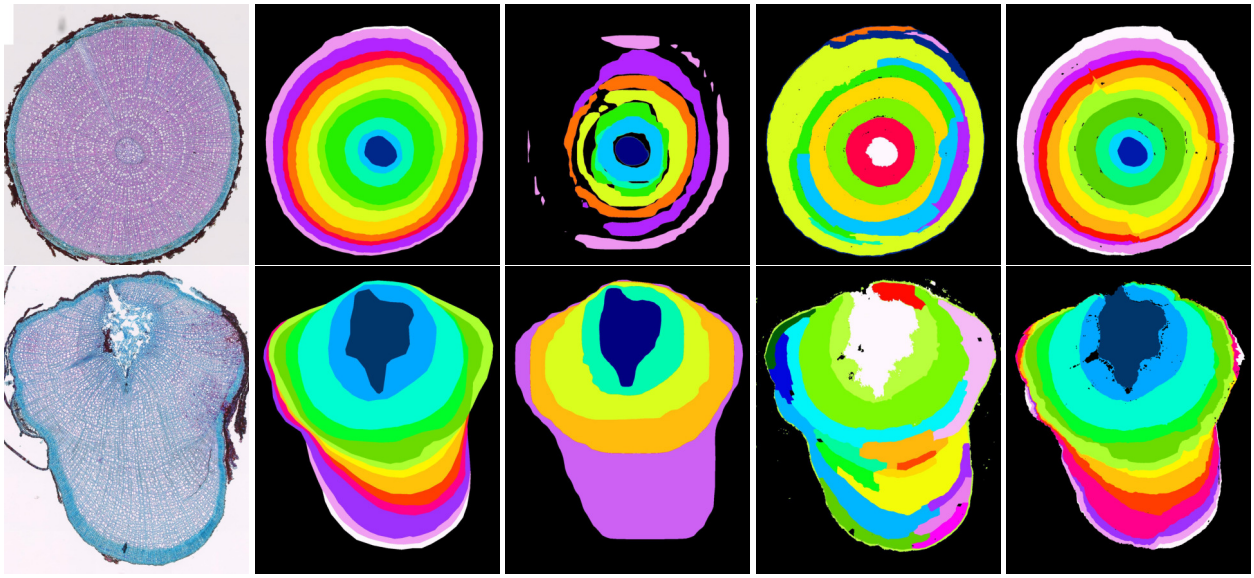
Additional evaluations on the effect of hyperparameters on the detection performance can be found in the supplementary materials.

## 5.3. Cross-species Performance

Dendro-ecological studies are rarely limited to the three plant species from our dataset, end users might want to analyze new species, for which trained models are not yet available. Therefore we test how well the compared methods generalize to unseen species. The results are presented in Table 4.

EH and VM show some level of similarity to each other and methods trained on one set can be used to a limited degree on the other one. These results might be insufficient for downstream tasks but could be used to generate new annotations for retraining, faster than creating them manually from scratch. DO on the other hand is visually dissimilar and requires networks specially trained on it.

Among the methods we observe no clear winner, though



(a) Input (b) Annotation (c) Mask-R-CNN (h)/(f) (d) Multicut (e) INBD (ours)

Figure 7. Qualitative comparison and examples of typical mistakes made by the compared methods

| Method | Training set | Test set | mAR↑ | ARAND↓ |
|---|---|---|---|---|
| INBD | EH | VM | .588 | .194 |
| Multicut | EH | VM | .580 | .166 |
| INBD | VM | EH | .472 | .262 |
| Multicut | VM | EH | .393 | .287 |
| INBD | EH | DO | .106 | .561 |
| Multicut | EH | DO | .116 | .552 |
| INBD | DO | EH | .219 | .435 |
| Multicut | DO | EH | .169 | .478 |

Table 4. Cross species ring detection performance

INBD is scoring on average slightly better. The results show that more research needs to be done into this direction.

### 5.4. Qualitative Results

Figure 7 shows typical mistakes caused by our method as well as the compared top-down and bottom-up procedures.

INBD tends to skip boundaries and this mistake often gets propagated onto the following rings since it is an iterative procedure. However, thanks to its iterative training procedure and boundary augmentations it can still recover from this.

As expected, the detector-based Mask-R-CNN struggles with the large overlap and fails to detect many rings, and the ones that get detected are very inaccurate. Bottom-up methods such as Multicut are prone to merging rings where boundaries are difficult to recognize and to splitting them on false positive boundary detections.

More qualitative results can be found in the supplement.

## 6. Concluding Remarks

Our dataset contains only images for which annotators were confident that they are annotated correctly. In real-world ecological studies, shrub samples, especially those from harsh climatic conditions, often contain many irregularities in their anatomical structures and may be extremely difficult to fully annotate, even for experts. In addition, fully annotating images with a large number of rings is very time consuming and costly. Therefore, future work could focus on weakly supervised training from partially annotated images and on developing methods that provide a confidence estimate for each detected ring or parts of it.

Moreover, as cross section images can vary widely depending on a variety of factors such as plant species, climatic conditions or sample preparation it is not unlikely that a single method trained on a single dataset will not suffice to cover all scenarios. Further research could be performed on cross-species training for better out-of-distribution generalization.

## ACKNOWLEDGEMENTS

This work has been supported by the European Social Fund (ESF) and the Ministry of Education, Science and Culture of Mecklenburg-Vorpommern, Germany under the project "DigIT!" (ESF/14-BM-A55-0015/19).

AAR was funded by a Postdoctoral Research Fellowship from the Alexander von Humboldt Foundation (Germany) and a Juan de la Cierva-Incorporación Grant by the Government of Spain.

## References


[1] Ignacio Arganda-Carreras, Srinivas C Turaga, Daniel R Berger, Dan Cireşan, Alessandro Giusti, Luca M Gambardella, Jürgen Schmidhuber, Dmitry Laptev, Sarvesh Dwivedi, Joachim M Buhmann, et al. Crowdsourcing the creation of image segmentation algorithms for connectomics. *Frontiers in neuroanatomy*, page 142, 2015. 6

[2] Alberto Bailoni, Constantin Pape, Nathan Hütsch, Steffen Wolf, Thorsten Beier, Anna Kreshuk, and Fred A. Hamprecht. Gasp, a generalized framework for agglomerative clustering of signed graphs and its application to instance segmentation. *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 11635–11645, 2022. 3, 6

[3] Sylvain Christin, Éric Hervet, and Nicolas Lecomte. Applications for deep learning in ecology. *Methods in Ecology and Evolution*, 10(10):1632–1644, 2019. 3

[4] Edward R. Cook. Methods of dendrochronology. 1990. 1

[5] Anna Fabijańska and Małgorzata Danek. Deepdendro–a tree rings detector based on a deep convolutional neural network. *Computers and electronics in agriculture*, 150:353–363, 2018. 3

[6] Jan Funke, Stephan Saalfeld, Davi Bock, Srini Turaga, and Eric Perlman. Cremi challenge. https://cremi.org, 2016. Accessed: 2022-10-20. 2

[7] Ángel M. García-Pedrero, Ana I. García-Cervigón, José Miguel Olano, Miguel García-Hidalgo, Mario Lillo-Saavedra, Consuelo Gonzalo-Martín, Cristina Caetano, and Saul Calderon-Ramirez. Convolutional neural networks for segmenting xylem vessels in stained cross-sectional images. *Neural Computing and Applications*, pages 1 – 13, 2019. 3

[8] Kaiming He, Georgia Gkioxari, Piotr Dollar, and Ross Girshick. Mask r-cnn. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, Oct 2017. 3, 6

[9] Jan Hosang, Rodrigo Benenson, Piotr Dollár, and Bernt Schiele. What makes for effective detection proposals? *IEEE transactions on pattern analysis and machine intelligence*, 38(4):814–830, 2015. 6

[10] Jörg Hendrik Kappes, Markus Speth, Björn Andres, Gerhard Reinelt, and Christoph Schnörr. Globally optimal image partitioning by multicuts. In *International Workshop on Energy Minimization Methods in Computer Vision and Pattern Recognition*, pages 31–44. Springer, 2011. 3, 6

[11] Jonathan GA Lageard. *Dendrochronology*, pages 180–197. Springer, 2016. 1

[12] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *European conference on computer vision*, pages 740–755. Springer, 2014. 2

[13] Zichen Liu, Jun Hao Liew, Xiangyu Chen, and Jiashi Feng. Dance: A deep attentive contour model for efficient instance segmentation. In *Proceedings of the IEEE/CVF winter conference on applications of computer vision*, pages 345–354, 2021. 3

[14] Ilya Loshchilov and Frank Hutter. Sgdr: Stochastic gradient descent with warm restarts. *arXiv: Learning*, 2017. 6

[15] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *ICLR*, 2019. 6

[16] Fausto Milletari, Nassir Navab, and Seyed-Ahmad Ahmadi. V-net: Fully convolutional neural networks for volumetric medical image segmentation. *2016 Fourth International Conference on 3D Vision (3DV)*, pages 565–571, 2016. 3

[17] Sida Peng, Wen Jiang, Huaijin Pi, Xiuli Li, Hujun Bao, and Xiaowei Zhou. Deep snake for real-time instance segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8533–8542, 2020. 3, 4, 6

[18] Miroslav Poláček, Alexis Arizpe, Patrick Hüther, Lisa Weidlich, Sonja Steindl, and Kelly Swarts. Automation of tree-ring detection and measurements using deep learning. *bioRxiv*, 2022. 3

[19] Giulia Resente, Alexander Gillert, Mario Trouillier, Alba Anadon-Rosell, Richard L Peters, Georg von Arx, Uwe von Lukas, and Martin Wilmking. Mask, train, repeat! artificial intelligence for quantitative wood anatomy. *Frontiers in plant science*, page 2526, 2021. 3

[20] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015*, pages 234–241, Cham, 2015. Springer International Publishing. 3

[21] Georg von Arx and Marco Carrer. Roxas – a new tool to build centuries-long tracheid-lumen chronologies in conifers. *Dendrochronologia*, 32:290–293, 2014. 3

[22] Georg von Arx and Hansjoerg Dietz. Automated image analysis of annual rings in the roots of perennial forbs. *International Journal of Plant Sciences*, 166:723 – 732, 2005. 3

[23] Stef Weijers, Rob A. Broekman, and Jelte Rozema. Dendrochronology in the high arctic: July air temperatures reconstructed from annual shoot length growth of the circumarctic dwarf shrub cassiope tetragona. *Quaternary Science Reviews*, 29:3831–3842, 2010. 1

[24] Martin Wilmking, Allan Buras, Jiri Lehejcek, Jelena Lange, Rohan Shetti, and Ernst van der Maaten. Influence of larval outbreaks on the climate reconstruction potential of an arctic shrub. *Dendrochronologia*, 2018. 1

[25] Adrian Wolny, Lorenzo Cerrone, Athul Vijayan, Rachele Tofanelli, Amaya Vilches Barro, Marion Louveaux, Christian Wenzl, Sören Strauss, David Wilson-Sánchez, Rena Lymbouridou, et al. Accurate and versatile 3d segmentation of plant tissues at cellular resolution. *Elife*, 9:e57613, 2020. 6, 7

[26] Enze Xie, Pei Sun, Xiaoge Song, Wenhai Wang, Xuebo Liu, Ding Liang, Chunhua Shen, and Ping Luo. Polarmask: Single shot instance segmentation with polar representation. *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 12190–12199, 2020. 3

[27] Xingyi Zhou, Dequan Wang, and Philipp Krähenbühl. Objects as points. *ArXiv*, abs/1904.07850, 2019. 7

# Supplementary Materials for
# Iterative Next Boundary Detection for Instance Segmentation of Tree Rings in Microscopy Images of Shrub Cross Sections

## 1. Influence of Hyperparameters

We additionally evaluate the role of the hyperparameters for INBD and Multicut. Important hyperparameters for INBD are the angular density $\alpha$ that controls the angular resolution $M$ and the number of iterations in one training epoch $n$. The results of our evaluations are presented in Figure 1.

The performance boost of iterative training diminishes and might even have a detrimental effect after 3 iterations. Contrary to our expectations and in contrast to other computer vision tasks like image classification, increasing the angular resolution has a negative effect on the detection performance, we attribute to a lower field of view.

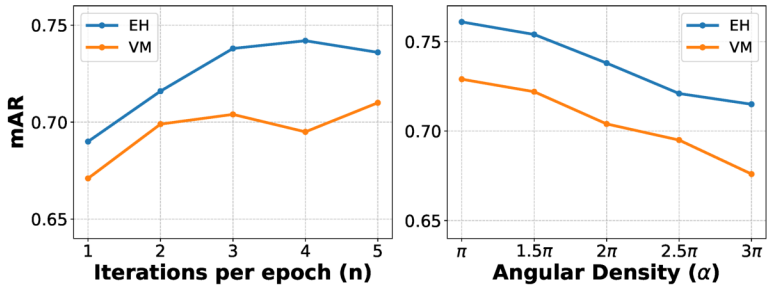



Figure 1. Influence of INBD hyperparameters on the detection performance.

For Multicut we have found that the smoothing factor for the watershed seed map (referred to as `sigma_seeds` in the PlantSeg source code) can be crucial and has to be tuned specifically to the plant species as shown in Figure 2. The results in the main paper show only the best values for each subset.

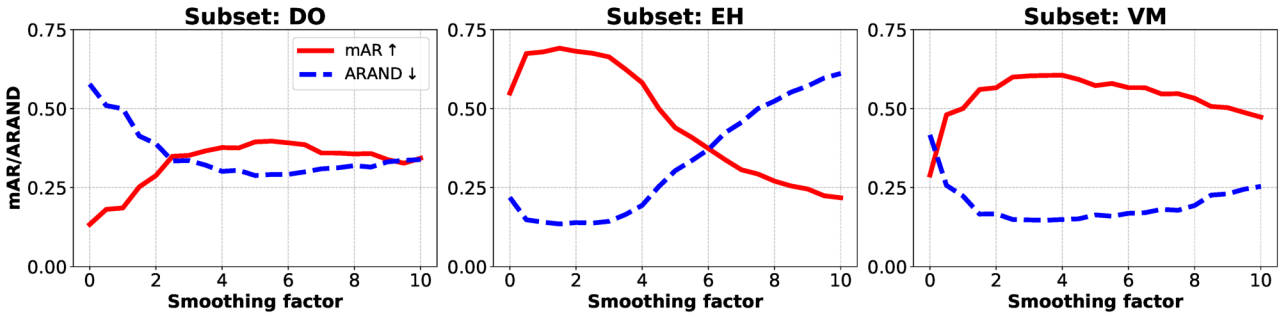
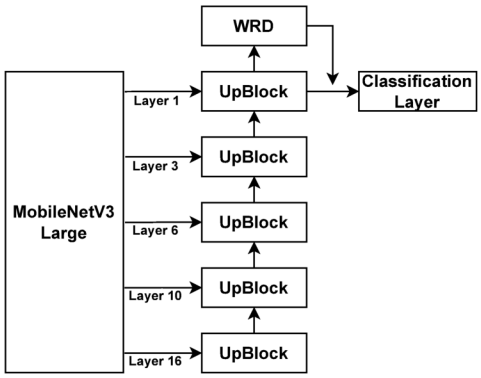



Figure 2. Influence of the Multicut watershed seed map smoothing factor on the detection performance.




(a) High level overview

| Operator | Kernel | # of Channels |
|---|---|---|
| `Upsample (x2)` | | x |
| `Concatenation` | | x+y |
| `Conv2D` | (1,1) | y |
| `CirularConv2D` | (3,3) | y |
| `InstanceNorm` | | |
| `ReLU` | | |

(b) INBD network UpBlock (data flow from top to bottom)

| Operator | Kernel/Stride |
|---|---|
| `MaxPool2D` | (2,1) |
| `Conv2D` | (1,1) |
| `InstanceNorm` | |
| `ReLU` | |
| `MaxPool2D` | (2,1) |
| `Conv2D` | (1,1) |
| `InstanceNorm` | |
| `ReLU` | |
| `Conv2D` | (1,1) |

(c) Wedging Ring Detection (WRD) Module

Figure 3. INBD network architecture

## 2. Network Architecture

For better reproducibility, we report more details on the used network architectures in Figure 3, however we note that our method is not dependent on this specific architecture, other segmentation networks should work as well. For all our experiments we have used a network architecture based on U-Net with a pretrained MobileNetV3-Large [1] backbone as implemented in torchvision (v0.11). This backbone was chosen as a compromise between prediction performance and speed: the high image resolution puts limits on the network size for both training and inference on an end user's device. Circular convolutions [2] are also used in the backbone and the circularity only applies to the angular axis, not to the radial one.

## 3. INBD with Cartesian Coordinates

INBD can in theory work with Cartesian coordinates as well, with the advantage that it is significantly easier to implement. We also evaluate how well this alternative performs. For this, we use the same architecture except with standard convolutions and without WRD. In each iteration step $i$ this network receives the outputs of the 3-class segmentation network $f$ as well as all previously detected rings and it is trained to segment the next ring $i+1$, akin to the our main method, but working on full images and not on polar grids. A basic result and comparison with polar coordinates can be found in Table 3 of the main paper. We observe that this alternative is prone to nonconvexities, an example is shown in Figure 5c. Polar coordinates on the other hand impose a prior on the shape, ensuring that it is coherent and (quasi-)convex.

We note that the image resolution has some influence on the overall detection performance: a high resolution allows for recognition of very indistinct boundaries but comes at the cost of a lower field of view which is needed for long-range dependencies and a consistent ring segmentation. An evaluation of the influence is shown in Figure 4. For our dataset, the optimal resolution lies around 768×768 pixels.

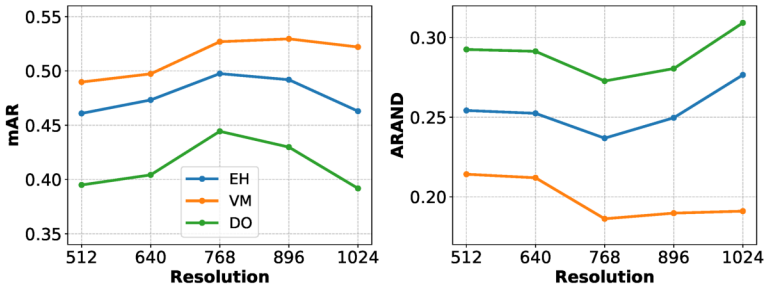
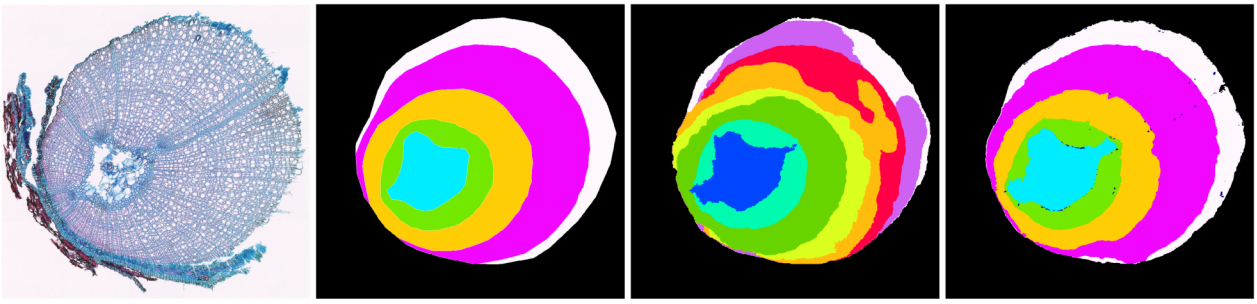
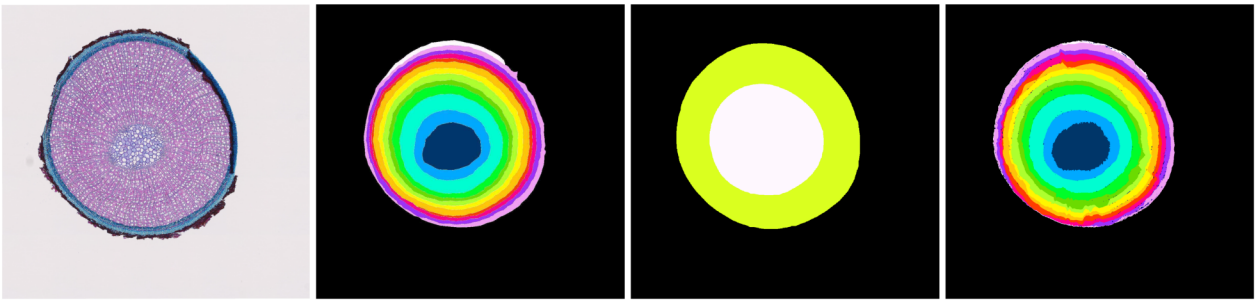



Figure 4. Influence of the image resolution on the performance of the Cartesian baseline



(a) Input (b) Annotation (c) INBD with Cartesian coordinates (d) INBD with polar coordinates

Figure 5. An image from the DO subset and comparison of INBD with Cartesian and polar coordinates. Note the typical nonconvex artifact on the orange ring in 5c.



(a) Input (b) Annotation (c) Deep Snake (d) INBD

Figure 6. An image from the VM subset and comparison of INBD with Deep Snake. Deep Snake inherently struggles detecting concentric objects. The result of INBD is better, correctly estimating the number of rings but still too inaccurate for further processing.

## 4. Additional Images

A comparison with Deep Snake can be seen in Figure 6. More failure cases are shown in Figure 7 which shows the need for more research into this direction. For better understanding of the application background, Figure 8 shows collected branch samples and the landscape where they were collected.

## References


[1] Andrew Howard, Mark Sandler, Grace Chu, Liang-Chieh Chen, Bo Chen, Mingxing Tan, Weijun Wang, Yukun Zhu, Ruoming Pang, Vijay Vasudevan, et al. Searching for mobilenetv3. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 1314–1324, 2019. 2

[2] Sida Peng, Wen Jiang, Huaijin Pi, Xiuli Li, Hujun Bao, and Xiaowei Zhou. Deep snake for real-time instance segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8533–8542, 2020. 2

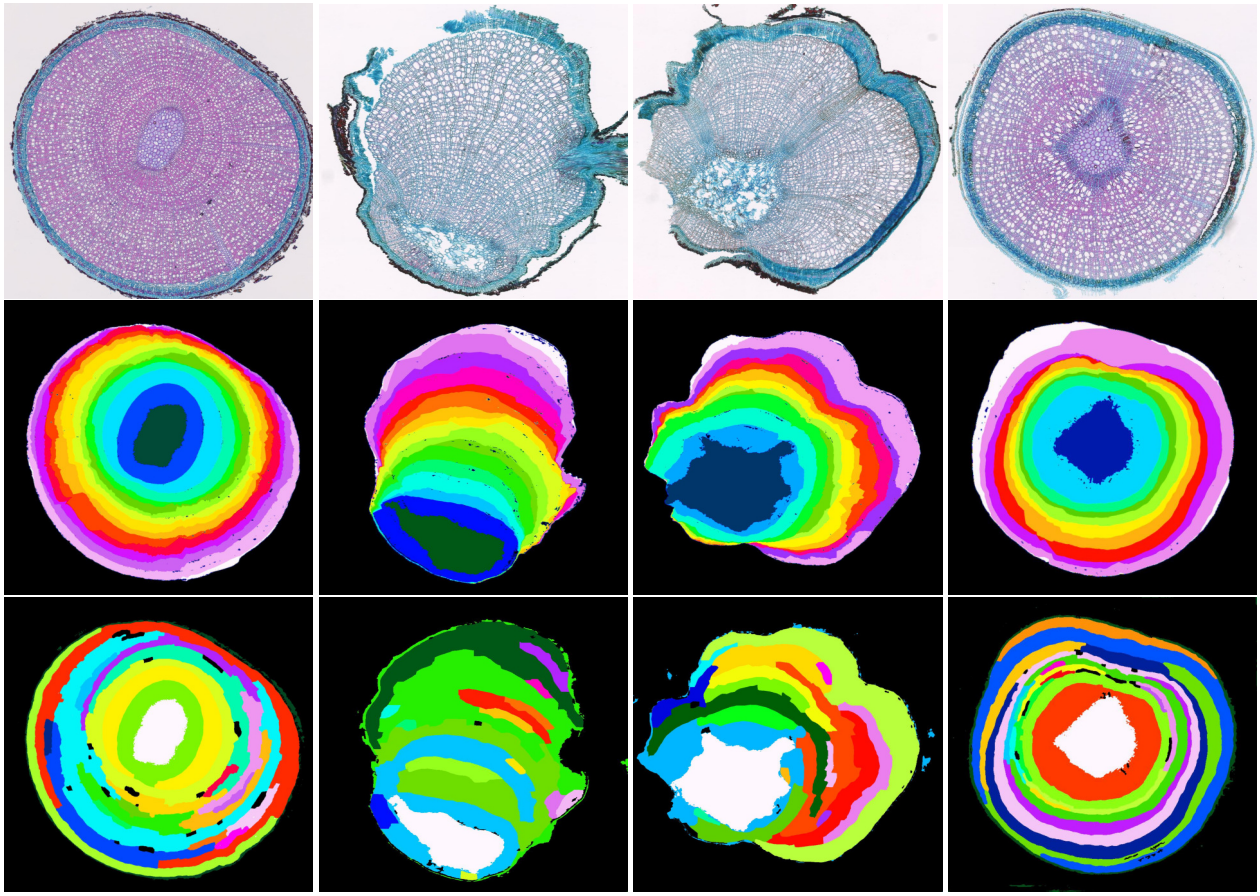


Figure 7. Example images on which none of the compared methods (center: INBD, bottom: Multicut) perform sufficiently well. These images are not in our published dataset because annotators were also not able to fully annotate them.

Figure 8. Images of branch samples (Dryas octopetala) from our dataset and the landscapes where samples were collected.

# 7. Tracking Growth and Decay of Plant Roots in Minirhizotron Images

| | | |
|---|---|---|
| Title | Tracking Growth and Decay of Plant Roots in Minirhizotron Images | |
| Authors | Alexander Gillert, Bo Peters, Uwe Freiherr von Lukas, Jürgen Kreyling, Gesche Blume-Werry | |
| Publication Venue | IEEE/CVF Winter Conference on Applications of Computer Vision (WACV) 2023, Waikoloa, USA | |
| Status | Published | |
| DOI | 10.1109/WACV56688.2023.00369 | |
| Venue Ratings as of 2023-03-05 | CORE2021[1] | A |
| | Research.com[2] | 9.70 (Impact Score)<br>7th among all computer vision conferences |
| | Google Scholar[3] | 76 (h5-Index)<br>9th among all computer vision venues |

[1] http://portal.core.edu.au/conf-ranks/763/

[2] https://web.archive.org/web/20230305125228/https://research.com/conference-rankings/computer-science/computer-vision

[3] https://web.archive.org/web/20230305125948/https://scholar.google.com/citations?view_op=top_venues&hl=en&vq=eng_computervisionpatternrecognition

# Tracking Growth and Decay of Plant Roots in Minirhizotron Images

Alexander Gillert[1] Bo Peters[2]
Uwe Freiherr von Lukas[1,3] Jürgen Kreyling[2] Gesche Blume-Werry[2,4]

[1]Fraunhofer Institute for Computer Graphics Research IGD, Rostock
[2]Institute of Botany and Landscape Ecology, Greifswald University
[3]Institute for Visual & Analytic Computing, University of Rostock
[4]Department of Ecology and Environmental Science, Umeå University
{alexander.gillert, uwe.freiherr.von.lukas}@igd-r.fraunhofer.de

## Abstract

*Plant roots are difficult to monitor and study since they are hidden belowground. Minirhizotrons offer an in-situ monitoring solution but their widespread adoption is still limited by the capabilities of automatic analysis methods. These capabilities so far consist only of estimating a single number (total root length) per image.*

*We propose a method for a more fine-grained analysis which estimates the root turnover, i.e. the amount of root growth and decay between two minirhizotron images. It consists of a neural network that computes which roots are visible in both images and is trained in an unsupervised manner without additional annotations.*

*Our code is available as a part of an analysis tool with a user interface ready to be used by ecologists.* [1]

## 1. Introduction

Quantification and close monitoring of plant root growth is of essential importance for many scientific fields, as only the inclusion of such data allows for accurate modeling of many ecosystem processes [20, 12]. For instance, depending on ecosystem type, 30-95% of plant biomass is accumulated below ground in the form of roots [13]. Rhizodeposition (release of organic compounds from roots) and root litter is a primary pathway for the transfer of atmospheric carbon into the soil [21].

Observation of plant roots, especially in-situ in the field, is difficult. Traditional methods, such as harvesting and washing out of roots, are highly destructive and can provide only a snapshot measurement as the plant is killed in the process, making observation of a single plant over longer periods infeasible.

[1]https://github.com/alexander-g/Root-Tracking

Therefore, the so called minirhizotron-technique [7] has become the most important tool for the monitoring of plant roots. Minirhizotrons are transparent tubes that are installed into the soil underneath a plant, commonly at a 45° angle. After this initial intervention, root growth at the soil-tube interface can be recorded without further disturbance with the help of specialized cameras or scanners that are inserted into the tube.

The most often used metric in root research is total root length. However, this metric does not capture the actual amount of growth that occurred inbetween measurements. Consider an observation station at a remote location such that measurements can be taken only once per year. The root length would stay roughly constant over this time period, not revealing the actual amount of root growth dynamics, i.e. their turnover. This is a real problem in root research and the time span does not have to be taken to such extremes. Fine roots which are of particular importance for water absorption and thus for the growth of the whole plant often have a life span of only a few days to weeks.

Our contribution aims to break down the total root length metric into its components, growth and decay, by comparing two images acquired on different days. An example of this problem can be seen in figure 1. Note that the problem is not simply about finding a single translation vector or a homography for a pair of images. Instead, we want to find a displacement for every root that exists in both images since roots can move over time as they grow, thereby changing the distance to each other nonlinearly. Moreover, soil can move as well for example due to swelling and shrinking caused by shifts in soil water content. This is especially prevalent in highly organic soils [9]. This movement often accounts for only a few pixels, but this is enough to distort measurements. Additional difficulties arise from the fact that roots can change their appearance (e.g. turn from white to red

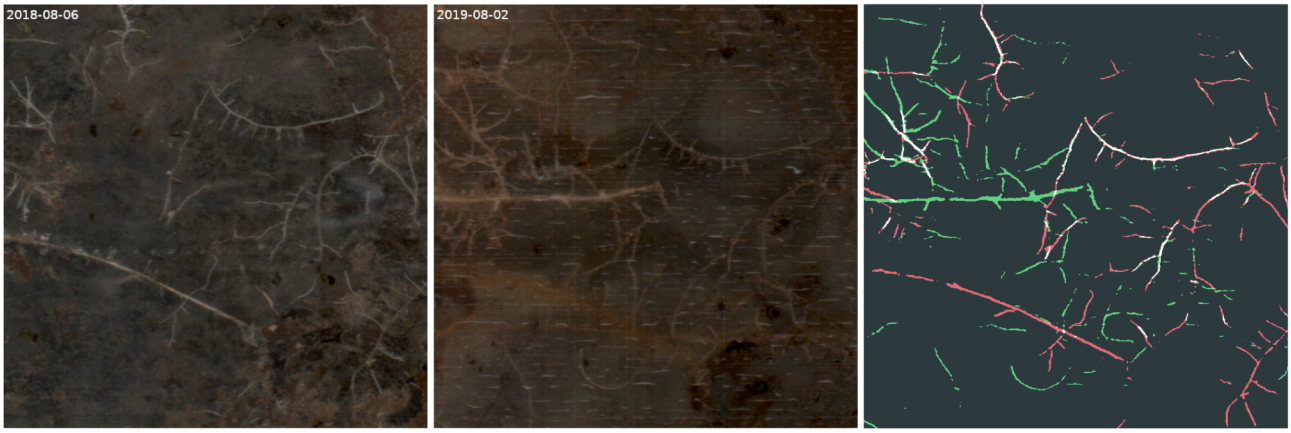



Figure 1: Problem Overview. Left and center: two minirhizotron images from the same experiment acquired approximately one year apart. Right: The result of our root tracking method. White pixels represent roots that are present in both images, red pixels are roots that are only visible in the first image (decayed or obstructed), green pixels are only visible in the second image (likely new growth).

color) or from changing environmental and image acquisition conditions. Examples of those issues are illustrated in figure 2.

We approach this task by finding correspondences in the image pairs with a neural network trained to compare image patches of roots. These correspondences are then used to warp the root segmentation map of the first image onto the second one. Importantly, our method is trained in an unsupervised manner with contrastive and self-supervised losses and does not require additional annotations, except for training the root detection network which are often already available anyway.

## 2. Related Work

Since manual tracing of roots in minirhizotron images is a slow and tedious task, several automatic analysis systems have been already introduced such as those in [16, 19] which are based on a segmentation neural network such as the popular U-Net[14] architecture. Further improvement of the analysis results has been achieved with methods like data augmentations [16] such as grid deformations. The authors of [23] used transfer learning by pretraining on different plant species and [25, 26] use weak supervision with multiple instance learning to reduce the amount of required annotations. In [2, 3] inpainting has been used against undersegmentation i.e. to correct gaps in segmentation results and in [6] oversegmentation of plants with dense root systems has been mitigated.

All of the works mentioned above only use a single image, i.e. one point in time, and mainly aim to improve the accuracy of the total root length measurements. To the best of our knowledge, no research has been published on the analysis of root growth from comparison of time series of minirhizotron images.

There exist several methods for other types of acquisition systems which take the temporal component into account. ChronoRoot[5] for example combines CNNs with LSTMs. Yet, these methods are dealing with image data from highly controlled and standardized experiments, e.g. single plants grown in petri dishes on transparent agarized medium and a fixed camera system with high temporal resolution such as PhenomNet [24]. These methods are difficult to apply to uncontrolled real-world environments and over longer periods of time.

A well established method to find correspondeces in two or more images is to use local feature descriptors such as the scale invariant feature transform (SIFT)[11], which computes a 3D histogram of local oriented gradients around a keypoint. It is widely used for problems like image stitching or localization and mapping. We have tried out this method but have found its performance to be insufficient in our case. A comparison with our method can be found in the evaluation section.

Neural network based feature matching methods such as SuperGlue [15], LoFTR [17] or COTR [10] promise better performance, however contrary to our method, they require ground truth annotations which are expensive to obtain with our images. Moreover, LoFTR has insufficient precision for fine roots as it only matches $8\times8$ patches.

A somewhat similar problem is *deformable image registration* from the medical domain, with methods such as VoxelMorph [1]. The goal here is to find a dense correspondence field that aligns two images. Specifically VoxelMorph can be trained in an unsupervised manner by min-

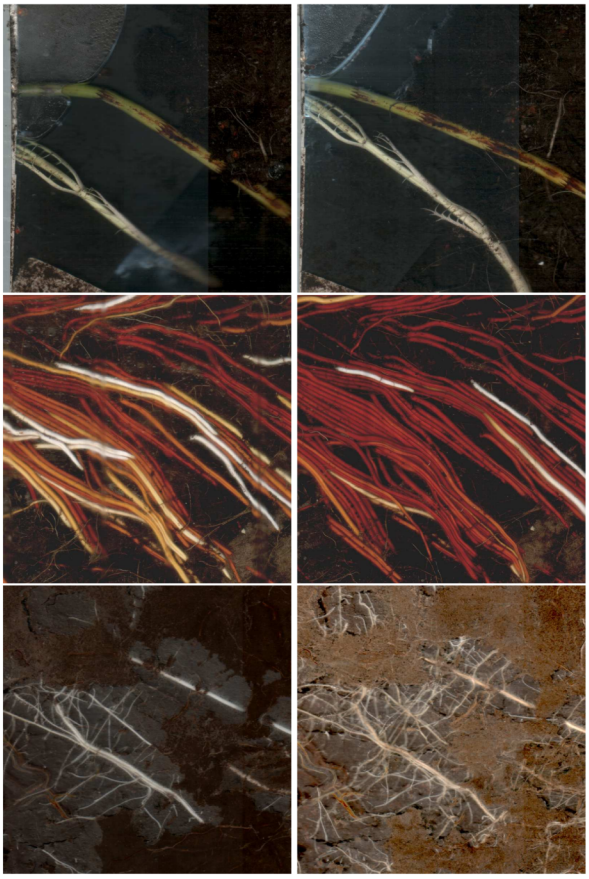



Figure 2: Some of the challenges encountered in this task. (a) Roots and soil can move over time, hence finding a homography is insufficient. (b) Roots can change their appearance, for example here their color. (c) Soil can change appearance, for example due to different moisture levels.

imizing the difference in RGB or grayscale values of the two input images. However, this task is not identical to our problem because in our case, roots that are present in one image are often missing in the second one (newly grown or decayed roots). Moreover, these techniques often make use of templates, i.e. a kind of prototypical representation of organs like brain, lungs etc, which is not applicable to our problem. We have evaluated the original VoxelMorph source code nevertheless, but have encountered some issues with it as explained in the evaluation section.

# 3. Method

## 3.1. Overview

Given two minirhizotron images $x0$ and $x1$ from the same experiment where $x0$ is the chronologically earlier one, we want to compute a dense displacement field that maps every pixel of $x0$ onto the corresponding location in $x1$ with a particular focus on correctness at the location of the roots.

We assume a pre-trained semantic segmentation network that can classify each pixel as "root" or "not-root". Such systems have been presented in [16, 19]. Specifically ,we use an architecture based on U-Net [14] with a pretrained backbone, but the method does not depend on this choice, other segmentation networks would work as well. In the following, this network is denoted with $f$ with $f(x)_i = 1$ being a detected root at pixel coordinate $i$ in image $x$.

The core of our root tracking system consists of a second neural network $g$ that is trained to learn the similarity of two image patches containing roots. We use the same architecture as for the root segmentation network except that we remove the last layer so that it returns the $c$-dimensional feature embeddings for each pixel ($c = 32$).

To compute how similar two locations in two images are, we use RoIAlign [8] to extract a (d×d×c)-sized descriptor with a box size of (b×b) of the output $g(x)$. Where not otherwise mentioned we use $b = 64$, $d = b/4$. $g(x)_i$ denotes an extracted descriptor from image $x$ at coordinate $i$. We then normalize the descriptors along the channel dimension and compute the cosine similarity.

## 3.2. Training

Our system is trained in two stages, neither of which requires additional annotations. In both stages, we use the weights of the semantic segmentation network $f$ as the initialization parameters and train with the SGD optimizer with a learning rate of $0.01$ and a momentum value of $0.9$ for 10 epochs where not otherwise mentioned. We have found the network to overfit quite easily as analyzed in the supplementary material.

### 3.2.1 Training Stage 1: Contrastive Learning

We want the neural network $g$ to return embeddings with a high similarity for descriptors extracted at the same root and low similarity for descriptors extracted at different roots. Since we do not know which roots from two images correspond to each other, we first train on descriptors extracted at the same location in the same image. To avoid trivial solutions, we utilize augmentations, as commonly used in contrastive learning methods [4].

Contrary to those generic contrastive learning methods, we are limited in the types of augmentations we can use for positive or negative views. For positive views we only use pixel-wise image transforms like color, contrast and brightness jitter to simulate different acquisition and environment conditions. For negative views, rotations and flipping operations are used. The idea here is that although roots can move, they rarely change their shape in a local area. This

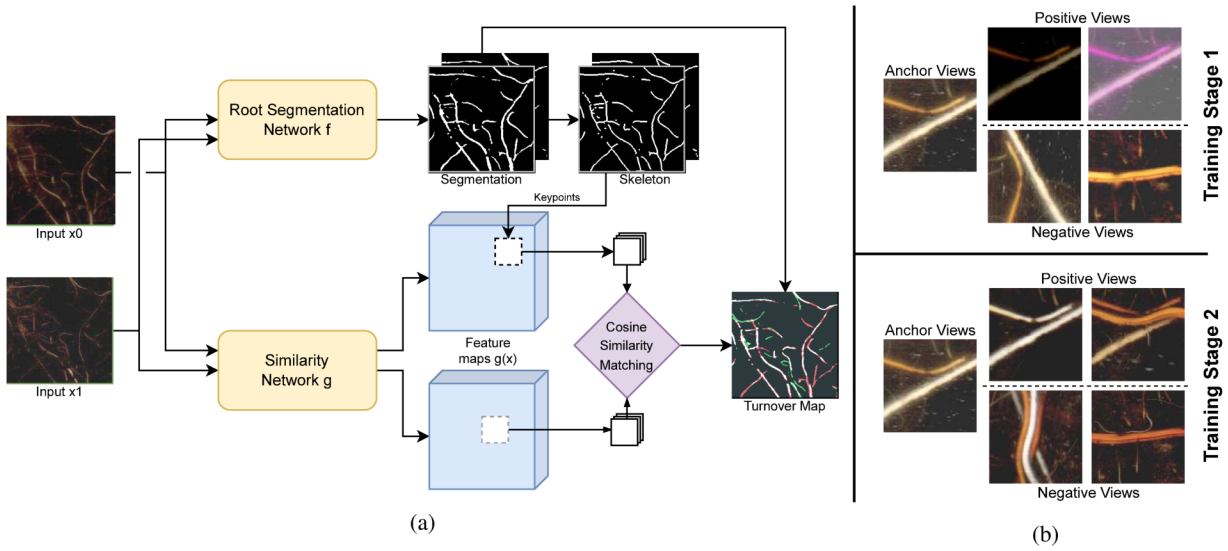



Figure 3: (a) Schematic overview of our root tracking system. (b) In the first training stage we train on augmented views of the same image. For positive views only pixel-wise augmentations like brightness and color jitter are used. Negative views consist of rotatations or flips of the same location and other locations in the same image. In the second training stage we use different images.

way, the network is trained to compare by shape rather than color or texture. Descriptors that were extracted at different locations serve as additional negative views. Example images of the augmented views are shown in figure 3b.

The losses for this stage are defined as:

$$\mathcal{L}_{pos}^{stage1} = \frac{1}{|I|} \sum_{i \in I} -log\ g(x)_i\ g(\phi_{pos}(x))_i \tag{1}$$

$$\mathcal{L}_{neg}^{stage1} = \frac{1}{|I|^2} \sum_{i \in I} \sum_{j \in I} -log\ 1 - g(x)_i\ g(\phi_{neg}(x))_j \tag{2}$$

where $I$ is a random subset of the root pixel locations $\{i \mid f(x)_i = 1\}$ while ensuring that each element of this subset has a minimum distance to others and $\phi_{pos}$, $\phi_{neg}$ are the positive and negative augmentation transforms respectively. In images where no roots could be found, completely random points are used. The final loss is then defined as $\mathcal{L}^{stage1} = \mathcal{L}_{pos}^{stage1} + \mathcal{L}_{neg}^{stage1}$.

#### 3.2.2 Training Stage 2: Self-supervision

The network above is already able to recognize same roots from different images quite well but it struggles with image pairs where the environment has changed considerably. A second training stage on different images is needed.

For this, we use the stage 1 model to find a set of correspondences $M$ in different images. These correspondences are then used instead of the augmented views of the same image. The positive loss is modified to:

$$\mathcal{L}_{pos}^{stage2} = \frac{1}{|M|} \sum_{i,j \in M} -log\ g(x_0)_i\ g(x_1)_j \tag{3}$$

with $i \in I$ a keypoint location in image $x0$ and $j \in J$ in image $x1$. The other losses are modified accordingly.

### 3.3. Inference

#### 3.3.1 Root Matching

We are only interested in matching roots with each other, therefore we directly use the coordinates where $f(x) = 1$ as keypoints. To reduce the number of points to match, we apply the skeletonization method [27] on the segmentation map. This effectively means that we only match the centers of roots to each other.

Similar to [11] we perform cross checking and a ratio test to reject descriptors that have ambiguous matching partners, i.e. we ensure that the best match has a significantly higher cosine similarity to the second-best match. Since our keypoints are non-sparse, i.e. many have a distance of 1 to each other, the descriptors of neighboring keypoints are often the best and second-best match. Therefore, when searching for the second-best match we exclude keypoints within a certain euclidean distance from the best match. Expressed more formally, the computation of the ratio looks as

follows:

$$r_i = \frac{\max\limits_{j \in J} g(x_0)_i g(x_1)_j}{\max\limits_{k \in K} g(x_0)_i g(x_1)_k}$$
$$K = \{k \mid \text{dist}\Big(\arg\max_j \; g(x_0)_i g(x_1)_j,\; k\Big) > t\} \subseteq J \tag{4}$$

where dist is the euclidean distance of the keypoint locations and $t$ a minimum distance threshold which set set to $t = 64$.

As a final step in the keypoint matching procedure, outlier rejection is performed. Although roots can move over time, this movement is usually limited to small distances. Therefore, we filter out matches that deviate from the median displacement vector by a threshold.

We define an image pair as successfully matched if there are at least two matched keypoint pairs left that have a minimum distance of 64 pixels to each other.

#### 3.3.2 Alignment and Turnover Map

Given a set of matched root keypoints in both images we warp the root segmentation result $f(x_0)$ onto $f(x_1)$. To do this, we construct a dense deformation field by performing triangulation on the matched keypoints of image $x_0$. The new coordinates are then computed in each triangle via linear barycentric interpolation of the corresponding keypoints coordinates in $x_1$ [18]. Values outside of the matched keypoints' convex hull are extrapolated by adding additional points to the four corners of the image $x0$ with values found via nearest neighbor interpolation.

Lastly, we construct the final result which we term a (root) *turnover map* containing the three classes "same" (root is detected in both images), "decay" (root is detected only in the first image) and "growth" (root is detected only in the second image):

$$T \coloneqq \begin{cases} same, & warp(f(x_0)) = 1 \text{ and } f(x_1) = 1 \\ decay, & warp(f(x_0)) = 1 \text{ and } f(x_1) = 0 \\ growth, & warp(f(x_0)) = 0 \text{ and } f(x_1) = 1 \end{cases} \tag{5}$$

where $warp$ aligns the segmentation map $f(x_0)$ to $f(x_1)$ as described above.

More important for the end user is the total length of the newly grown or decayed roots. The skeletonization method [27] is commonly used to estimate this property for binary segmentation maps, however it cannot be used for multiclass ones. Applying it on a single class is not an option either as it would create artifacts at root borders, for example when a root gains width. Therefore, we redefine it for our case as combinations of the binary outputs and the turnover map $T$:

$$S(T) \coloneqq \begin{cases} same, & S(\quad f(x_1)\ ) = 1 \text{ and } T = same \\ decay, & S(warp(f(x_0))) = 1 \text{ and } T = decay \\ growth, & S(\quad f(x_1)\ ) = 1 \text{ and } T = growth \end{cases} \tag{6}$$

where $S(x)$ is the skeletonization method applied on a binary image $x$. The total lengths are then estimated with the sum of skeleton pixels of $S(T)$ over each class.

## 4. Experimental Setup

### 4.1. Datasets and Annotation

Our main dataset consists of 2550×2273px minirhizotron images acquired with a CI-600 In-Situ Root Imager (CID Bio-Science Inc.). The images stem from mesocosm (outdoor pot experiments under semi-controlled, roughly constant conditions) and field (outdoor and uncontrolled) experiments. Overall 854 unannotated images were used for the training which mostly contain roots of Carex rostrata, Mentha aquatica and Equisetum fluviatile plant species.

For the evaluation, we have annotated additional 62 image pairs. We have not considered image pairs in which even annotators were not able to find correspondences. The annotation consists of matched root keypoints that human annotators regarded to be the same root in both images and the corresponding turnover maps. Creating such an annotation from scratch is an extremely tedious and slow process, therefore annotators were tasked only to correct mistakes caused by the root matching algorithm that was presented in section 3, i.e. to add missing matches or to remove incorrect ones. Despite this simplification, a single image pair can take up to two hours to annotate due to the high image resolution and sometimes many fine roots.

The annotation was created with a custom user interface which was built specifically for this task. The user interface allows to add new matches either by clicking on corresponding locations in both images or by clicking and dragging within a single turnover map. This process is illustrated in figure 4.

To test how well our method adapts to other datasets we additionally use the Sunflower subset of the PRMI dataset [22] as a secondaray dataset. This dataset has an agricultural background for which root turnover is of lesser importance. For a fair evaluation all networks were retrained on this dataset only. It is used only where explicitly mentioned, otherwise the data refer to the main mesocosm and field dataset.

### 4.2. Evaluation Metrics

We use the following metrics for the evaluation:

- **Intersection over Union** (IoU) applied on the classes "same" and "growth" of the turnover map. Note that the class "decay" cannot be used because the annotation is focused on matching roots that are present in

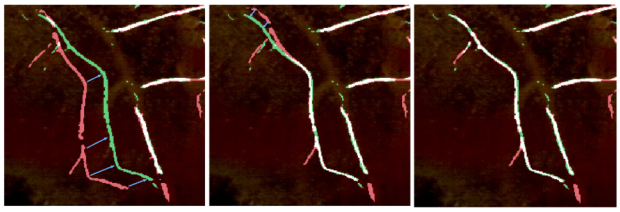


Figure 4: Our main annotation procedure. Users can iteratively correct mistakes in the turnover map, which we then use for the evaluation. Note that the branching decayed root (red) at the bottom changes its position in the process. The intersection with the original prediction is zero and the IoU metric would count this as incorrect although it actually isn't (the class was correctly predicted). This metric cannot be applied on the "decay" class for this reason.

both images. The location of decayed roots cannot be accurately annotated in our turnover map. This problem is also explained in figure 4. IoU is widely used in segmentation tasks but in our case this metric can be unreliable due to the mode of annotation, as dicussed in subsection 5.3.

- **Relative error** in the total number of "same", "growth" and "decay" **pixel counts** in the turnover map $T$. This can be interpreted as an estimate of total root biomass.
- **Relative error** in the total number of "same", "growth" and "decay" skeleton pixels of the skeletonized turnover map $S(T)$. This metric can be interpreted as an estimate of total root **length**.
- Matching **success rate** with the success condition as defined in 3.3.1

### 4.3. Compared Methods

Since there are no dedicated methods for tracking roots in minirhizotron images yet, we source similar methods from other domains as baselines for the comparative evaluation. We have considered the following alternatives and compare them here to our method:

- **SIFT** [11] serves as a minimal baseline to compute correspondences in an image pair. These are then fed into the pipeline described in subsections 3.3.1 and 3.3.2 to create a turnover map.
- **SIFT** is not aware of the root locations in the image and thus uses keypoints at arbitrary locations. To improve performance we add additional keypoints at the **skeleton pixels**. The scale parameter for these keypoints is estimated from the median of the original SIFT keypoints. Denoted with SIFT(sk).
- **VoxelMorph** [1] by itself struggles to align two minirhizotron images because the offsets can be very large. Therefore, we use it as a second step after applying the SIFT method. Specifically, we train it on the segmentation maps $warp(f(x0))$ warped with the correspondences found with SIFT. This is the best combination that we have tried. We have used the source code as published by the authors.
- **Feature embeddings of the root segmentation network** $f$ instead of $g$. The rationale here is that $f$ already knows what roots look like and thus might be enough to compare them. This method has the advantage that only one network has to be trained. Denoted with $f$ Emb.

As of our method, we compare the two different training stages:

- **Stage 1**: Trained via contrastive learning only on augmented views from the same image. This stage is described in subsection 3.2.1.
- **Stage 2**: Trained via self-supervision on different images. This is our main method and is described in subsection 3.2.2.

## 5. Results

Additional evaluations and full sized results can be found in the supplement.

### 5.1. Main Results

Our main evaluation results are presented in table 1. We observe a significantly better performance of our method on all evaluated metrics. Simply using the traditional SIFT method alone can lead to deviations of almost 50% in root length measurements. Adding additional keypoints at the root locations does help but it is still outcompeted by our method. VoxelMorph is able to align roots quite well to each other as long as they are present in both images but struggles with roots that are visible only in one of the images. As can be seen in figure 5 it often tries to extend roots from the first image into newly grown ones and even more often shrinks down decayed roots to a thin line, resulting in large errors in the "decay" class. Despite good length errors on the "same" and "growth" classes, this issue makes it less trustworthy and explainable than our method. Additionally, errors are more difficult to correct manually.

The results on the PRMI dataset (table 2) are similar. One notable difference is that the feature embeddings of the root segmentation network $f$ perform better than our stage 1 model. We attribute this to less environmental variance, so that pixel-wise augmentations have a smaller positive effect.

| Method | Mesocosms & Field ($n = 62$) | | | |
|---|---|---|---|---|
| | IoU↑ s/g | Counts↓ s/d/g | Lengths↓ s/d/g | Success Rate↑ |
| SIFT | .59/.62 | .30/.29/.38 | .35/.44/.45 | 77.4% |
| SIFT(sk) | .65/.67 | .22/.27/.25 | .28/.41/.30 | 88.7% |
| SIFT+VoxelMorph | .76/.64 | .13/.75/.18 | .10/.52/**.10** | 88.7% |
| $f$ Emb. | .69/.69 | .18/.23/.20 | .23/.33/.25 | 88.7% |
| Ours (Stage 1) | .75/.74 | .12/.20/.12 | .13/.19/.13 | **93.5**% |
| Ours (Stage 2) | **.83/.81** | **.08/.10/.09** | **.09/.12/.10** | **93.5**% |

Table 1: Main results. Bold font indicates best values. Counts and Lengths are relative error values. s/d/g stands for the turnover map classes "same", "decay" and "growth".

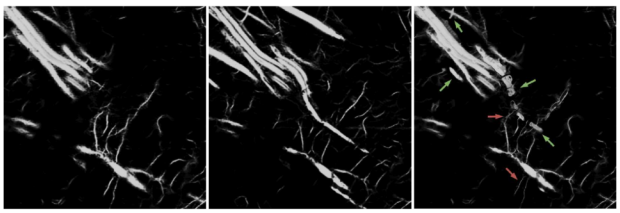


Figure 5: Typical failure case of VoxelMorph. From left to right: root segmentation maps $f(x0)$, $f(x1)$, VoxelMorph output, i.e. warped $f(x0)$. VoxelMorph is prone to hallucinating new roots (indicated by green arrows) or shrinking decayed ones (red arrows).

| Method | PRMI Sunflower ($n = 20$) | | | |
|---|---|---|---|---|
| | mIoU↑ | Counts↓ | Lengths↓ | Success↑ |
| SIFT | 0.51 | 0.82 | 1.17 | 80% |
| SIFT(sk) | 0.58 | 0.53 | 0.77 | **100**% |
| $f$ Emb. | 0.77 | 0.18 | 0.22 | 80% |
| Ours (Stage 1) | 0.72 | 0.28 | 0.38 | 90% |
| Ours (Stage 2) | **0.86** | **0.12** | **0.16** | 90% |

Table 2: Results on the PRMI dataset. Bold font indicates best values. Counts and Lengths are relative error values.

### 5.2. Time Dependence

Additionally, we compare how the performance depends on the acquisition time difference of the compared images. The longer the time span the larger the changes in roots and soil, thus in theory, the performance should go down. Figure 6 shows the same metrics evaluated on a subset of the data above, broken down into 5 time frames from one week to half a year. Specifically, image pairs from 4 mesocosm experiments were used, each pair starting with the same image.

As expected, the performance is best over shorter time periods, with no longer than 2 weeks being an optimum for our method. Contrary to our expectation, training on different images (i.e. stage 2) does not extend the optimum time span significantly. Instead, its main advantage is simply being able to match more keypoints as depicted in the additional figures in the supplementary material. Changing environmental conditions are still a challenge for it. Vanilla SIFT has even problems at detecting any correspondences at all over longer periods with matching success rate dropping down to zero after two months.

### 5.3. Annotation Bias Analysis

The annotation as used above, was created from the output of the stage 2 model as the starting point. As a result, this annotation might be biased towards this model because annotators might have skipped areas which they deemed good enough. Only a few pixels of shift in one direction might accumulate in a significant amount of bias especially for the IoU metric. An example of this type of bias is shown in figure 7.

To analyze the extent of this bias, we have re-annotated a random 10% of the same image pairs by correcting the output of the stage 1 model which we denote with $Ann_{S1}$. The same 10% of the previous annotation (with stage 2 as the starting point) is denoted as $Ann_{S2}$. We have re-evaluated the metrics for this new annotation, the results are shown in table 3.

| | Mean IoU↑ | | Mean Counts↓ | | Mean Lengths↓ | |
|---|---|---|---|---|---|---|
| | $Ann_{S1}$ | $Ann_{S2}$ | $Ann_{S1}$ | $Ann_{S2}$ | $Ann_{S1}$ | $Ann_{S2}$ |
| $Ann_{S1}$ | - | 0.845 | - | 0.016 | - | 0.014 |
| $Ann_{S2}$ | 0.845 | - | 0.016 | - | 0.014 | - |
| Stage 1 | **0.794** | 0.772 | 0.151 | 0.144 | 0.153 | 0.141 |
| Stage 2 | 0.793 | **0.831** | **0.099** | **0.093** | **0.110** | **0.097** |

Table 3: Re-evaluated metrics for different annotations. Bold font indicates best values for each annotation.

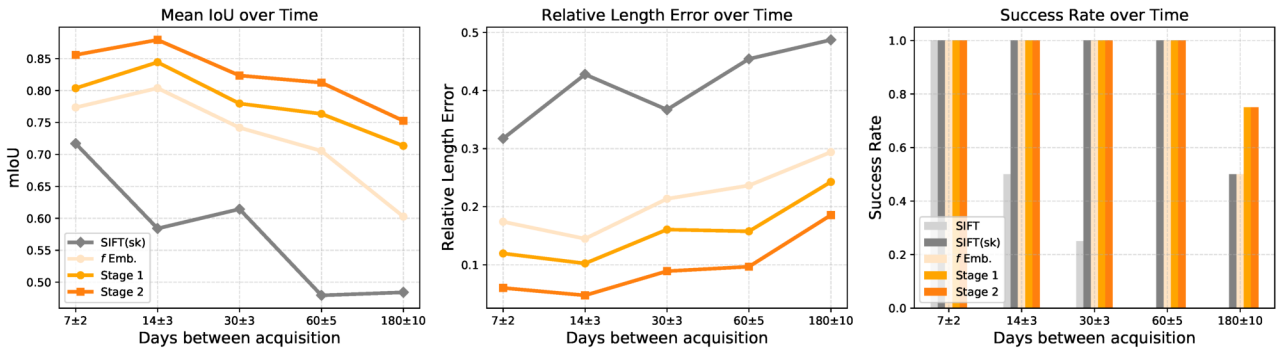



Figure 6: Influence of the acquisition time difference on the IoU, length error and success rate.

Figure 7: Typical source for bias in the annotation. Left: output of stage 1, right: stage 2. The stage 1 model did not match this root perfectly, yet both cases were left untouched by annotators thus favoring the corresponding model during evaluation. The shift amounts to only a few pixels but it accumulates along the length of the root.

As can be seen, a bias is certainly present since the performance gain of the stage 2 model is completely gone when comparing with the IoU metric on the $Ann_{S1}$ annotation. This bias is much less pronounced on the relative pixel count error and relative length error metrics on which the stage 2 model still performs significantly better. The same pattern can be observed when comparing both sets of annotations to each other. The rather low IoU value of $0.845$ indicates a high variation on a per-pixel level, but this goes down to less than $2\%$ for the image-level metrics.

We conclude that IoU is unreliable for our mode of annotation and focus should be put on the other metrics instead, which are of higher importance for the end user anyway.

## 6. Concluding Remarks

Our measurements of newly grown and decayed roots are only estimates. For one, large movements of roots still pose a challenge as illustrated in figure 8. Secondly, roots can get obstructed by soil which our method would count as decay. This is rather an inherent limitation of minirhizotron imagery. Future work might focus on improving the matching performance for longer time periods. VoxelMorph shows partially promising results but has drawbacks that need to be addressed. A combination with our work might be possible.

Our method is already in a usable state and we hope that it can enable new insights in the field of root research.

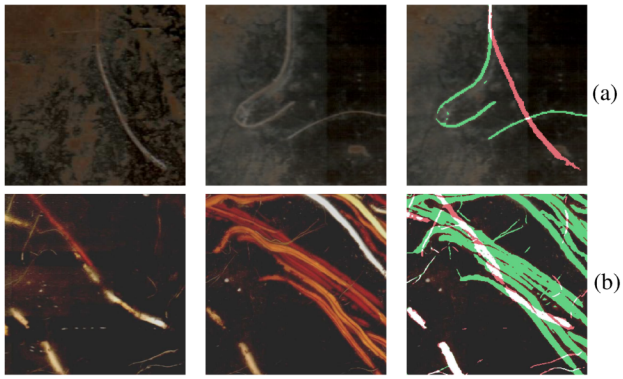



Figure 8: Failure cases of our method.
(a) Large movement of roots poses a problem since our descriptors are trained to learn the shape and are not rotation invariant. (b) A root hidden behind others can be classified as “same” in the turnover map due to its rather simple definition of overlapping segmentations.

## ACKNOWLEDGEMENTS

This work has been supported by the European Social Fund (ESF) and the Ministry of Education, Science and Culture of Mecklenburg-Vorpommern, Germany under the project ”DigIT!” (ESF/14-BM-A55-0015/19).

## References


[1] Guha Balakrishnan, Amy Zhao, Mert Rory Sabuncu, John V. Guttag, and Adrian V. Dalca. Voxelmorph: A learning framework for deformable medical image registration. *IEEE Transactions on Medical Imaging*, 38:1788–1800, 2019.

[2] Hao Chen, Mario Valerio Giuffrida, Sotirios A Tsaftaris, and Peter Doerner. Root gap correction with a deep inpainting model. In *BMVC*, page 325, 2018.

[3] Hao Chen, Mario Valerio Giuffrida, Peter Doerner, and Sotirios A Tsaftaris. Adversarial large-scale root gap inpainting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pages 0–0, 2019.

[4] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey E. Hinton. A simple framework for contrastive learning of visual representations. In *Proceedings of the 37th International Conference on Machine Learning, ICML 2020, 13-18 July 2020, Virtual Event*, volume 119 of *Proceedings of Machine Learning Research*, pages 1597–1607. PMLR, 2020.

[5] Nicols Gaggion, Federico Ariel, Vladimir Daric, ric Lambert, Simon Legendre, Thomas Roul, Alejandra Camoirano, Diego H Milone, Martin Crespi, Thomas Blein, and Enzo Ferrante. ChronoRoot: High-throughput phenotyping by deep segmentation networks reveals novel temporal parameters of plant root system architecture. *GigaScience*, 10(7), 07 2021.

[6] Alexander Gillert, Bo Peters, Uwe Freiherr von Lukas, and Jürgen Kreyling. Identification and measurement of individual roots in minirhizotron images of dense root systems. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV) Workshops*, pages 1323–1331, October 2021.

[7] AC Hansson and ELIEL Steen. Root growth of daily irrigated and fertilized barley investigated with ingrowth cores, soil cores and minirhizotrons. *Swedish Journal of Agricultural Research (Sweden)*, 1992.

[8] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 2961–2969, 2017.

[9] Colleen M Iversen, Meaghan T Murphy, Michael F Allen, Joanne Childs, David M Eissenstat, Erik A Lilleskov, Tytti M Sarjala, Victoria L Sloan, and Patrick F Sullivan. Advancing the use of minirhizotrons in wetlands. *Plant and Soil*, 352(1):23–39, 2012.

[10] Wei Jiang, Eduard Trulls, Jan Hosang, Andrea Tagliasacchi, and Kwang Moo Yi. Cotr: Correspondence transformer for matching across images. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 6207–6217, October 2021.

[11] David G Lowe. Distinctive image features from scale-invariant keypoints. *International journal of computer vision*, 60(2):91–110, 2004.

[12] M Luke McCormack, Elizabeth Crisfield, Brett Raczka, Frank Schnekenburger, David M Eissenstat, and Erica AH Smithwick. Sensitivity of four ecological models to adjustments in fine root turnover rate. *Ecological modelling*, 297:107–117, 2015.

[13] Karel Mokany, R John Raison, and Anatoly S Prokushkin. Critical analysis of root: shoot ratios in terrestrial biomes. *Global change biology*, 12(1):84–96, 2006.

[14] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *MICCAI*, 2015.

[15] Paul-Edouard Sarlin, Daniel DeTone, Tomasz Malisiewicz, and Andrew Rabinovich. Superglue: Learning feature matching with graph neural networks. *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4937–4946, 2020.

[16] Abraham George Smith, Jens Petersen, Raghavendra Selvan, and Camilla Ruø Rasmussen. Segmentation of roots in soil with u-net. *Plant Methods*, 16(1):1–15, 2020.

[17] Jiaming Sun, Zehong Shen, Yuang Wang, Hujun Bao, and Xiaowei Zhou. Loftr: Detector-free local feature matching with transformers. *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8918–8927, 2021.

[18] Pauli Virtanen, Ralf Gommers, Travis E. Oliphant, Matt Haberland, Tyler Reddy, David Cournapeau, Evgeni Burovski, Pearu Peterson, Warren Weckesser, Jonathan Bright, Stéfan J. van der Walt, Matthew Brett, Joshua Wilson, K. Jarrod Millman, Nikolay Mayorov, Andrew R. J. Nelson, Eric Jones, Robert Kern, Eric Larson, C J Carey, İlhan Polat, Yu Feng, Eric W. Moore, Jake VanderPlas, Denis Laxalde, Josef Perktold, Robert Cimrman, Ian Henriksen, E. A. Quintero, Charles R. Harris, Anne M. Archibald, Antônio H. Ribeiro, Fabian Pedregosa, Paul van Mulbregt, and SciPy 1.0 Contributors. SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python. *Nature Methods*, 17:261–272, 2020. implemented as scipy.interpolate.LinearNDInterpolator.

[19] Tao Wang, Mina Rostamza, Zhihang Song, Liangju Wang, G McNickle, Anjali S Iyer-Pascuzzi, Zhengjun Qiu, and Jian Jin. Segroot: a high throughput segmentation method for root image analysis. *Computers and Electronics in Agriculture*, 162:845–854, 2019.

[20] Jeffrey M Warren, Paul J Hanson, Colleen M Iversen, Jitendra Kumar, Anthony P Walker, and Stan D Wullschleger. Root structural and functional dynamics in terrestrial biosphere models–evaluation and recommendations. *New Phytologist*, 205(1):59–78, 2015.

[21] JM Whipps and JM Lynch. Substrate flow and utilization in the rhizosphere of cereals. *New phytologist*, 95(4):605–623, 1983.

[22] Weihuang Xu, Guohao Yu, Yiming Cui, Romain Gloaguen, Alina Zare, Jason Bonnette, Joel Reyes-Cabrera, Ashish Rajurkar, Diane Rowland, Roser Matamala, et al. Prmi: A dataset of minirhizotron images for diverse plant root study. *AI for Agriculture and Food Systems (AIAFS) Workshops at the AAAI conference on artificial intelligence*, 2022.

[23] Weihuang Xu, Guohao Yu, Alina Zare, Brendan Zurweller, Diane L. Rowland, Joel Reyes-Cabrera, Felix B. Fritschi, Roser Matamala, and Thomas E. Juenger. Overcoming small

minirhizotron datasets using transfer learning. *Computers and Electronics in Agriculture*, 175:105466, 2020.

[24] Robail Yasrab, Michael P Pound, Andrew P French, and Tony P Pridmore. Phenomnet: bridging phenotype-genotype gap: a cnn-lstm based automatic plant root anatomization system. *bioRxiv*, 2020.

[25] Guohao Yu, Alina Zare, Hudanyun Sheng, Roser Matamala, Joel Reyes-Cabrera, Felix B Fritschi, and Thomas E Juenger. Root identification in minirhizotron imagery with multiple instance learning. *Machine Vision and Applications*, 31(6):1–13, 2020.

[26] G. Yu, A. Zare, Weihuang Xu, R. Matamala, J. Reyes-Cabrera, F. Fritschi, and T. Juenger. Weakly supervised minirhizotron image segmentation with mil-cam. In *ECCV Workshops*, 2020.

[27] T. Y. Zhang and C. Y. Suen. A fast parallel algorithm for thinning digital patterns. *Commun. ACM*, 27(3):236239, Mar. 1984.

# Tracking Growth and Decay of Plant Roots in Minirhizotron Images (Supplementary Material)

Alexander Gillert[1] Bo Peters[2]
Uwe Freiherr von Lukas[1,3] Jürgen Kreyling[2] Gesche Blume-Werry[2,4]

[1]Fraunhofer Institute for Computer Graphics Research IGD, Rostock
[2]Institute of Botany and Landscape Ecology, Greifswald University
[3]Institute for Visual & Analytic Computing, University of Rostock
[4]Department of Ecology and Environmental Science, Umeå University
{alexander.gillert, uwe.freiherr.von.lukas}@igd-r.fraunhofer.de

## 1. Varying Hyperparameters

In this section we analyze how the method hyperparameters affect the overall performance of our method. As figure **??** shows our network ist highly susceptible to overfitting. Particularly the stage 1 model which is trained on the same images sees a rapid performance decline, whereas the stage 2 model is slightly more robust thanks to the training on pairs of different images but declines as well after an optimum at around 10 epochs.

Another imporant hyperparameter is the box size $b$ of the extracted descriptors which roughly corresponds to its field of view. It should not be set too low with ca. 32 pixels being the optimum for stage 2.

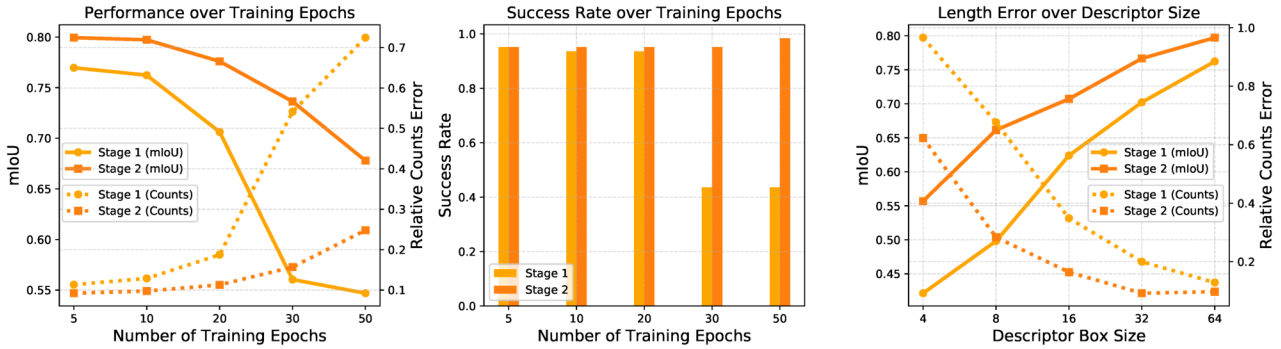



Figure 1: Influence of hyperparameters on the mean IoU, relative pixel counts error and success rate metrics.

## 2. Regression Analysis

Regression analysis is more common in ecology to describe the predicting power of a model with the coefficient of determination or $r^2$-value giving the overall fitness. Figure **??** shows regression plots of our stage 2 model. We get excellent $r^2$-values of over 0.99 for all three turnover map classes.

## 3. Mesocosms vs Field Experiments

For a more fine-grained analysis we have also evaluated only on images from mesocosms or only field experiments. The rationale here is that field experiments are uncontrolled and conditions may vary greatly over time and thus should be more

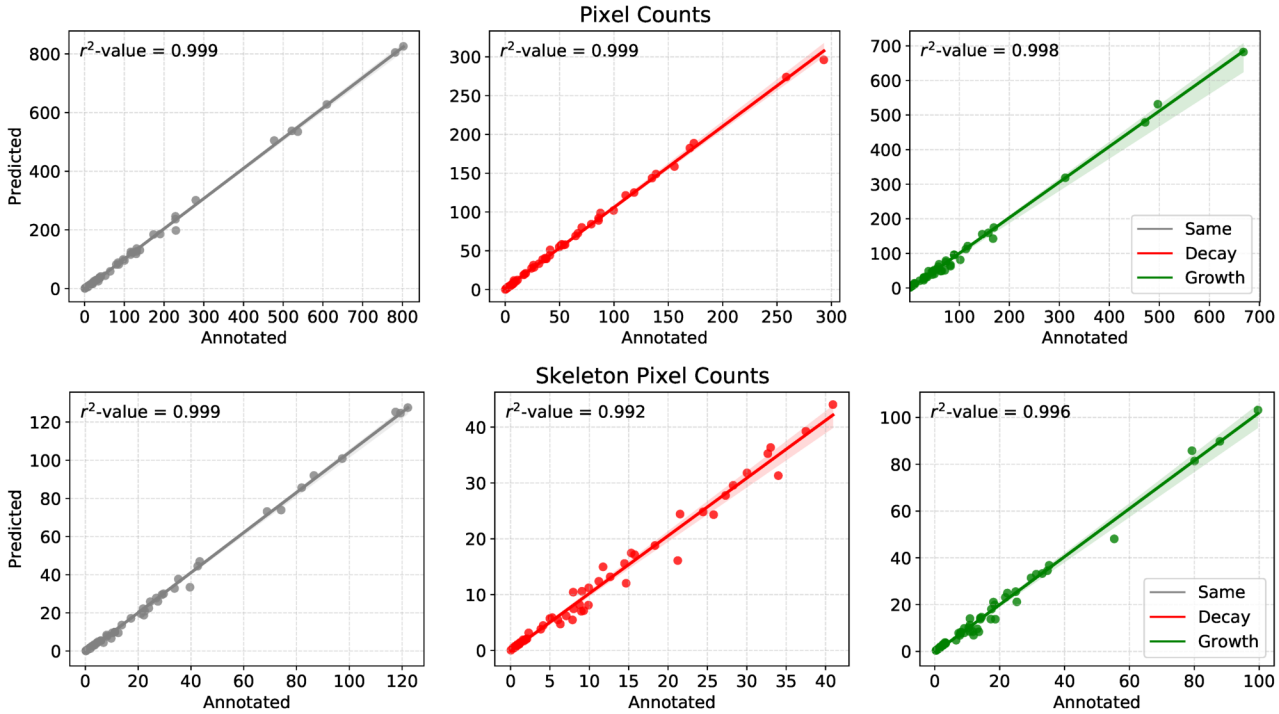



Figure 2: Regression plots with 99% confidence intervals of our stage 2 model. Units are pixels (thousands).

| Method | Mesocosms ($n = 40$) IoU↑ s/g | Counts↓ s/d/g | Lengths↓ s/d/g | Success Rate↑ |
|---|---|---|---|---|
| SIFT | .59/.59 | .30/.25/.42 | .36/.36/.49 | 75.0% |
| SIFT(sk) | .66/.63 | .22/.28/.29 | .28/.38/.34 | 82.5% |
| SIFT+VoxelMorph | .76/.61 | .13/.75/.18 | .11/.53/**.10** | 82.5% |
| $f$ Emb. | .70/.67 | .17/.22/.20 | .22/.27/.24 | 87.5% |
| Ours (Stage 1) | .75/.73 | .11/.18/.12 | .11/.17/.13 | **92.5%** |
| Ours (Stage 2) | **.84/.78** | **.07/.09/.09** | **.08/.11/.10** | 90.0% |

| Method | Field ($n = 22$) IoU↑ s/g | Counts↓ s/d/g | Lengths↓ s/d/g | Success Rate↑ |
|---|---|---|---|---|
| SIFT | .60/.67 | .29/.33/.32 | .33/.57/.38 | 81.8% |
| SIFT(sk) | .64/.73 | .23/.26/.19 | .27/.46/.24 | **100%** |
| SIFT+VoxelMorph | .75/.68 | .13/.76/.17 | **.08**/.50/**.09** | **100%** |
| $f$ Emb. | .67/.71 | .21/.24/.21 | .25/.42/.27 | 90.9% |
| Ours (Stage 1) | .76/.77 | .11/.18/.12 | .12/.21/.12 | 95.4% |
| Ours (Stage 2) | **.81/.84** | **.10/.13/.08** | .09/**.14/.09** | **100%** |

Table 1: The same results as in table 1 of the main paper, differentiating between mesocosm and field experiments.

difficult to process. Table **??** shows the same results as table 1 of the main paper, split into the two experiment types. Contrary to our expectation, the performance differences are not very significant.

## 4. Additional Images

Figure **??** shows additional images of minirhizotron experiments for better understanding of the application background. Figures **??**, **??** and **??** show additional full-sized images and results.

(a) Minirhizotron tube installation

(b) A minirhizotron inserted into soil with high water content which is particularly prone to movement

(c) Mesocosm Experiments

(d) Field Experiments

Figure 3: Setup of minirhizotron experiments

Figure 4: Additional full-sized result illustrating the amount of root turnover that can occur over 10 months (from May to March). Our method is still able to capture the few remaining roots and differentiate them from new ones.

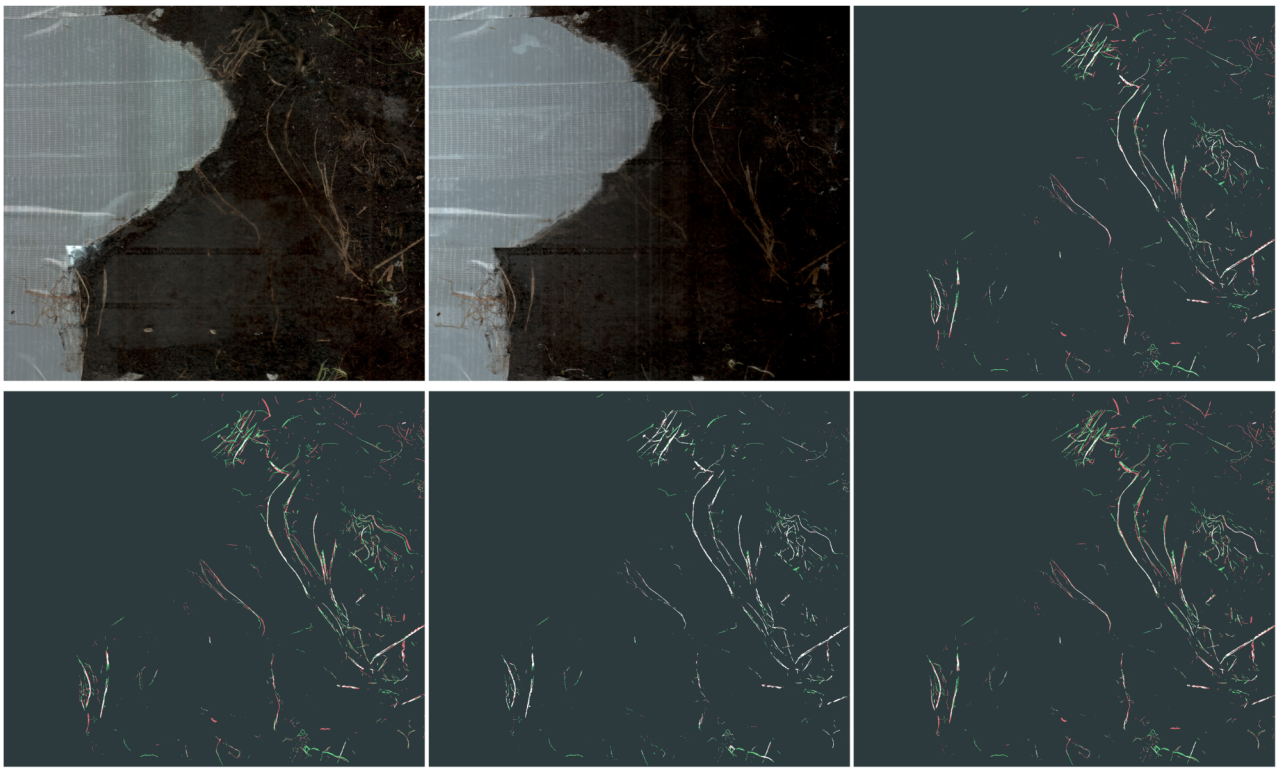


Figure 5: Additional full-sized results. Top row: input images and annotation turnover map. Bottom row: Outputs of the methods SIFT(sk), VoxelMorph and our stage 2 method.

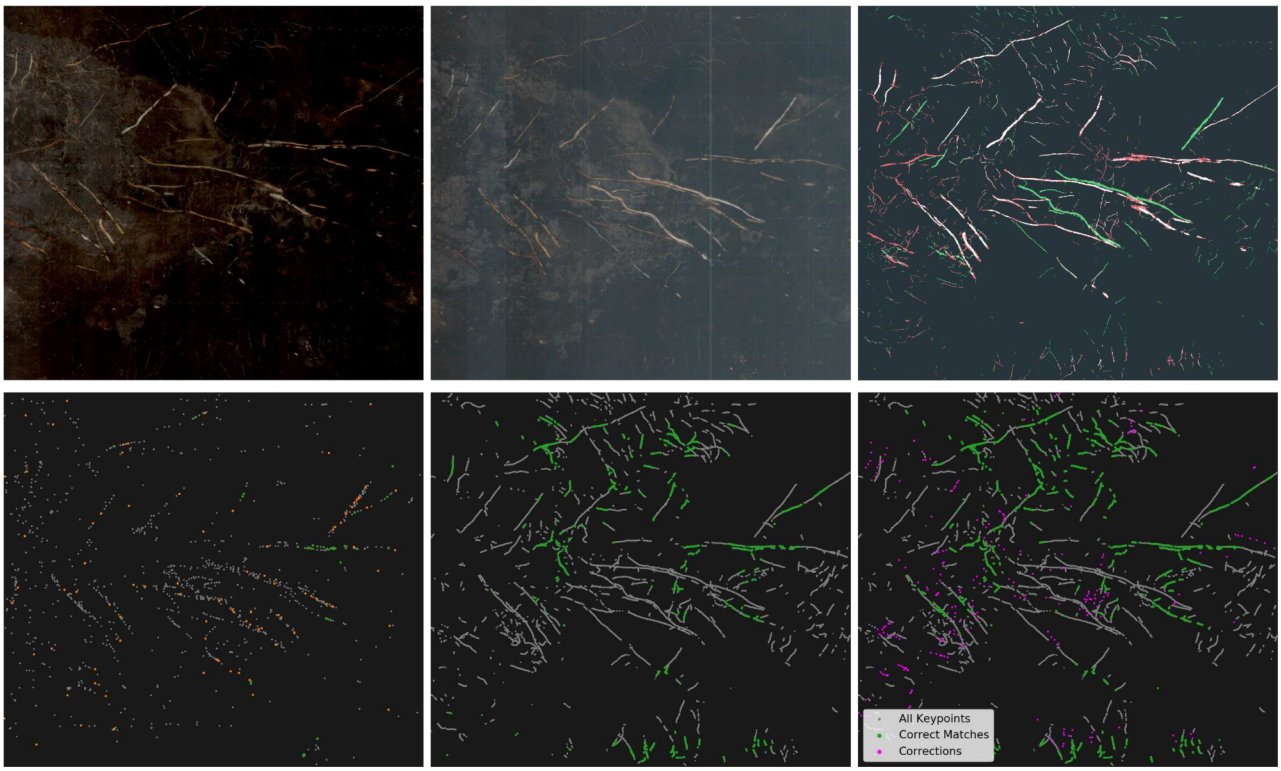



Figure 6: Top row: Full sized input images and stage 2 output turnover map
Bottom row: Keypoints of SIFT, stage1 and stage 2. Keypoints in gray were rejected by the ratio test, cross checking or outlier rejection. Magenta keypoints are the manual corrections added by annotators.

# 8. Identification and Measurement of Individual Roots in Minirhizotron Images of Dense Root Systems

| | | |
|---|---|---|
| Title | Identification and Measurement of Individual Roots in Minirhizotron Images of Dense Root Systems | |
| Authors | Alexander Gillert, Bo Peters, Uwe Freiherr von Lukas, Jürgen Kreyling | |
| Publication Venue | Computer Vision in Plant Phenotyping and Agriculture (CVPPA) Workshop at the IEEE/CVF International Conference on Computer Vision (ICCV) 2021, Virtual | |
| Status | Published | |
| DOI | 10.1109/ICCVW54120.2021.00153 | |
| Venue Ratings as of 2023-03-05 | CORE2021[1] | A* (Main conference) |
| | Research.com[2] | 13.30 (Impact Score, Workshop proceedings)<br>5th among all computer vision conferences |
| | Google Scholar[3] | 71 (h5-Index, Workshop proceedings)<br>13th among all computer vision venues |

[1] http://portal.core.edu.au/conf-ranks/638/

[2] https://web.archive.org/web/20230305125228/https://research.com/conference-rankings/computer-science/computer-vision

[3] https://web.archive.org/web/20230305125948/https://scholar.google.com/citations?view_op=top_venues&hl=en&vq=eng_computervisionpatternrecognition

# Identification and Measurement of Individual Roots in Minirhizotron Images of Dense Root Systems

Alexander Gillert[1] Bo Peters[2] Uwe Freiherr von Lukas[1,3] Jürgen Kreyling[2]

[1]Fraunhofer Institute for Computer Graphics Research IGD, Rostock
[2]Institute of Botany and Landscape Ecology, Greifswald University
[3]Institute for Visual & Analytic Computing, University of Rostock
{alexander.gillert, uwe.freiherr.von.lukas}@igd-r.fraunhofer.de
{bo.peters, juergen.kreyling}@uni-greifswald.de

## Abstract

*Semantic segmentation networks are prone to oversegmentation in areas where objects are tightly clustered. In minirhizotron images with densely packed plant root systems this can lead to a failure to separate individual roots, thereby skewing the root length and width measurements.*

*We propose to deal with this problem by adding additional output heads to the segmentation model, one of which is used with a ridge detection algorithm as an intermediate step and a second one that directly estimates root width. With this method we are able to improve detection and width measurements in densely packed roots systems without negative effects on sparse root systems.*

## 1. Introduction

Plant roots play a critical role in plant growth and many ecosystem processes and as such have become of increasing interest for ecosystem and climate modelling [1]. Despite their importance, research on belowground growth dynamics remains sparse due to inaccessibility of root systems and the often costly methods required for observation. With the development of rhizotrons and subsequently minirhizotrons, a nondestructive method for long-term monitoring of roots became available [5]. Rhizotrons often consist of two vertical glass panels separated by a thin layer of soil. Root growth in the soil along the inside of the transparent panels is then documented visually or photographically. Likewise, Minirhizotrons, transparent (acrylic-)glass tubes inserted into the soil, allow for in-situ monitoring of plant growth in natural conditions as root growth alongside the tube walls is documented with specialized scanners or cameras from within the tube.

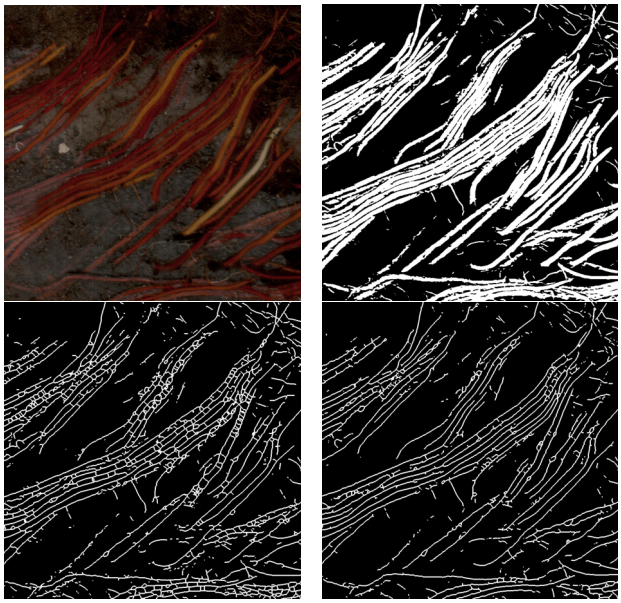


Figure 1: Top left: input image. Top right: output of a U-Net segmentation network that was trained to detect roots. Bottom left: the same output after applying the skeletonization postprocessing step. Many artifacts and loops are present due to the imperfection of the segmentation map. Bottom right: our method. (Skeleton images dilated for better visibility. Zoom in for details.)

The shape and size of plant root systems vary greatly between species and environmental con-

ditions. Whereas previous works on automated (mini)rhizotron imagery analysis [10, 7] mostly deal with plant species with sparse root systems with few or far apart growing roots (e.g. soy beans), in this work we are more interested in those with densely packed roots. Especially many graminoid (grass-like) species have a fibrous root system where all roots originate at the point where the aboveground plant body interfaces with the soil, resulting in densely packed root systems with roots growing in parallel and with little space in between.

Some of the most important metrics for plant growth research are total root length and root width. Estimating the root length in the images by simply taking the sum of all segmented root pixels leads to a bias towards large diameter roots. Therefore, most of the previous works [10, 7] employ a skeletonization postprocessing step to get a more accurate root length estimation. This works sufficiently well in images where individual roots are far apart, in scenarios with densely packed roots however, the segmentation network usually has difficulties classifying the boundaries between roots correctly and results in oversegmentation in those areas. The skeletonization method is then either unable to distinguish between individual roots and merges them into one, or even worse leads to loops which does not represent the true root system topology and results in incorrect measurements. This problem is illustrated in figure 1. One might argue that increasing the segmentation threshold would help to separate those roots, however this in turn would also negatively affect the detection of fine roots and width measurement.

We present a method to mitigate this problem by adding an intermediate ridge detection step. Specifically, we convolve a learned distance transform map with the second derivative of a gaussian kernel and analyze its response for curvature. Experimental results show a significant improvement in skeleton metrics for images with dense root systems, without negative impact on those with sparse ones. Moreover, we propose to estimate the width of the roots directly via regression which we have found to outperform baselines.

## 2. Related Work

Basic minirhizotron imagery analysis systems have been presented in [10, 7] and consist of a deep segmentation neural network based on the U-Net [6] architecture or similar. Research on improving the quality of results has focused on transfer learning [11] by pretraining on different plant species, data augmentation [7] in form of grid deformations, weak supervision [13, 14] with multiple instance learning to reduce the amount of required data annotations and inpainting [2, 3] to correct for undersegmentation, i.e. gaps in the segmentation results. The goal of our method can be seen as the opposite of the latter because we aim to mitigate the effects of oversegmentation. So far, no work has been published on this specific problem setting.

Most of these works postprocess the segmentation results with the topology preserving thinning algorithm published by Zhang and Suen [16] and implemented in the scikit-image library [9] as the skeletonize procedure. This algorithm works by iteratively removing the contour pixels on object borders until only the skeleton pixels are left. An issue with this algorithm is that it assumes perfect topology in the binary input image, which is not always the case in the output returned by a segmentation network and results in artifacts. Steger [8] proposed an algorithm for the detection of curvilinear structures, which we use in a simplified form as an intermediate step before skeletonization to mitigate these problems.

Already Ronneberger et al. in the original U-net paper [6] dealt with the problem of separating touching objects (HeLa cells). They used a weightmap to put additional emphasis on the border between the cells. The authors of [15] approach the problem of counting densely clustered objects by introducing an additional artificial "border" class and training a multi-class segmentation network. Although their field of application was counting grapevine berries, this approach is also applicable to root detection and we compare it to our method in our experiments.

A somewhat similar problem is instance segmentation where the goal is to separately segment different (possibly overlapping) objects belonging to the same class. However, standard architectures from this research area (e.g. Mask-RCNN [4]) are not applicable to minirhizotron images because roots with their elongated shapes do not fit well into the boxes prior and often cannot be seen as separate objects either, as they may branch off.

## 3. Methods

We approach the described problem by adding additional auxiliary output heads to a segmentation network which learn the distance transform of the ground truth. One of them is used to detect ridges, i.e. continuous curves of local maxima and the second one is used for width estimation. A schematic overview of our method is provided in figure 2.

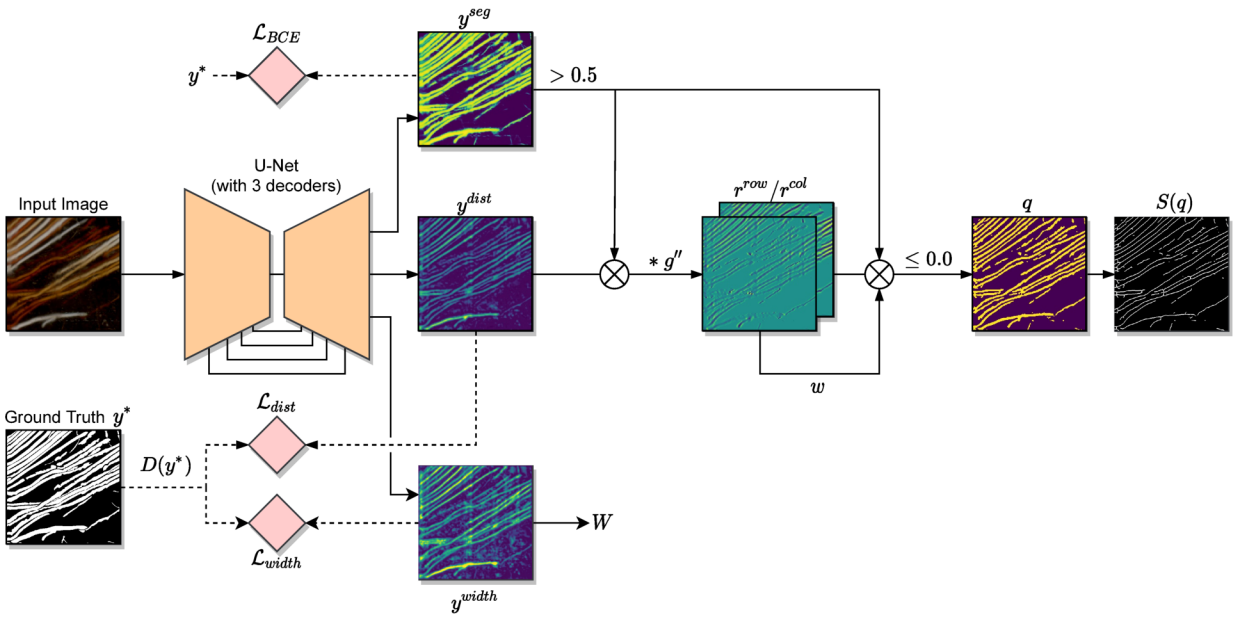



Figure 2: Schematic overview of our method. Solid lines represent data flow, dashed lines represent losses.

### 3.1. Network Architecture and Training

We use the U-Net [6] architecture as the base for our network and modify it to produce three output maps which we denote with $y^{seg}$, $y^{dist}$ and $y^{width}$. Each output is computed by an own separate decoder network, connected to a shared encoder.

The main segmentation head is trained in a standard supervised manner with the binary crossentropy (BCE) loss, whereas the auxiliary heads are trained on the distance transform of the binary ground truth segmentation map $y^*$. Specifically for $y^{dist}$ we use the mean absolute error on pixels where $y^*$ is positive and ignore pixels that are zero to avoid distraction:

$$\mathcal{L}_{dist} = \frac{y^* \cdot |y^{dist} - D(y^*)|}{\sum_i y_i^*} \tag{1}$$

where $D$ is the distance transform. During inference we zero out the predicted distance values where the output of the main segmentation head $y^{seg}$ is below a threshold (0.5 where not stated otherwise). In the following we use $y^{dist}$ as a shorthand notation for $y^{dist} \cdot (y^{seg} > 0.5)$.

Although $y^{dist}$ learns the distance transform and in theory could be directly used for width prediction, in practice this output head is biased towards small values because it also learns the pixels close to the root border. Therefore, we train the $y^{width}$ head to directly estimate the width of the roots by only learning the distance transform of the center pixels:

$$\mathcal{L}_{width} = \frac{S(y^*) \cdot |y^{width} - D(y^*)|}{\sum_i S(y^*)_i} \tag{2}$$

where $S$ is the skeletonization method [16, 9]. For both auxiliary heads no additional annotation except for the standard binary segmentation map is required.

The overall loss function is given by:

$$\mathcal{L} = \mathcal{L}_{BCE} + \lambda_0 \mathcal{L}_{dist} + \lambda_1 \mathcal{L}_{width} \tag{3}$$

where $\lambda_0$ and $\lambda_1$ are balancing hyperparameters which we both set to 0.01. We have found larger values to have a negative effect on the main segmentation head.

We use the SGD optimizer for 15 epochs with a learning rate of 0.1, momentum 0.9 and reduce the learning rate twice by a factor of 0.1.

### 3.2. Ridge Detection

As discussed in more detail by Steger in [8], a well established method to detect lines in a one-dimensional function is to convolve it with the second derivative of a gaussian kernel and use the zero crossings of the result as the edges of the line. The second derivative of the gaussian kernel is defined as:

$$g''_\sigma(x) = \frac{x^2 - \sigma^2}{\sqrt{2\pi}\sigma^5} e^{-\frac{x^2}{2\sigma^2}} \tag{4}$$

where $\sigma$ is the standard deviation hyperparameter which affects the width of the kernel. We use $\sigma = 3$ where not otherwise mentioned. This kernel converts the signal into a scale-space description and smoothes out noise in the data.

Since it holds that $(g'' * f)(x) = (g * f)''(x)$, the response of this convolution can be regarded as the smoothed second derivative of the function $f$, i.e. it represents its curvature. By looking at negative values of the response, one can find right-handed turns in the function, i.e. local maxima.

For two-dimensional data, Steger [8] recommends applying the 1D convolution in the direction perpendicular to the line or ridge. This brings the drawbacks of firstly the need to estimate the angle at each line location and secondly many separate convolutions at different angles. In large images and with many ridges, as can be the case in minirhizotron data, this gets very costly. Therefore we opt for a simpler method of convolving in only two directions, namely row-wise and column-wise to get the response maps $r^{row}$ and $r^{col}$:

$$\begin{aligned} r^{row}_{ij} &= (g''_\sigma * y^{dist}_{row_i})_j \\ r^{col}_{ij} &= (g''_\sigma * y^{dist}_{col_j})_i \end{aligned} \tag{5}$$

where $y^{dist}_{row_i}$ and $y^{dist}_{col_j}$ represent the $i$-th row and the $j$-th column of $y^{dist}$ respectively.

The magnitude of the response in $r^{row}$ is the largest for ridges running in vertical direction with still reasonable results diagonally but reduces to noise in the horizontal direction. The opposite applies to $r^{col}$. To get an acceptable response in all directions we combine both response maps with weight maps $w^{row}$ and $w^{col}$ which are constructed as:

$$w^{row} = \frac{\delta_\rho(-r^{row})}{\delta_\rho(-r^{row}) + \delta_\rho(-r^{col}) + \epsilon} \tag{6}$$

where $\epsilon = 10^{-6}$ a small constant to guard against zero division and $\delta_\rho$ the dilation operation with a $\rho$ pixels sized structuring element. In practice we use the max-pooling operation with a kernel of size $\rho$. Where not otherwise mention we set $\rho = 11$. $w^{col}$ is defined analogously. These weight maps are then used to combine the two response maps into $\hat{r}$:

$$\hat{r} = w^{row}\, r^{row} + w^{col}\, r^{col} \tag{7}$$

Next, we define ridges as pixels where $\hat{r}$ is negative and since the convolution operation might spill over, also check for $y^{seg}$:

$$q = \begin{cases} 1, & \hat{r} < 0.0 \text{ and } y^{seg} > 0.5 \\ 0, & \text{otherwise} \end{cases} \tag{8}$$

Lastly, we apply the skeletonization method [16, 9] on $q$ to estimate the centers of the roots which we denote with $S(q)$.

### 3.3. Width Estimation

A disadvantage of convolving only in two directions instead of perpendicular to the ridge direction is that the width of the detected ridges cannot be directly determined. Instead, we simply take the regressed values of $y^{width}$ where the center of the ridges was detected:

$$W = \begin{cases} y^{width}, & \text{where } S(q) = 1 \\ \text{undefined}, & \text{otherwise} \end{cases} \tag{9}$$

We have considered other choices, for example using the distance transform $D(y^{seg})$ instead. We evaluate this choice in the experimental section.

## 4. Experimental Setup

### 4.1. Dataset

Our training dataset consists of 64 minirhizotron images acquired from mesocosm and field experiments and an additional of 32 images is used for evaluation. The images mostly contain roots of Carex rostrata, Mentha aquatica and Equisetum fluviatile plant species and were acquired with a CI-600 In-Situ Root Imager (CID Bio-Science Inc.). All images stem from a facility that was established in fall 2018 at the Institute of Botany and Landscape Ecology in Greifswald, Mecklenburg Western Pomerania.

The dataset was carefully annotated by ecologists. Annotators were explicitly asked to leave a minimum one-pixel wide boundary inbetween the roots to make sure that the distance transform and skeletonization methods perform without issues on the ground truth.

For training and inference, the images are sliced into overlapping 512x512px patches with 32px overlap. Ridge detection and evaluations were performed on the stitched full-sized pictures.

For a more fine-grained evaluation we manually classify images as containing dense or sparse root systems. We count 22 sparse and 10 dense images in our test set. The seemingly low number of evaluation images is due to the very high cost of manual annotation which can be as high as 15 man-hours for a single image.

### 4.2. Metrics

Standard metrics that are often used in semantic segmentation like precision, recall or the Dice coefficient are not well suited to be directly used with skeletons due to their sparseness: a shift by a single pixel can have a large impact on the result. Therefore, our main metrics for the identification of roots are *skeleton completeness* $C_p$ and *correctness* $C_r$, which are discussed in more detail in [12] and defined as:

$$C_r = \frac{TP_\rho}{TP_\rho + FP_\rho} \quad C_p = \frac{TP_\rho}{TP_\rho + FN_\rho} \tag{10}$$

where $TP_\rho$, $FP_\rho$ and $FN_\rho$ are the *buffered* versions of the number of true positives, false positives and false negatives. They are defined with dilated skeletonized predictions $\delta_\rho(S(y))$ and dilated skeletonized ground truth images $\delta_\rho(S(y^*))$ as:

$$\begin{aligned} TP_\rho &= S(y) \cap \delta_\rho(S(y^*)) \\ FP_\rho &= S(y) \cap \overline{\delta_\rho(S(y^*))} \\ FN_\rho &= \overline{\delta_\rho(S(y))} \cap S(y^*) \end{aligned} \tag{11}$$

Intuitively speaking, a predicted skeletonized pixel is considered a true positive if it is within $\rho$ pixels distance of a skeletonized ground truth pixel. We use $\rho = 1$ for all our evaluation experiments.

Moreover, we evaluate the harmonic mean of both metrics calculated as $H = \frac{2C_rC_p}{C_r+C_p}$ (also known as the F1 score when using precision and recall) and the overall root length in an image, which we estimate with the sum of all skeletonized pixels.

We compare the width measurements via a *histogram* and a *direct comparison* metric. For the histogram metric we count the skeletonized root pixels into three categories based on the measured width: fine (<3px), medium (3-7px) and coarse (>7px). Then we compare the absolute and relative error of those bins. For the relative error we use the mean absolute percentage error (MAPE):

$$MAPE = \frac{1}{N} \sum \frac{|x^{true} - x^{predicted}|}{\max(\epsilon, x^{true})} \tag{12}$$

where $\epsilon$ guards against small values in the denominator. Since our ground truth values are not well scaled, i.e. can vary in a large range from zero to tens of thousands counted pixels within the same bin, we set it to the average value in the histogram bin: $\epsilon = \frac{1}{N} \sum x^{true}$.

This is also the metric that the end user would be most interested in, however it is dependent on the quality of the upstream root detection system: if a root is not detected it cannot be sorted into a bin. Therefore we also directly compare the widths at the $TP_\rho$ locations to isolate the width measurement evaluation from the detection.

The width ground truth is computed via the distance transform of the ground truth segmentation map.

### 4.3. Compared Methods

For skeleton metrics, we compare the following approaches:

- *Baseline*: Segmentation network trained only with binary cross-entropy loss. Skeletonization applied directly on the thresholded $y^{seg}$.
- *Weightmap*: Same as baseline, but trained with a weightmap as in [6] that puts additional emphasis on the pixels inbetween roots.
- *Multi-class segmentation* similar to that of [15]. For this method we train a segmentation network to classify each pixel into three classes, namely "root", "border" and "background". The annotation for the "border" class was automatically generated by applying the dilation operation onto the segmentation map with a 2px sized structuring element. For the evaluation we use only the pixels that were classified as "root".
- *Ridge detection* applied on the segmentation head output $y^{seg}$. This method can be used with a normal segmentation network without adjustments to the architecture.
- *Ridge detection* applied on the auxiliary head output $y^{dist}$. This is our main method as described in section 3

For a fair comparison, neural networks for methods which do not require $y^{dist}$ were trained without the auxiliary heads.

For the width measurement we compare the *regression* based approach as in subsection 3.3 to measuring the width via distance transform on either the baseline skeleton or the skeleton as computed in subsection 3.2.

## 5. Results

Our main results for the skeleton metrics are presented in table 1. The multi-class segmentation method provides better skeleton correctness performance but at the cost of a worse skeleton completeness, thus no clear improvement is made with this

| | Dense | | | Sparse | | |
|---|---|---|---|---|---|---|
| Method | Cp↑ | Cr↑ | H↑ | Cp↑ | Cr↑ | H↑ |
| Baseline | 0.571 | 0.639 | 0.599 | 0.525 | 0.613 | 0.583 |
| Weightmap | 0.591 | 0.658 | 0.618 | 0.518 | 0.653 | 0.581 |
| Multi-class Segmentation | 0.509 | 0.649 | 0.568 | 0.460 | 0.665 | 0.543 |
| Ridge Detection on $y^{seg}$ | **0.606** | 0.676 | 0.634 | 0.525 | 0.612 | 0.583 |
| Ridge Detection on $y^{dist}$ | 0.596 | **0.733** | **0.653** | **0.533** | **0.676** | **0.603** |

Table 1: Mean skeleton metrics of of the compared methods. Bold font indicates best values.

| | Dense | | | | Sparse | | | |
|---|---|---|---|---|---|---|---|---|
| Method | Fine↓ | Medium↓ | Coarse↓ | Direct↓ | Fine↓ | Medium↓ | Coarse↓ | Direct↓ |
| Baseline | 10837.6 / 0.299 | 7822.4 / 0.118 | 5228.9 / 0.755 | 0.997 | 2741.7 / 0.203 | 2280.7 / 0.410 | 38.0 / 0.375 | 0.654 |
| Weightmap | **9610.2** / 0.282 | 7015.0 / 0.123 | 3390.3 / 0.295 | 0.965 | 2720.6 / 0.203 | 2564.5 / 0.474 | 41.5 / 0.271 | 0.648 |
| Ridge Detection | 10860.5 / 0.303 | **6246.8** / **0.087** | 3201.1 / 0.285 | 0.905 | 2493.8 / 0.193 | 2333.8 / 0.427 | **29.0** / **0.237** | 0.645 |
| $y^{width}$ Regression | 9877.2 / **0.273** | 6357.1 / 0.089 | **1841.2** / **0.192** | **0.810** | **1972.8** / **0.180** | **1745.9** / **0.275** | 30.7 / 0.238 | **0.578** |

Table 2: Mean width measurement errors for the compared methods. The vales for the fine, medium and coarse histogram bins are average count errors/MAPE. Direct stands for directly compared width values at $TP_\rho$ coordinates in pixel units. Bold font indicates best values.

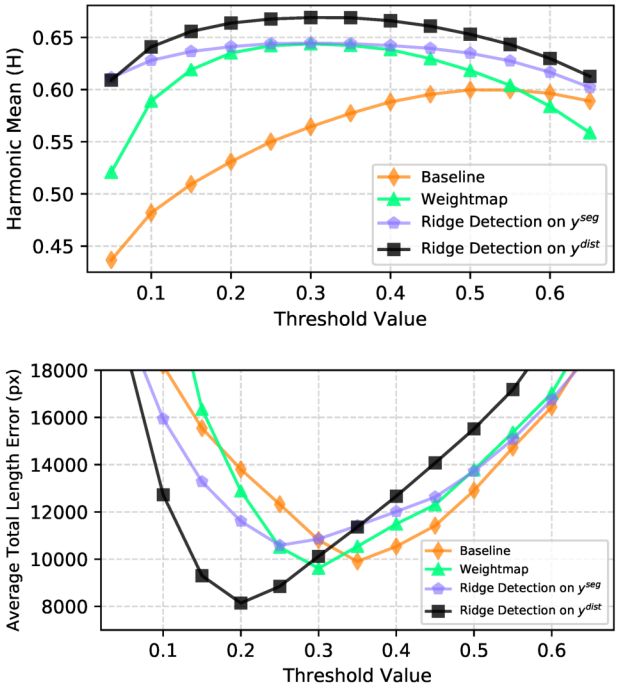



Figure 3: Influence of the segmentation threshold on the skeleton metrics and total root length. Advanced methods like ridge detection benefit more from a lower threshold than the baseline. (Evaluated on dense root systems.)

method. The weightmap method is overall more beneficial however with a slight degradation for sparse root systems.

For dense root systems, ridge detection applied to $y^{seg}$ improves both skeleton completeness (i.e. less false negatives) and correctness (less false positives). At the same time no significant deterioration in performance is observed for sparse root systems. It performs even better if applied on $y^{dist}$ where both dense and sparse metrics are improved. We explain this with the smoother surface of $y^{dist}$ compared to $y^{seg}$ which benefits ridge detection.

We note that higher performance is achievable by reducing the segmentation threshold as is shown in figure 3. In the baseline method, increasing or decreasing the threshold is mostly a tradeoff between false positives or false negatives. This is also illustrated in the example in figure 5 where either four roots get detected as two, or a fine root won't get recognized. Ridge detection on the other hand overall benefits from lower thresholds. This is because the full information of the raw decimal values is used directly, instead of just binary thresholded values.

The threshold should also be adjusted to achieve a better total root length estimate. With the default threshold of 0.5 the baseline seemingly performs better on this metric, however, this is due to a higher number of false positives which balance out the general underestimaton of the overall root length. With a lower threshold of around 0.2, our method gives the best length estimate of the compared methods.

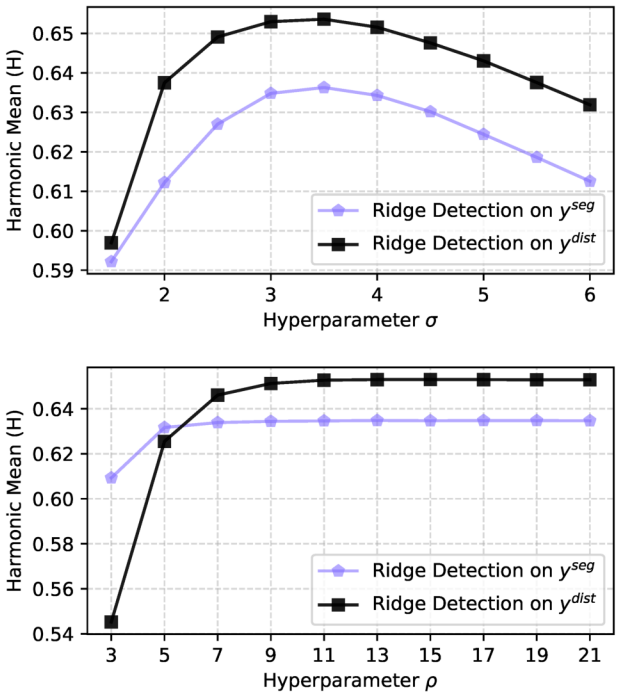



Figure 4: Influence of hyperparameters on the performance. Top: The width of the gaussian kernel $\sigma$ has an optimium in the range 3 to 4. Bottom: The dilation parameter $\rho$ for the width map computation should have minimum value of 9. Larger values do not lead to much improvement but only increase computation times. (Evaluated on dense root systems.)

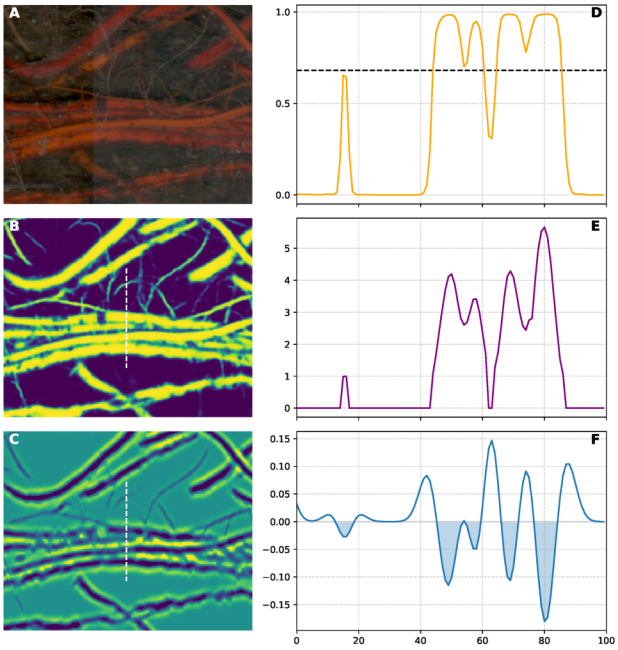



Figure 5: A particularly challenging example. (A) patch of an input image; (B) Output of the segmentation head $y^{seg}$; (C) $\hat{r}$ as computed in equation 7; (D) plot of $y^{seg}$ extracted at the white dashed line. Not all roots can be detected or separated via simple thresholding, no matter where the threshold is set, as indicated by the black dashed line; (E) plot of $y^{dist}$ at the white dashed line; (F) plot of $\hat{r}$ at the white dashed line, the shaded areas represent values below zero and thus detected ridges as in equation 8. All five roots can be detected.

For the width measurements in table 2 we observe that regression gives the best performance in almost all measured metrics, especially for sparse root systems. We attribute this to oversegmentation in the segmentation-based approaches. With fine roots, oversegmentation is due to the difficulty to reliably segment thin objects whereas with coarse roots this is rather due to their proximity to each other. Regression is immune to this.

Some qualitative results are shown in figure 6. More hyperparameter tests can be seen in figure 4.

## 6. Conclusion

We have presented a method for improving the topology reconstruction of dense root systems in minirhizotron images as well as width estimation of individual roots. This is done by performing an intermediate ridge detection step on a learned distance map before the commonly used skeletonization method. This helps to mitigate the oversegmentation of close or overlapping roots. Images with predominantly sparse root systems are not negatively affected.

As a slight drawback of our width estimation method can be regarded that it is based on regression, thus it functions as a black box and lacks interpretability. However, in our experiments it clearly outperforms baselines.

## ACKNOWLEDGEMENTS

This work has been supported by the European Social Fund (ESF) and the Ministry of Education, Science and Culture of Mecklenburg-Vorpommern, Germany under the project "DigIT!" (ESF/14-BM-A55-0015/19).

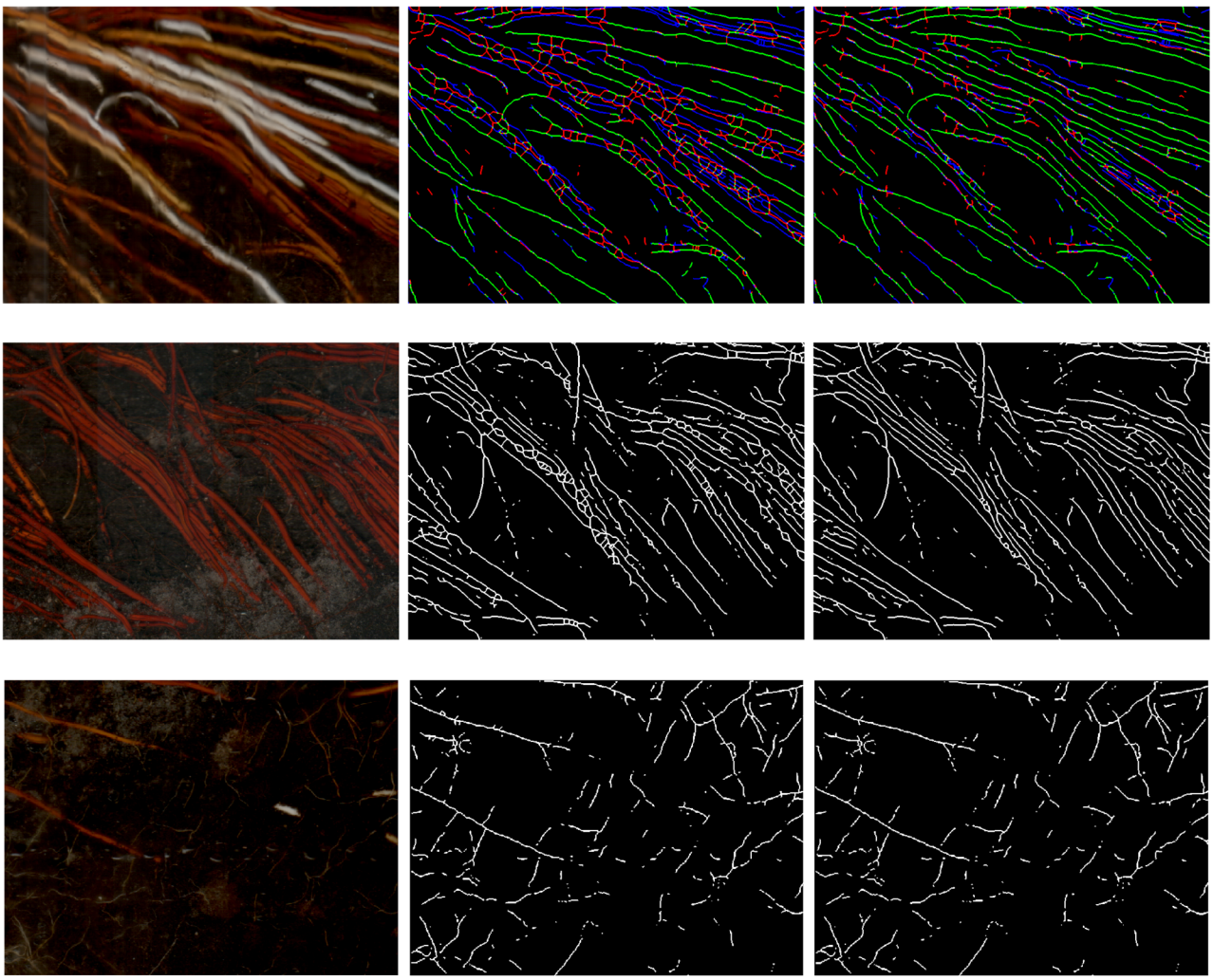


Figure 6: Some qualitative example results. From left to right: input image, baseline skeletonization, our method. In the top row green pixels represent $TP_\rho$, red pixels $FP_\rho$, blue pixels $FN_\rho$. Note the blurriness due to high humidity in this image. The bottom row shows an example with a sparse root system, almost no differences between the two methods in this case. (Skeleton images dilated for better visibility.)

## References


[1] Gordon B Bonan. Forests and climate change: forcings, feedbacks, and the climate benefits of forests. *science*, 320(5882):1444–1449, 2008. 1

[2] Hao Chen, Mario Valerio Giuffrida, Sotirios A Tsaftaris, and Peter Doerner. Root gap correction with a deep inpainting model. In *BMVC*, page 325, 2018. 2

[3] Hao Chen, Mario Valerio Giuffrida, Peter Doerner, and Sotirios A Tsaftaris. Adversarial large-scale root gap inpainting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pages 0–0, 2019. 2

[4] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross B. Girshick. Mask r-cnn. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 42:386–397, 2020. 2

[5] Mark G Johnson, David T Tingey, Donald L Phillips, and Marjorie J Storm. Advancing fine root research with minirhizotrons. *Environmental and Experimental Botany*, 45(3):263–289, 2001. 1

[6] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer, 2015. 2, 3, 5

[7] Abraham George Smith, Jens Petersen, Raghavendra

Selvan, and Camilla Ruø Rasmussen. Segmentation of roots in soil with u-net. *Plant Methods*, 16(1):1–15, 2020. 2

[8] Carsten Steger. An unbiased detector of curvilinear structures. *IEEE Transactions on pattern analysis and machine intelligence*, 20(2):113–125, 1998. 2, 3, 4

[9] Stéfan van der Walt, Johannes L. Schönberger, Juan Nunez-Iglesias, François Boulogne, Joshua D. Warner, Neil Yager, Emmanuelle Gouillart, Tony Yu, and the scikit-image contributors. scikit-image: image processing in Python. *PeerJ*, 2:e453, 6 2014. 2, 3, 4

[10] Tao Wang, Mina Rostamza, Zhihang Song, Liangju Wang, G McNickle, Anjali S Iyer-Pascuzzi, Zhengjun Qiu, and Jian Jin. Segroot: a high throughput segmentation method for root image analysis. *Computers and Electronics in Agriculture*, 162:845–854, 2019. 2

[11] Weihuang Xu, Guohao Yu, Alina Zare, Brendan Zurweller, Diane L. Rowland, Joel Reyes-Cabrera, Felix B. Fritschi, Roser Matamala, and Thomas E. Juenger. Overcoming small minirhizotron datasets using transfer learning. *Computers and Electronics in Agriculture*, 175:105466, 2020. 2

[12] Rabaa Youssef, Anne Ricordeau, Sylvie Sevestre-Ghalila, and Amel Benazza-Benyahya. Evaluation protocol of skeletonization applied to grayscale curvilinear structures. In *2015 International Conference on Digital Image Computing: Techniques and Applications (DICTA)*, pages 1–6, 2015. 5

[13] Guohao Yu, Alina Zare, Hudanyun Sheng, Roser Matamala, Joel Reyes-Cabrera, Felix B Fritschi, and Thomas E Juenger. Root identification in minirhizotron imagery with multiple instance learning. *Machine Vision and Applications*, 31(6):1–13, 2020. 2

[14] G. Yu, A. Zare, Weihuang Xu, R. Matamala, J. Reyes-Cabrera, F. Fritschi, and T. Juenger. Weakly supervised minirhizotron image segmentation with mil-cam. In *ECCV Workshops*, 2020. 2

[15] Laura Zabawa, A. Kicherer, L. Klingbeil, Andres Milioto, R. Töpfer, H. Kuhlmann, and R. Roscher. Detection of single grapevine berries in images using fully convolutional neural networks. *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 2571–2579, 2019. 2, 5

[16] T. Y. Zhang and C. Y. Suen. A fast parallel algorithm for thinning digital patterns. *Commun. ACM*, 27(3):236239, Mar. 1984. 2, 3, 4

# 9. Towards Combined Open Set Recognition and Out-of-Distribution Detection for Fine-grained Classification

| | | |
|---|---|---|
| Title | Towards Combined Open Set Recognition and Out-of-Distribution Detection for Fine-grained Classification | |
| Authors | Alexander Gillert, Uwe Freiherr von Lukas | |
| Publication Venue | International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications (VISAPP) 2021, Virtual | |
| Status | Published | |
| DOI | 10.5220/0010340702250233 | |
| Venue Ratings as of 2023-03-05 | CORE2021[1] | B |
| | Research.com[2] | 1.90 (Impact Score) |

[1] http://portal.core.edu.au/conf-ranks/958/

[2] https://web.archive.org/web/20230305130805/https://research.com/conference/international-joint-conference-on-computer-vision-imaging-and-computer-graphics-theory-and-applications

# Towards Combined Open Set Recognition and Out-of-Distribution Detection for Fine-Grained Classification

Alexander Gillert[1], and Uwe Freiherr von Lukas[1,2]
[1]*Fraunhofer Institute for Computer Graphics Research IGD, Rostock, Germany*
[2]*Department of Computer Science, University of Rostock, Germany*
{*alexander.gillert, uwe.freiherr.von.lukas*}*@igd-r.fraunhofer.de*

Keywords: Fine-Grained Classification, Out-of-Distribution Detection, Open Set Recognition

Abstract: We analyze the two very similar problems of Out-of-Distribution (OOD) Detection and Open Set Recognition (OSR) in the context of fine-grained classification. Both problems are about detecting object classes that a classifier was not trained on, but while the former aims to reject invalid inputs, the latter aims to detect valid but unknown classes. Previous works on OOD detection and OSR methods are evaluated mostly on very simple datasets or datasets with large inter-class variance and perform poorly in the fine-grained setting. In our experiments, we show that object detection works well to recognize invalid inputs and techniques from the field of fine-grained classification, like individual part detection or zooming into discriminative local regions, are helpful for fine-grained OSR.

## 1 Introduction

According to recent estimates (Barrowclough et al., 2016) there may be more than 18,000 species of birds in the world. When building a vision based bird classification system, it is infeasible to maintain an image dataset for training on all of them, especially since many are still undiscovered. Even limiting the classification to species from a local area is extremely challenging due to Zipf's law (Zipf, 1932), which implies that for the majority of object classes only few data samples are available. Thus, one usually has to resort to train on a dataset of only those species for which enough training data are available, which leaves room for error when the system encounters rare birds which are not in the training dataset. Additionally, in the end there is often little control over whether the deployed system will be used only on the species from that local area or on birds at all. In short: the testing distribution of deployed systems is rarely the same as the training distribution. This problem applies to many more areas, not only bird classification.

In machine learning, this problem is known as **Open Set Recognition (OSR)** or **Out-of-Distribution (OOD) Detection**. The difference between OOD detection and OSR is subtle and those two terms are sometimes used synonymously in literature. Strictly speaking however, in OOD detection, a classifier is trained on one dataset and evaluated on another, usually completely unrelated dataset, whereas in OSR a subset of classes from a dataset is used for training and a hold-out set of other classes from the same dataset is used for evaluation. OOD detection is thus more concerned with recognizing

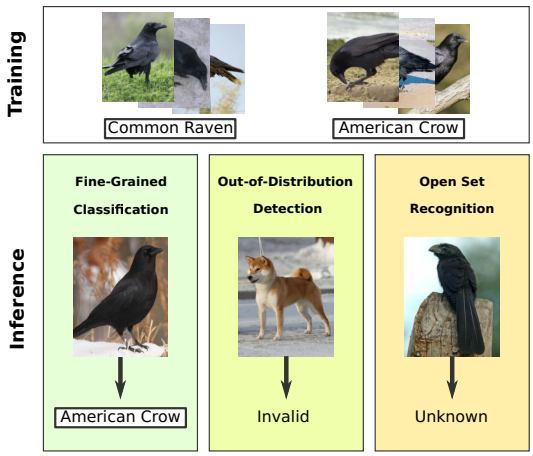



Figure 1: Illustration of the problem: A classifier is trained on images of two very similar classes. During inference, when presented with an image from one of those classes it should predict the correct label. If the input image is from a class that is not in the training distribution, it should either reject the sample as invalid or mark it as a valid but unknown class, depending on the semantic distance.

or rejecting invalid inputs, the goal of OSR on the other hand is to recognize new or unknown (but valid) classes. Figure 1 gives an illustration of the two problems.

In this work, we are interested in combining both problems: without additional training data detecting object classes that the classification system was not trained on but also making the distinction between completely invalid or valid but unknown classes. We approach the OOD detection objective as an object detection task: object detectors learn to differentiate between object and background within the same image. The detected objects, or object parts, are then useful to differentiate between known and unknown classes for the OSR objective.

It is well known (Guo et al., 2017), that despite ever increasing accuracy, modern neural networks tend to be poorly calibrated. This means, they are prone to give overly confident prediction results, even in when the prediction is incorrect. Even worse, high confidence predictions are often returned if the input is completely unrecognizable (Nguyen et al., 2015), slightly perturbed (Goodfellow et al., 2014) or irrelevant (Hendrycks and Gimpel, 2017) to the task, i.e input outside of the distribution that the network was not trained on. We observe that this problem worsens even more when dealing with fine-grained distributions.

The contributions of this paper are as follows:

- A combination of OSR and OOD detection by making a distinction between **known**, **unknown** and **invalid** classes
- A framework based on **object detection** with both **strong and weak supervision** that is able to recognize the above error cases without explicitly being trained on them
- Baseline evaluations on several realistic **fine-grained** datasets

## 2 Related Work

### 2.1 Open Set and Out-of-Distribution Detection

Despite the difference between OSR and OOD detection noted above, we treat both as the same problem in this section, since the methods are mostly applicable to both.

The work of (Hendrycks and Gimpel, 2017) has set up a baseline for OOD detection. They noticed that trained neural networks predict higher softmax scores for examples that are close to the training dataset than for those new to them. ODIN (Liang et al., 2018) extends this idea by using temperature scaling and modifying the input images with pertrubations created from the backpropagated gradient to separate in-distribution from out-of-distribution samples. OpenMax (Bendale and Boult, 2016) fits a Weibull probability distribution on a per-class basis to recalibrate the output activations between the trained classes and an additional rejection class. If the rejection class has the maximum activation or if the maximum activation falls below a threshold, the input is rejected. (Lee et al., 2018) defines a confidence score using the Mahalanobis distance at multiple layers within a network.

A common issue with the above works is that they mostly use very simple datasets for evaluation: often used datasets are MNIST, CIFAR-10 or even random noise. Very few works evaluate on ImageNet (Russakovsky et al., 2015) and similar datasets. In our evaluation we are interested in more challenging and realistic fine-grained datasets.

The authors of (Ren et al., 2019) recognized the need for more realistic evaluations in this field and published a dataset for OOD prediction of genomic sequences of bacteria. For this task, they introduce likelihood ratios, which can be also applied to images, yet they also evaluate only on coarse image datasets.

An obvious method for detecting unknown classes is regularization with a background class during training. In (Hendrycks et al., 2018), the authors vastly improved OOD detection performance by using an auxiliary dataset as background examples. In a sense, we also use this method, albeit implicitly since we employ object detectors. In object detection, an image is divided into positive and background samples. We thus do not use additional data but only train on the images that are relevant for the main classification task.

A somewhat related area of research is **Generalized Zero-Shot Learning**: here, a classifier is trained on images together with a vector of attributes for each class. At test time, new classes along with their attributes are added to the pool and the classifier has to predict the correct class according to the attributes. Naturally, the classifier is biased towards the old, seen classes, thus many algorithms employ a gating mechanism which tries to predict whether the input image belongs to the seen or to the unseen classes. (Chen et al., 2020) used a a spherical variational autoencoder to achieve remarkable OSR performance on the fine-grained Caltech-UCSD-Birds (CUB) (Wah et al., 2011) and Oxford Flowers datasets (Nilsback and Zisserman, 2008). However, this method requires the ad-

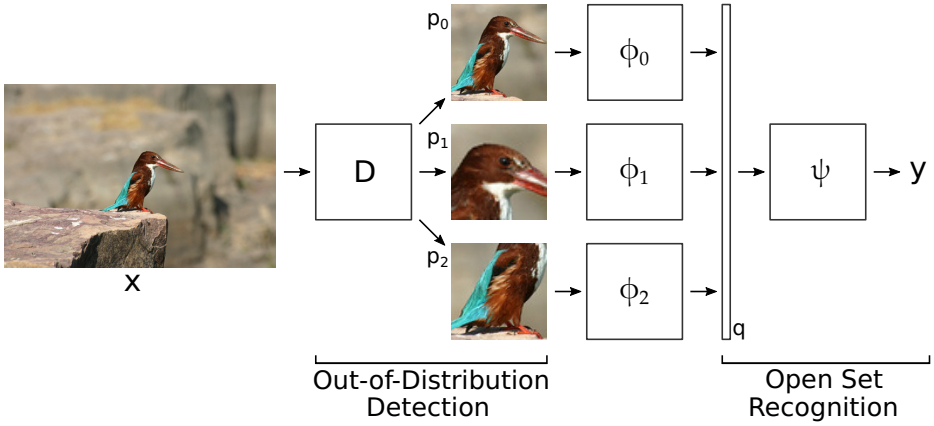



Figure 2: Overview of our classification framework

ditional annotation of visual attributes. In contrast, in our weakly supervised method we only use image-level labels.

Worth noting is also the research area **Selective Prediction** or **Classification with Reject Option**. Here, the goal is to make a model abstain from making a prediction when it is not confident enough (Geifman and El-Yaniv, 2019), for example if the input is too difficult or noisy. However, the works in this field do not evaluate on new or unknown classes, but only on the same classes that the model was trained on.

### 2.2 Fine-Grained Classification

Fine-grained object categories exhibit a low inter-class and a high intra-class variance. Therefore, for the classification of those objects, subtle details on (body) parts often play an important role. Many previous works have focused on discrimitive local part localization to improve performance, e.g (Zhang et al., 2014; Lin et al., 2015; Berg and Belhumeur, 2013) only to name some of the most important ones. Specifically, the method of (Zhang et al., 2014) first detects those regions as bounding boxes, then extracts features from those regions with specialized feature extractors and finally combines those reigons with a SVM. We pick up this idea for our classification framework with minor modifications.

To avoid having to rely on costly additional part annotations, a large body of works has focused on weakly supervised methods. For example, (Fu et al., 2017) uses a recursive attention mechanism to zoom into the object of interest at different levels. Simply getting a higher resolution crop of the object helps to improve the classification accuracy and as we show in this paper, also the OSR performance. We also use weak supervision, albeit on a much simpler scale and only on one level.

## 3 Methods

### 3.1 Classification Framework

Our general intuition is that discriminitve local parts are beneficial for fine-grained open set recognition, as is the case for classification. Furthermore a failure to detect those parts should indicate that the input is invalid.

Our classification system is based on that of (Zhang et al., 2014), an overview is shown in figure 2. We first train an object detector $D$ to detect individual object parts. The targets for the object detector come either directly from annotations if available (strong supervision) or from pseudo-labels generated from class activation maps as described in 3.2 (weak supervision). We do not use the geometric constraints between individual parts that were introduced in (Zhang et al., 2014), because we have found the detector to perform reasonably well without them.

The output of $D$ is a set of tuples $(b_i, s_i, c_i)$ representing the predicted bounding box, confidence score and part class, respectively. We take boxes $b_i$ with the highest score for each part class $c_i$ and feed the corresponding image crops $p_i$ into feature extractors $\phi_i$, that were fine-tuned for the respective parts. The features $\phi_i(p_i)$ are then concatenated into vector $\mathbf{q}$ and fed into a final small network $\psi$ consisting of two linear layers with ReLU activation which gives the classification result $y$.

### 3.2 Weakly Supervised Bounding Boxes

Since bounding box annotations for body parts are difficult to obtain, we additionally conduct experiments with automatically generated pseudo label bounding boxes as training targets for the object detector. We opt for the simple method based on class activation maps (CAM) as described by (Zhou et al., 2016). The CAM heatmaps are created from an image classifier that was trained on whole images.

For simplicity reasons, we only generate a single box which represents the whole object instead of individual body parts. The bounding box is generated by thresholding the CAM heatmap and then taking the minimum and maximum coordinates of the largest connected component. For the threshold we use a percentage of the maxium CAM value. We use 50% for all evaluation datasets.

The generated pseudo label boxes are sometimes not very accurate but since they are only used as training targets for the object detector, outliers are mostly recovered after training.

### 3.3 OOD Detection Decision

For the OOD detection decision, i.e. whether an object is valid or invalid, we directly use the scores $s_i$ returned by the object detector. For multiple boxes, the score is averaged and thresholded with a value $\delta_{OOD}$ that has to be calibrated through cross validation. Formally, the decision function looks as follows:

$$f_{OOD}(x) = \begin{cases} \text{valid} & \text{if } \frac{1}{N}\sum_i^N s_i \geq \delta_{OOD} \\ \text{invalid} & \text{otherwise} \end{cases} \quad (1)$$

### 3.4 OSR Decision

If the image has been classified as valid, we then apply the ODIN (Liang et al., 2018) method for the decision whether an image belongs to a known object or an unknown one. This method requires backpropagation of the gradients to create a perturbation of the inputs. We avoid performing this costly operation through all the feature extractors and only backpropagate up to the concatenation point $\mathbf{q}$, i.e. we are only perturbing the input of the linear layers in $\psi$:

$$\tilde{\mathbf{q}} = \mathbf{q} - \varepsilon \,\text{sign}\big(-\nabla_q \log \sigma_{\hat{y}}(\psi(\mathbf{q})/T)\big) \quad (2)$$

where $\sigma_i$ is the value of the softmax function at index $i$, $\hat{y} = argmax_i \sigma_i$, $T$ is the temperature hyperparameter and $\varepsilon$ is the perturbation magnitude hyperparameter. The perturbed feature vector $\tilde{\mathbf{q}}$ is again fed through $\psi$ to give the OSR decision function:

$$f_{OSR}(x) = \begin{cases} \text{known} & \text{if } max_i \sigma_i\big(\psi(\tilde{\mathbf{q}})/T\big) \geq \delta_{OSR} \\ \text{unknown} & \text{otherwise} \end{cases} \quad (3)$$

As before, the threshold $\delta_{OSR}$ should be calibrated through cross validation.

## 4 Experiments

### 4.1 Experimental Setup

For our experiments we use an image resolution of 224x224 pixels (where not otherwise noted), ResNet50 (He et al., 2016) architecture for the feature extractors and Faster-RCNN (Ren et al., 2015) with ResNet50 backbone as implemented in the Detectron2 framework (Wu et al., 2019).

#### 4.1.1 Datasets and Splits

Here we give an overview over the datasets used in our experiments and which classes we use for training or exclude for evaluation of OSR performance. As often done, we use neural networks that were pretrained on the ImageNet (Russakovsky et al., 2015) dataset as a starting point and fine-tune on the target dataset. As noticed by (Xian et al., 2018), classes which are contained in both ImageNet and the target dataset exhibit a higher performance than those only in the target dataset. Therefore, care must be taken when selecting the evaluation splits to avoid overlap with ImageNet, since we want to test on classes completely unseen by our neural network.

**Caltech-UCSD Birds-200-2011 (CUB)** (Wah et al., 2011) contains 11,788 images of 200 bird species. We train on 150 species and use the remaining 50 species for the evaluation of the OSR performance. To avoid overlap with the ImageNet dataset, we use the split proposed by (Xian et al., 2018). The dataset annotation includes up to 15 body part locations per image as keypoints which we combine to head and torso bounding boxes together with the additional annotated whole body bounding box for our strongly supervised scenario.

**Oxford-IIIT Pet Dataset (PET)** (Parkhi et al., 2012) contains 7,349 images of 37 breeds of cats and dogs. We select the 3 dog breeds and 5 cat breeds listed in table 1 because they are not contained in ImageNet as a hold-out evaluation set. The annotated head bounding box and the bounding box containing

the segmentation mask are used for the strong supervision.

| | | |
|---|---|---|
| American Bulldog | Abyssinian | Maine Coon |
| Havanese | Birman | Russian Blue |
| Shiba Inu | British Shorthair | |

Table 1: Dog and cat breeds from the PET dataset used as a hold-out set for OSR. These classes are not in ImageNet.

**Stanford Cars** (Krause et al., 2013) contains 16,185 images of 196 classes of cars. The classes have several levels of granularity, namely make, model and year. We create two splits: in the easier split we exclude makes and in the more challenging one we exclude single car models, leaving at least one model from each make in the training data. ImageNet contains several coarse "car" classes and car parts, however not subdivided into makes or even models. Therefore, we do not take additional precautions and select the hold-out sets semi-randomly as listed in the tables 2 and 3. We only evaluate weak supervision for this dataset.

| | | | | |
|---|---|---|---|---|
| Acura | Daewoo | HUMMER | Jaguar | Mitsubishi |
| Audi | Ferrari | Honda | Lincoln | Porsche |

Table 2: Car makes from the Standford Cars dataset used as a hold-out evaluation set

| | |
|---|---|
| Acura ZDX Hatchback 2012 | HUMMER H3T Crew Cab 2010 |
| Audi RS 4 Convertible 2008 | Ferrari FF Coupe 2012 |
| Audi 100 Sedan 1994 | Ferrari 458 Italia Coupe 2012 |
| Audi S4 Sedan 2012 | Honda Accord Sedan 2012 |
| BMW 1 Series Coupe 2012 | Hyundai Accent Sedan 2012 |
| BMW X3 SUV 2012 | Hyundai Azera Sedan 2012 |
| Bentley Mulsanne Sedan 2011 | Jeep Patriot SUV 2012 |
| Cadillac SRX SUV 2012 | Jeep Compass SUV 2012 |
| Chrysler Aspen SUV 2009 | Lamborghini Aventador Coupe 2012 |
| Dodge Caliber Wagon 2007 | Mercedes-Benz S-Class Sedan 2012 |
| Dodge Caravan Minivan 1997 | Nissan Leaf Hatchback 2012 |
| Dodge Charger Sedan 2012 | Suzuki SX4 Sedan 2012 |

Table 3: Car models from the Standford Cars dataset used as a hold-out evaluation set. Note that for every of those models there is at least one model from the same make in the training set.

Additionally we use the following datasets for evaluation: iNaturalist2017 (iNat17) (Van Horn et al., 2018), NABirds (Van Horn et al., 2015), Stanford Dogs (Khosla et al., 2011) and FGVC-Aircraft (FGVC) (Maji et al., 2013)

#### 4.1.2 Evaluation Metrics

We use the two standard metrics, already used by previous works:

**FPR95** False positive rate at 95% true positive rate. Since we want to accept as many positive (in-distribution) samples as possible, we search for a threshold that gives a high acceptance rate (or true positive rate (TPR)) and calculate the false positive rate for this threshold. It can be easily interpreted but is prone to small changes of the threshold.

**AUROC** Area under receiver operating characteristic. This metric is calculated by computing the FPR and TPR values at different thresholds and taking the area between the resulting curve and the x-axis. It therefore does not rely on a single threshold and is less prone to fluctuations than FPR95, giving a good general performance estimate.

### 4.2 OOD Detection Results

Table 4 shows the OOD detection performance of models trained on CUB, CAR and PET and evaluated on other datasets that do not contain birds, cars or cats and dogs, respectively. For each of these datasets we randomly choose 1000 images as negative samples and 1000 images from the training dataset (both known and unknown classes, but always unseen images) as positive samples.

We only compare to ODIN (Liang et al., 2018) because this method is generally regarded as the state of the art, which is also confirmed in our experiments and in a review in (Roady et al., 2019). There is a significant performance improvement when using the box scores for the OOD decision instead of using the ODIN score. We attribute this to the way an object detector learns: it adds a background class and subdivides an image into a grid, learning for each of the grid cells if it is background or not. This can be seen as a kind of outlier exposure similar to (Hendrycks et al., 2018), but within the same image and without additional data.

The performance for some object classes is clearly worse than for others, for example a model trained on CUB can reject cars with almost perfect certainty, whereas other animal families, such as mammals, are much more difficult to reject if not explicitly seen before. After all, they may still have similar body parts (head, eyes) that resemble those of birds.

Both strongly and weakly supervised object detection prove to be superior to the ODIN method. Moreover, strong supervision has a clear advantage over weak supervision, which is due to the better quality of the box targets and the additional body part boxes. An interesting failure case of weak supervision can be seen in the evaluation of the Arachnida and Insecta superclasses from iNat17: the FPR95 metric is with

| Training Dataset | Test Dataset | Whole Image (ODIN) | Strong Supervision | | Weak Supervision | |
|---|---|---|---|---|---|---|
| | | | Whole Object (ODIN) | Box Scores | Whole Object (ODIN) | Box Scores |
| CUB | CAR | 0.981 / 0.09 | 0.972 / 0.17 | **0.997** / <**0.01** | 0.986 / 0.07 | **0.997** / <**0.01** |
| | PET | 0.813 / 0.74 | 0.870 / 0.67 | 0.970 / 0.15 | 0.907 / 0.48 | **0.979** / **0.08** |
| | Arachnida | 0.827 / 0.69 | 0.882 / 0.58 | **0.976** / **0.13** | 0.880 / 0.58 | 0.780 / 0.51 |
| | Insecta | 0.836 / 0.64 | 0.873 / 0.59 | **0.969** / **0.19** | 0.881 / 0.58 | 0.776 / 0.51 |
| | Mammalia | 0.828 / 0.67 | 0.825 / 0.69 | **0.927** / **0.32** | 0.854 / 0.65 | 0.874 / 0.38 |
| | Plantae | 0.892 / 0.51 | 0.930 / 0.37 | **0.992** / **0.03** | 0.914 / 0.44 | 0.952 / 0.14 |
| | Protozoa | 0.871 / 0.52 | 0.899 / 0.49 | **0.994** / **0.01** | 0.875 / 0.58 | 0.947 / 0.17 |
| | Reptilia | 0.837 / 0.62 | 0.840 / 0.64 | **0.981** / **0.11** | 0.840 / 0.68 | 0.924 / 0.23 |
| PET | CUB | 0.891 / 0.51 | 0.895 / 0.52 | **0.990** / **0.05** | 0.900 / 0.50 | 0.964 / 0.13 |
| | CAR | 0.994 / 0.02 | 0.997 / <**0.01** | **0.999** / <**0.01** | 0.998 / <**0.01** | **0.999** / <**0.01** |
| CAR Makes | CUB | 0.954 / 0.23 | - | - | 0.935 / 0.40 | **0.999** / <**0.01** |
| | PET | 0.908 / 0.45 | - | - | 0.853 / 0.71 | **0.999** / <**0.01** |
| | FGVC | 0.980 / 0.12 | - | - | 0.957 / 0.27 | **0.995** / **0.01** |
| CAR Models | CUB | 0.952 / 0.30 | - | - | 0.972 / 0.14 | **0.999** / <**0.01** |
| | PET | 0.940 / 0.35 | - | - | 0.922 / 0.46 | **0.999** / <**0.01** |
| | FGVC | 0.976 / 0.15 | - | - | 0.982 / 0.10 | **0.996** / **0.02** |

Table 4: OOD detection performance for models trained on CUB, PET and CAR and evaluated on other datasets or sub-datasets from iNat17. The values represent the AUROC↑ / FPR95↓ metrics. Bold values indicate the overall best result, underlined values indicate the best result with only image-level labels.

| Training Dataset | Test Dataset | Whole Image (ODIN) | Strong Supervision | | | | Weak Supervision |
|---|---|---|---|---|---|---|---|
| | | | Whole Object | Head | Torso | Combined | Whole Object |
| CUB | Hold-out | 0.769 / 0.81 | 0.821 / 0.72 | 0.789 / 0.77 | 0.753 / 0.82 | **0.866** / **0.66** | 0.829 / 0.68 |
| CUB | NABirds | 0.718 / 0.89 | 0.786 / 0.78 | 0.800 / 0.78 | 0.733 / 0.85 | **0.841** / **0.72** | 0.772 / 0.81 |
| PET | Hold-out | 0.821 / 0.71 | 0.860 / 0.63 | 0.846 / 0.62 | - | **0.893** / **0.52** | 0.857 / 0.62 |
| PET | Dogs | 0.687 / 0.86 | 0.747 / 0.84 | 0.715 / 0.89 | - | **0.789** / **0.79** | 0.732 / 0.83 |
| CAR Makes | Hold-out | 0.899 / 0.55 | - | - | - | - | **0.943** / **0.33** |
| CAR Models | Hold-out | 0.812 / 0.69 | - | - | - | - | **0.835** / **0.62** |

Table 5: OSR performance for the 3 main datasets. The values represent the AUROC↑ / FPR95↓ metrics. Bold values indicate the overall best result, underlined values indicate the best result with only image-level labels.

| (Sub-)Dataset | Strong Supervision | Weak Supervision |
|---|---|---|
| NABirds | <0.01 | <0.01 |
| iNat2017 (Aves) | 0.14 | 0.05 |

Table 6: OOD detection performance for a model trained on CUB and evaluated on datasets that contain only images of birds i.e. there are no negative samples. The values represent the FPR↓ metric with a fixed threshold $\delta_{OOD}$ of 0.5

around 50% only slightly better than ODIN and AUROC is even worse. The disparity between those two metrics indicates a strong separation in easy and hard images within the dataset. With insects flying through the air or spiders hanging on webs, the object detector confuses them with birds. With low-scoring body part boxes these cases can still be rejected. Some common examples are illustrated in figure 3.

Table 6 shows the OOD detection performance of the same model on all-birds datasets. Since these datasets do not contain invalid images, the AUROC and FPR95 metrics cannot be computed and we resort to the FPR metric with a fixed threshold $\delta_{OOD}$ of 0.5. The degraded performance on the iNaturalist2017 dataset is mostly due to the difference in image quality: it contains many images with birds far away from the camera, whereas the images in CUB and NABirds are mostly well focused on the target.

## 4.3 OSR Results

The main results for OSR are presented in table 5. Here too, we only compare to ODIN because our method is simply a set of additions to it and we want to show that these additions are responsible for the improved performance. These changes would also be beneficial if applied to some other base method.

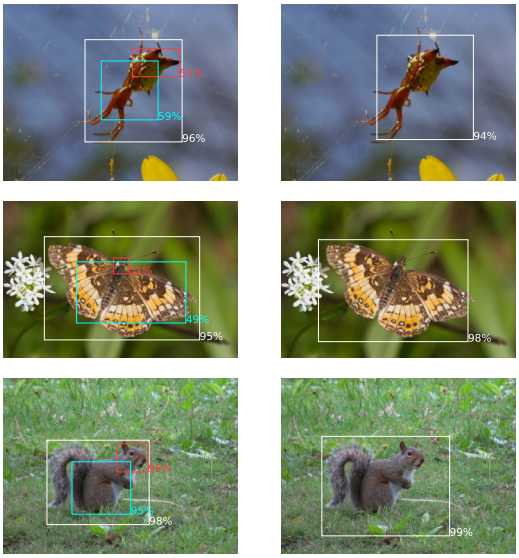



Figure 3: Illustration of common OOD detection failure cases. Left column: boxes predicted in the strongly supervised setting, right column: weakly supervised. Percentages represent the box scores as returned by the object detector. Despite being trained only on birds, the object detector is sometimes able to recognize other animals, such as spiders, butterflies or squirrels and their body parts. The whole object box (white) is often even predicted with a high confidence score. Zoom in for details.

The evaluations are performed on the hold-out splits as defined in 4.1.1 and the additional datasets NABirds and Stanford Dogs. We manually count 108 common bird species in CUB and NABirds (excluding female and juvenile birds which are mostly not present in CUB) and 18 common dog species in the PET and Dogs datasets.

We observe an improvement in performance for zooming in on the target object to get a higher resolution and an additional improvement for the combination of different body parts. The ROC curves for a model trained on CUB with strong supervision are also shown in figure 4 for better illustration. Nevertheless, with around 66% false positive rate as the best value for the CUB dataset and even worse when evaluated on NABirds, the results are still improvable. The performance disparity between the CAR makes and CAR models splits demonstrates that difficulty increases with finer granularity in the data.

In general, we notice that the OSR performance strongly correlates with the general classification accuracy for the in-distribution classes. Therefore, standard techniques that help to improve the accuracy should also be beneficial for OSR. To test this intuition, we conduct more experiments with additional data and different image resolutions. Keeping the same hold-out set, we add more data from the Caltech-UCSD Birds 200 2010 dataset (Welinder et al., 2010) which contains the same classes but is still disjoint with the 2011 version, that we use above. The results can be seen in figure 5 and mostly confirm our intuition.

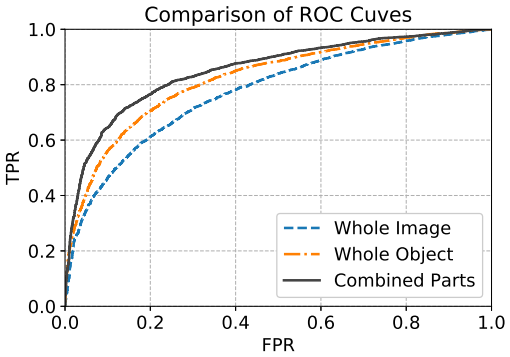



Figure 4: Comparison of ROC curves for OSR. The model was trained on the CUB dataset

## 5 Conclusion and Discussion

We have presented a framework for the detection of known, unknown and invalid classes. We have found that object detection can be an excellent choice

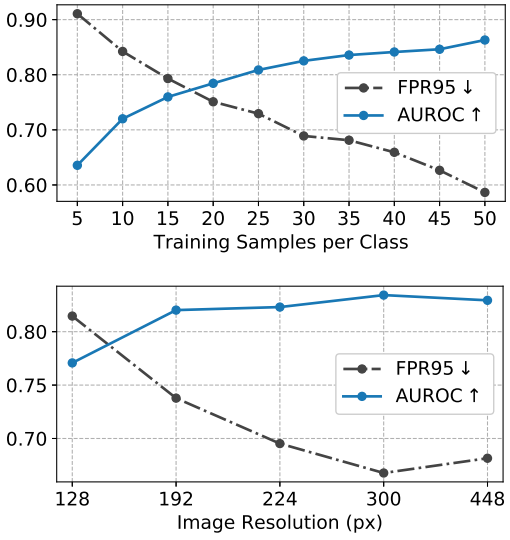



Figure 5: Influence of amount of training data (top) and different image resolutions (bottom) on the OSR performance for the weakly supervised setting and the CUB split

for the detection of invalid images, because it learns to differentiate objects and background within the same image. It can be even used with only image-level labels but improves with ground truth annotations. The resulting bounding boxes can then be used to improve the recognition of valid-but-unknown classes.

One limitation of this approach might be the requirement for object classes as apposed to non-object categories on which an object detector cannot be applied, particularly it cannot be used for non-vision machine learning problems.

Our results for fine-grained open set recognition are in no way meant to be interpreted as final but should only serve as a baseline for future comparisons. They only show the enormous difficulty of the OSR problem, especially for fine-grained data distributions. More work needs to be done in this direction.

## ACKNOWLEDGEMENTS

This work has been supported by the European Social Fund (ESF) and the Ministry of Education, Science and Culture of Mecklenburg-Vorpommern, Germany under the project "DigIT!" (ESF/14-BM-A55-0015/19).

## REFERENCES

Barrowclough, G., Cracraft, J., Klicka, J., and Zink, R. (2016). How many kinds of birds are there and why does it matter? *PLoS ONE*, 11.

Bendale, A. and Boult, T. E. (2016). Towards open set deep networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1563–1572.

Berg, T. and Belhumeur, P. N. (2013). Poof: Part-based one-vs.-one features for fine-grained categorization, face verification, and attribute estimation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 955–962.

Chen, X., Lan, X., Sun, F., and Zheng, N. (2020). A boundary based out-of-distribution classifier for generalized zero-shot learning. In *Proceedings of the European Conference on Computer Vision (ECCV)*.

Fu, J., Zheng, H., and Mei, T. (2017). Look closer to see better: Recurrent attention convolutional neural network for fine-grained image recognition. *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4476–4484.

Geifman, Y. and El-Yaniv, R. (2019). Selectivenet: A deep neural network with an integrated reject option. *arXiv preprint arXiv:1901.09192*.

Goodfellow, I. J., Shlens, J., and Szegedy, C. (2014). Explaining and harnessing adversarial examples. *arXiv preprint arXiv:1412.6572*.

Guo, C., Pleiss, G., Sun, Y., and Weinberger, K. Q. (2017). On calibration of modern neural networks. *ArXiv*, abs/1706.04599.

He, K., Zhang, X., Ren, S., and Sun, J. (2016). Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Hendrycks, D. and Gimpel, K. (2017). A baseline for detecting misclassified and out-of-distribution examples in neural networks. *ICLR*, abs/1610.02136.

Hendrycks, D., Mazeika, M., and Dietterich, T. (2018). Deep anomaly detection with outlier exposure. In *International Conference on Learning Representations*.

Khosla, A., Jayadevaprakash, N., Yao, B., and Fei-Fei, L. (2011). Novel dataset for fine-grained image categorization. In *First Workshop on Fine-Grained Visual Categorization, IEEE Conference on Computer Vision and Pattern Recognition*, Colorado Springs, CO.

Krause, J., Stark, M., Deng, J., and Fei-Fei, L. (2013). 3d object representations for fine-grained categorization. In *4th International IEEE Workshop on 3D Representation and Recognition (3dRR-13)*, Sydney, Australia.

Lee, K., Lee, K., Lee, H., and Shin, J. (2018). A simple unified framework for detecting out-of-distribution samples and adversarial attacks. In *Advances in Neural Information Processing Systems*, pages 7167–7177.

Liang, S., Li, Y., and Srikant, R. (2018). Enhancing the reliability of out-of-distribution image detection in neural networks. In *International Conference on Learning Representations*.

Lin, D., Shen, X., Lu, C., and Jia, J. (2015). Deep lac: Deep localization, alignment and classification for fine-grained recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1666–1674.

Maji, S., Kannala, J., Rahtu, E., Blaschko, M., and Vedaldi, A. (2013). Fine-grained visual classification of aircraft. Technical report.

Nguyen, A., Yosinski, J., and Clune, J. (2015). Deep neural networks are easily fooled: High con-

fidence predictions for unrecognizable images. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 427–436.

Nilsback, M.-E. and Zisserman, A. (2008). Automated flower classification over a large number of classes. In *Proceedings of the Indian Conference on Computer Vision, Graphics and Image Processing*.

Parkhi, O. M., Vedaldi, A., Zisserman, A., and Jawahar, C. V. (2012). Cats and dogs. In *IEEE Conference on Computer Vision and Pattern Recognition*.

Ren, J., Liu, P. J., Fertig, E., Snoek, J., Poplin, R., DePristo, M. A., Dillon, J. V., and Lakshminarayanan, B. (2019). Likelihood ratios for out-of-distribution detection. In *NeurIPS*.

Ren, S., He, K., Girshick, R., and Sun, J. (2015). Faster r-cnn: Towards real-time object detection with region proposal networks. In *Advances in neural information processing systems*, pages 91–99.

Roady, R., Hayes, T. L., Kemker, R., Gonzales, A., and Kanan, C. (2019). Are out-of-distribution detection methods effective on large-scale datasets? *arXiv preprint arXiv:1910.14034*.

Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., Berg, A. C., and Fei-Fei, L. (2015). ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision (IJCV)*, 115(3):211–252.

Van Horn, G., Branson, S., Farrell, R., Haber, S., Barry, J., Ipeirotis, P., Perona, P., and Belongie, S. (2015). Building a bird recognition app and large scale dataset with citizen scientists: The fine print in fine-grained dataset collection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Van Horn, G., Mac Aodha, O., Song, Y., Cui, Y., Sun, C., Shepard, A., Adam, H., Perona, P., and Belongie, S. (2018). The inaturalist species classification and detection dataset. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Wah, C., Branson, S., Welinder, P., Perona, P., and Belongie, S. (2011). The Caltech-UCSD Birds-200-2011 Dataset. Technical Report CNS-TR-2011-001, California Institute of Technology.

Welinder, P., Branson, S., Mita, T., Wah, C., Schroff, F., Belongie, S., and Perona, P. (2010). Caltech-UCSD Birds 200. Technical Report CNS-TR-2010-001, California Institute of Technology.

Wu, Y., Kirillov, A., Massa, F., Lo, W.-Y., and Girshick, R. (2019). Detectron2. https://github.com/facebookresearch/detectron2.

Xian, Y., Lampert, C. H., Schiele, B., and Akata, Z. (2018). Zero-shot learning a comprehensive evaluation of the good, the bad and the ugly. *IEEE transactions on pattern analysis and machine intelligence*, 41(9):2251–2265.

Zhang, N., Donahue, J., Girshick, R., and Darrell, T. (2014). Part-based r-cnns for fine-grained category detection. In *European conference on computer vision*, pages 834–849. Springer.

Zhou, B., Khosla, A., Lapedriza, A., Oliva, A., and Torralba, A. (2016). Learning deep features for discriminative localization. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2921–2929.

Zipf, G. K. (1932). Selected studies of the principle of relative frequency in language. *Harvard university press*.

# Bibliography

[Badrinarayanan et al., 2015] Badrinarayanan, V., Kendall, A., and Cipolla, R. (2015). Segnet: A deep convolutional encoder-decoder architecture for image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39:2481–2495.

[Bastl et al., 2017] Bastl, K., Berger, U., and Kmenta, M. (2017). Evaluation of pollen apps forecasts: The need for quality control in an ehealth service. *Journal of Medical Internet Research*, 19.

[Beery et al., 2021] Beery, S., Agarwal, A., Cole, E., and Birodkar, V. (2021). The iwildcam 2021 competition dataset. *ArXiv*, abs/2105.03494.

[Bertrand et al., 2019] Bertrand, C., Eckerter, P. W., Ammann, L., Entling, M. H., Gobet, E., Herzog, F., Mestre, L., Tinner, W., and Albrecht, M. (2019). Seasonal shifts and complementary use of pollen sources by two bees, a lacewing and a ladybeetle species in european agricultural landscapes. *Journal of Applied Ecology*.

[Birks and Berglund, 2018] Birks, H. J. B. and Berglund, B. E. (2018). One hundred years of quaternary pollen analysis 1916–2016. *Vegetation History and Archaeobotany*, 27:271–309.

[Blume-Werry et al., 2016] Blume-Werry, G., Wilson, S. D., Kreyling, J., and Milbau, A. (2016). The hidden season: growing season is 50% longer below than above ground along an arctic elevation gradient. *New Phytologist*, 209(3):978–986.

[Bridgham et al., 2006] Bridgham, S. D., Megonigal, J. P., Keller, J. K., Bliss, N. B., and Trettin, C. C. (2006). The carbon balance of north american wetlands. *Wetlands*, 26:889–916.

[Brovelli et al., 2020] Brovelli, M. A., Sun, Y., and Yordanov, V. (2020). Monitoring forest change in the amazon using multi-temporal remote sensing data and machine learning classification on google earth engine. *ISPRS Int. J. Geo Inf.*, 9:580.

[Cerda et al., 2007] Cerda, M., Hitschfeld-Kahler, N., and Mery, D. (2007). Robust tree-ring detection. In *Pacific-Rim Symposium on Image and Video Technology*.

[Charney et al., 2016] Charney, N. D., Babst, F., Poulter, B., Record, S., Trouet, V. M., Frank, D., Enquist, B. J., and Evans, M. E. K. (2016). Observed forest sensitivity to climate implies large changes in 21st century north american forest growth. *Ecology Letters*, 19(9):1119–1128.

[Cheng and Vasconcelos, 2021] Cheng, J. and Vasconcelos, N. (2021). Learning deep classifiers consistent with fine-grained novelty detection. *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1664–1673.

[Christin et al., 2019] Christin, S., Hervet, É., and Lecomte, N. (2019). Applications for deep learning in ecology. *Methods in Ecology and Evolution*, 10(10):1632–1644.

[de Geus et al., 2019] de Geus, A. R., Barcelos, C. A. Z., Batista, M. A., and da Silva, S. F. (2019). Large-scale pollen recognition with deep learning. *2019 27th European Signal Processing Conference (EUSIPCO)*, pages 1–5.

[Deng et al., 2009] Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., and Fei-Fei, L. (2009). Imagenet: A large-scale hierarchical image database. *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 248–255.

[Diffenbaugh et al., 2017] Diffenbaugh, N. S., Singh, D., Mankin, J. S., Horton, D. E., Swain, D. L., Touma, D., Charland, A., Liu, Y., Haugen, M., Tsiang, M., and Rajaratnam, B. (2017). Quantifying the influence of global warming on unprecedented extreme climate events. *Proc. Natl. Acad. Sci.*, 144(19):4881–4886.

[Dosovitskiy et al., 2020] Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M.,

Heigold, G., Gelly, S., Uszkoreit, J., and Houlsby, N. (2020). An image is worth 16x16 words: Transformers for image recognition at scale. *ArXiv*, abs/2010.11929.

[Eissenstat et al., 2000] Eissenstat, D., Wells, C., Yanai, R., and Whitbeck, J. (2000). Building roots in a changing environment: implications for root longevity. *New Phytologist*, 147(1):33–42.

[Fabijańska and Danek, 2018] Fabijańska, A. and Danek, M. (2018). Deepdendro - a tree rings detector based on a deep convolutional neural network. *Comput. Electron. Agric.*, 150:353–363.

[Fabijańska et al., 2017] Fabijańska, A., Danek, M., Barniak, J., and Piórkowski, A. (2017). Towards automatic tree rings detection in images of scanned wood samples. *Comput. Electron. Agric.*, 140:279–289.

[Fonti et al., 2010] Fonti, P., von Arx, G., García-González, I., Eilmann, B., Sass-Klaassen, U., Gärtner, H., and Eckstein, D. (2010). Studying global change through investigation of the plastic responses of xylem anatomy in tree rings. *New Phytologist*, 185(1):42–53.

[Forbes et al., 2010] Forbes, B. C., Fauria, M. M., and Zetterberg, P. (2010). Russian arctic warming and 'greening' are closely tracked by tundra shrub willows. *Global Change Biology*, 16.

[Frick et al., 2020] Frick, W. F., Kingston, T., and Flanders, J. (2020). A review of the major threats and challenges to global bat conservation. *Annals of the New York Academy of Sciences*, 1469.

[Gaggion et al., 2020] Gaggion, N., Ariel, F. D., Daric, V., Lambert, E. R., Legendre, S., Roulé, T., Camoirano, A., Milone, D. H., Crespi, M. D., Blein, T., and Ferrante, E. (2020). Chronoroot: High-throughput phenotyping by deep segmentation networks reveals novel temporal parameters of plant root system architecture. *GigaScience*, 10.

[García-Pedrero et al., 2018] García-Pedrero, Á. M., García-Cervigón, A. I., Caetano, C., Ramírez, S. C., Olano, J. M., Gonzalo-Martín, C., Lillo-Saavedra, M., and García-Hidalgo, M. (2018). Xylem vessels segmentation through a deep learning approach: a first look. *2018 IEEE International Work Conference on Bioinspired Intelligence (IWOBI)*, pages 1–9.

[García-Pedrero et al., 2019] García-Pedrero, Á. M., García-Cervigón, A. I., Olano, J. M., García-Hidalgo, M., Lillo-Saavedra, M., Gonzalo-Martín, C., Caetano, C., and Calderon-Ramirez, S. (2019). Convolutional neural networks for segmenting xylem vessels in stained cross-sectional images. *Neural Computing and Applications*, 32:17927 – 17939.

[Geirhos et al., 2020] Geirhos, R., Jacobsen, J.-H., Michaelis, C., Zemel, R. S., Brendel, W., Bethge, M., and Wichmann, F. (2020). Shortcut learning in deep neural networks. *Nature Machine Intelligence*, 2:665 – 673.

[Gillert et al., 2021] Gillert, A., Peters, B., von Lukas, U. F., and Kreyling, J. (2021). Identification and Measurement of Individual Roots in Minirhizotron Images of Dense Root Systems. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV) Workshops*, pages 1323–1331.

[Gillert et al., 2023a] Gillert, A., Peters, B., von Lukas, U. F., Kreyling, J., and Blume-Werry, G. (2023a). Tracking Growth and Decay of Plant Roots in Minirhizotron Images. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 3699–3708.

[Gillert et al., 2023b] Gillert, A., Resente, G., Anadon-Rosell, A., Wilmking, M., and von Lukas, U. (2023b). Iterative Next Boundary Detection for Instance Segmentation of Tree Rings in Microscopy Images of Shrub Cross Sections. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. Preprint: `https://arxiv.org/pdf/2212.03022.pdf`.

[Gillert and von Lukas, 2021] Gillert, A. and von Lukas, U. (2021). Towards Combined Open Set Recognition and Out-of-Distribution Detection for Fine-grained Classification. In *VISIGRAPP*.

[Girshick, 2015] Girshick, R. B. (2015). Fast r-cnn. *2015 IEEE International Conference on Computer Vision (ICCV)*, pages 1440–1448.

[Goodwin et al., 2022] Goodwin, M., Halvorsen, K. T., Jiao, L., Knausgård, K. M., Martin, A. H., Moyano, M., Oomen, R. A., Rasmussen, J. H., Sørdalen, T. K., and Thorbjørnsen, S. H. (2022). Unlocking the potential of deep learning for marine ecology: overview, applications, and outlook. *ICES Journal of Marine Science*, 79(2):319–336.

[Grigorescu et al., 2019] Grigorescu, S. M., Trasnea, B., Cocias, T. T., and Macesanu, G. (2019). A survey of deep learning techniques for autonomous driving. *Journal of Field Robotics*, 37:362 – 386.

[Guo et al., 2017] Guo, C., Pleiss, G., Sun, Y., and Weinberger, K. Q. (2017). On calibration of modern neural networks. In *International conference on machine learning*, pages 1321–1330. PMLR.

[Hart et al., 1968] Hart, P. E., Nilsson, N. J., and Raphael, B. (1968). A formal basis for the heuristic determination of minimum cost paths. *IEEE Transactions on Systems Science and Cybernetics*, 4(2):100–107.

[He et al., 2017] He, K., Gkioxari, G., Dollár, P., and Girshick, R. (2017). Mask R-CNN. In *Proceedings of the IEEE international conference on computer vision*, pages 2961–2969.

[He et al., 2015] He, K., Zhang, X., Ren, S., and Sun, J. (2015). Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. *2015 IEEE International Conference on Computer Vision (ICCV)*, pages 1026–1034.

[He et al., 2016] He, K., Zhang, X., Ren, S., and Sun, J. (2016). Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778.

[Hendrick and Pregitzer, 1996] Hendrick, R. L. and Pregitzer, K. S. (1996). Applications of minirhizotrons to understand root function in forests and other natural ecosystems. *Plant and Soil*, 185:293–304.

[Hendrycks et al., 2018] Hendrycks, D., Mazeika, M., and Dietterich, T. G. (2018). Deep anomaly detection with outlier exposure. *ArXiv*, abs/1812.04606.

[Hochreiter and Schmidhuber, 1997] Hochreiter, S. and Schmidhuber, J. (1997). Long short-term memory. *Neural Computation*, 9:1735–1780.

[Hooker, 2020] Hooker, S. (2020). The hardware lottery. *Communications of the ACM*, 64:58 – 65.

[Hyde and Williams, 1944] Hyde, H. and Williams, D. (1944). Studies in atmospheric pollen. i. a daily census of pollens at cardiff, 1942. *The New Phytologist*, 43(1):49–61.

[Iversen et al., 2011] Iversen, C. M., Murphy, M. T., Allen, M. F., Childs, J., Eissenstat, D. M., Lilleskov, E. A., Sarjala, T., Sloan, V. L., and Sullivan, P. F. (2011). Advancing the use of minirhizotrons in wetlands. *Plant and Soil*, 352:23–39.

[Jezequel et al., 2021] Jezequel, L., Vu, N.-S., Beaudet, J., and Histace, A. (2021). Fine-grained anomaly detection via multi-task self-supervision. *2021 17th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS)*, pages 1–8.

[Ji et al., 2023] Ji, W., Li, J., Bi, Q., Li, W., and Cheng, L. (2023). Segment anything is not always perfect: An investigation of sam on different real-world applications. *arXiv preprint arXiv:2304.05750*.

[Kaya et al., 2013] Kaya, Y., Erez, M. E., Karabacak, O., Kayci, L., and Fidan, M. (2013). An automatic identification method for the comparison of plant and honey pollen based on glcm texture features and artificial neural network. *Grana*, 52:71 – 77.

[Khanzhina et al., 2018] Khanzhina, N., Putin, E., Filchenkov, A., and Zamyatina, E. (2018). Pollen grain recognition using convolutional neural network. In *The European Symposium on Artificial Neural Networks*.

[Kimura et al., 1999] Kimura, K., Kikuchi, S., and ichi Yamasaki, S. (1999). Accurate root length measurement by image analysis. *Plant and Soil*, 216:117–127.

[Kirillov et al., 2023] Kirillov, A., Mintun, E., Ravi, N., Mao, H., Rolland, C., Gustafson, L., Xiao, T., Whitehead, S., Berg, A. C., Lo, W.-Y., et al. (2023). Segment anything. *arXiv preprint arXiv:2304.02643*.

[Krivek et al., 2023a] Krivek, G., Gillert, A., Harder, M., Fritze, M., Frankowski, K., Timm, L., Meyer-Olbersleben, L., von Lukas, Uwe; Kerth, G., and van Schaik, J. (2023a). BatNet: a deep learning-based tool for automated bat species identification from camera trap images.

[Krivek et al., 2023b] Krivek, G., Mahecha, E. P. N., Meier, F., Kerth, G., and van Schaik, J. (2023b). Counting in the dark: estimating population size and trends of bat assemblages at hibernacula using infrared light barriers. *Animal Conservation*.

[Krivek et al., 2022] Krivek, G., Schulze, B., Poloskei, P. Z., Frankowski, K., Mathgen, X., Douwes, A., and van Schaik, J. (2022). Camera traps with white flash are a minimally invasive method for long-term bat monitoring. *Remote Sensing in Ecology and Conservation*, 8(3):284–296.

[Krizhevsky et al., 2012] Krizhevsky, A., Sutskever, I., and Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. *Communications of the ACM*, 60:84 – 90.

[Kunz et al., 2011] Kunz, T. H., de Torrez, E. C. B., Bauer, D. M., Lobova, T. A., and Fleming, T. H. (2011). Ecosystem services provided by bats. *Annals of the New York Academy of Sciences*, 1223.

[Laliberté, 2017] Laliberté, E. (2017). Below-ground frontiers in trait-based plant ecology. *The New phytologist*, 213 4:1597–1603.

[LeCun et al., 2015] LeCun, Y., Bengio, Y., and Hinton, G. (2015). Deep learning. *Nature*, 521:436–444.

[Li and Hoiem, 2020] Li, Z. and Hoiem, D. (2020). Improving confidence estimates for unfamiliar examples. *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2683–2692.

[Lin et al., 2016] Lin, T.-Y., Dollár, P., Girshick, R. B., He, K., Hariharan, B., and Belongie, S. J. (2016). Feature pyramid networks for object detection. *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 936–944.

[Linderman et al., 2022] Linderman, R., Zhang, J., Inkawhich, N., Li, H. H., and Chen, Y. (2022). Fine-grain inference on out-of-distribution data with hierarchical classification. *ArXiv*, abs/2209.04493.

[Litjens et al., 2017] Litjens, G. J. S., Kooi, T., Bejnordi, B. E., Setio, A. A. A., Ciompi, F., Ghafoorian, M., van der Laak, J., van Ginneken, B., and Sánchez, C. I. (2017). A survey on deep learning in medical image analysis. *Medical image analysis*, 42:60–88.

[Martinez-Garcia et al., 2022] Martinez-Garcia, J., Stelzner, I., Stelzner, J., Gwerder, D., Million, S., Nelle, O., and Schuetz, P. (2022). Curvilinear-tree-ring measurements in archaeological wood samples from x-ray computed tomography. *Dendrochronologia*.

[Matamala et al., 2003] Matamala, R., Gonzàlez-Meler, M. A., Jastrow, J. D., Norby, R. J., and Schlesinger, W. H. (2003). Impacts of fine root turnover on forest npp and soil c sequestration potential. *Science*, 302(5649):1385–1387.

[McCrea et al., 2023] McCrea, R. S., King, R., Graham, L., and Börger, L. (2023). Realising the promise of large data and complex models. *Methods in Ecology and Evolution*, 14.

[Menad et al., 2019] Menad, H., Ben-Naoum, F., and Amine, A. (2019). Deep convolutional neural network for pollen grains classification. In *National Study Day on Research on Computer Sciences*.

[Mertens et al., 2009] Mertens, T., Kautz, J., and Reeth, F. V. (2009). Exposure fusion: A simple and practical alternative to high dynamic range photography. *Computer Graphics Forum*, 28.

[Miao et al., 2019] Miao, Z., Gaynor, K. M., Wang, J., Liu, Z., Muellerklein, O., Norouzzadeh, M. S., McInturff, A., Bowie, R. C., Nathan, R., Yu, S. X., et al. (2019). Insights and approaches using deep learning to classify wildlife. *Scientific reports*, 9(1):8137.

[Moeller et al., 2019] Moeller, B., Chen, H., Schmidt, T., Zieschank, A., Patzak, R., Türke, M., Weigelt, A., and Posch, S. (2019). rhizotrak: a flexible open source fiji plugin for user-friendly manual annotation of time-series images from minirhizotrons. *Plant and Soil*, 444:519 – 534.

[Narisetti et al., 2021] Narisetti, N., Henke, M., Seiler, C., Junker, A., Ostermann, J., Altmann, T., and Gladilin, E. (2021). Fully-automated root image analysis (faria). *Scientific Reports*, 11.

[Nguyen et al., 2017] Nguyen, H., Maclagan, S. J., Nguyen, T. D., Nguyen, T., Flemons, P., Andrews, K., Ritchie, E. G., and Phung, D. Q. (2017). Animal recognition and identification with deep convolutional neural networks for automated wildlife monitoring. *2017 IEEE International Conference on Data Science and Advanced Analytics (DSAA)*, pages 40–49.

[Norby et al., 2004] Norby, R. J., Ledford, J., Reilly, C. D., Miller, N. E., and O'neill, E. (2004). Fine-root production dominates response of a deciduous forest to atmospheric co2 enrichment. *Proceedings of the National Academy of Sciences of the United States of America*, 101 26:9689–93.

[Norouzzadeh et al., 2017] Norouzzadeh, M. S., Nguyen, A. M., Kosmala, M., Swanson, A., Palmer, M. S., Packer, C., and Clune, J. (2017). Automatically identifying, counting, and describing wild animals in camera-trap images with deep learning. *Proceedings of the National Academy of Sciences of the United States of America*, 115:E5716 – E5725.

[Olsson et al., 2021] Olsson, O., Karlsson, M., Persson, A. S., Smith, H. G., Varadarajan, V., Yourstone, J., and Stjernman, M. (2021). Efficient, automated and robust pollen analysis using deep learning. *Methods in Ecology and Evolution*, 12:850 – 862.

[Paszke et al., 2019] Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., Desmaison, A., Kopf, A., Yang, E., DeVito, Z., Raison, M., Tejani, A., Chilamkurthy, S., Steiner, B., Fang, L., Bai, J., and Chintala, S. (2019). PyTorch: An Imperative Style, High-Performance Deep Learning Library. In Wallach, H., Larochelle, H., Beygelzimer, A., d'Alché Buc, F., Fox, E., and Garnett, R., editors, *Advances in Neural Information Processing Systems 32*, pages 8024–8035. Curran Associates, Inc.

[Peters et al., 2023] Peters, B., Blume-Werry, G., Gillert, A., Schwieger, S., von Lukas, U. F., and Kreyling, J. (2023). As good as human experts in detecting plant roots in minirhizotron images but efficient and reproducible: the convolutional neural network "RootDetector". *Scientific Reports*, 13. https://doi.org/10.1038/s41598-023-28400-x.

[Phalempin et al., 2020] Phalempin, M., Lippold, E., Vetterlein, D., and Schlueter, S. (2020). An improved method for the segmentation of roots from x-ray computed tomography 3d images: Rootine v.2. *Plant Methods*, 17.

[Power et al., 2022] Power, C. C., Assmann, J. J., Prendin, A. L., Treier, U. A., Kerby, J. T., and Normand, S. (2022). Improving ecological insights from dendroecological studies of arctic shrub dynamics: Research gaps and potential solutions. *The Science of the total environment*, page 158008.

[Punyasena et al., 2022] Punyasena, S. W., Haselhorst, D. S., Kong, S., Fowlkes, C. C., and Moreno, J. E. (2022). Automated identification of di-

verse neotropical pollen samples using convolutional neural networks. *Methods in Ecology and Evolution*, 13:2049 – 2064.

[Rayback et al., 2012] Rayback, S. A., Henry, G. H. R., and Lini, A. (2012). Multiproxy reconstructions of climate for three sites in the canadian high arctic using cassiope tetragona. *Climatic Change*, 114:593–619.

[Ren et al., 2015] Ren, S., He, K., Girshick, R., and Sun, J. (2015). Faster R-CNN: Towards real-time object detection with region proposal networks. *Advances in neural information processing systems*, 28.

[Resente et al., 2021] Resente, G., Gillert, A., Trouillier, M., Anadon-Rosell, A., Peters, R. L., von Arx, G., von Lukas, U., and Wilmking, M. (2021). Mask, Train, Repeat! Artificial Intelligence for Quantitative Wood Anatomy. *Frontiers in Plant Science*, page 2526.

[Rixen et al., 2004] Rixen, C., Casteller, A., Schweingruber, F. H., and Stoeckli, V. A. (2004). Age analysis helps to estimate plant performance on ski pistes. *Botanica Helvetica*, 114:127–138.

[Robinson, 2004] Robinson, D. (2004). Scaling the depths: below-ground allocation in plants, forests and biomes. *Functional Ecology*, 18(2):290–295.

[Ronneberger et al., 2015] Ronneberger, O., Fischer, P., and Brox, T. (2015). U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer.

[Schweingruber et al., 1996] Schweingruber, F. H. et al. (1996). *Tree rings and environment: dendroecology.* Paul Haupt AG Bern.

[Schwieger et al., 2022] Schwieger, S., Kreyling, J., Peters, B., Gillert, A., von Lukas, U. F., Jurasinski, G., Köhn, D., and Blume-Werry, G. (2022). Rewetting prolongs root growing season in minerotrophic peatlands and mitigates negative drought effects. *Journal of Applied Ecology*.

[Seneviratne et al., 2012] Seneviratne, S. I., Nicholls, N., Easterling, D., Goodess, C. M., Kanae, S., Kossin, J., Luo, Y., Marengo, J., McInnes, K., Rahimi, M., and et al. (2012). *Changes in Climate Extremes and their*

*Impacts on the Natural Physical Environment*, pages 109–230. Cambridge University Press.

[Sevillano and Aznarte, 2018] Sevillano, V. and Aznarte, J. L. (2018). Improving classification of pollen grain images of the polen23e dataset through three different applications of deep learning convolutional neural networks. *PLoS ONE*, 13.

[Sevillano et al., 2020] Sevillano, V., Holt, K., and Aznarte, J. L. (2020). Precise automatic classification of 46 different pollen types with convolutional neural networks. *PLoS ONE*, 15.

[Smith et al., 2019] Smith, A. G., Petersen, J., Selvan, R., and Rasmussen, C. R. (2019). Segmentation of roots in soil with u-net. *Plant Methods*, 16.

[Soltaninejad et al., 2019] Soltaninejad, M., Sturrock, C. J., Griffiths, M., Pridmore, T. P., and Pound, M. P. (2019). Three dimensional root ct segmentation using multi-resolution encoder-decoder networks. *bioRxiv*.

[Stebich, 1999] Stebich, M. (1999). Palynologische untersuchungen zur vegetationsgeschichte des-weichsel-spätglazial und frühholozän an jährlich-geschichteten sedimenten des meerfelder maares (eifel).

[Steger, 1998] Steger, C. (1998). An unbiased detector of curvilinear structures. *IEEE Trans. Pattern Anal. Mach. Intell.*, 20:113–125.

[Sun et al., 2022] Sun, J., Wang, H., and Dong, Q. (2022). Spatial-temporal attention network for open-set fine-grained image recognition. *ArXiv*, abs/2211.13940.

[Swanson et al., 2015] Swanson, A., Kosmala, M., Lintott, C., Simpson, R., Smith, A., and Packer, C. (2015). Snapshot serengeti, high-frequency annotated camera trap images of 40 mammalian species in an african savanna. *Scientific data*, 2(1):1–14.

[Tabak et al., 2020] Tabak, M. A., Norouzzadeh, M. S., Wolfson, D. W., Newton, E. J., Boughton, R. K., Ivan, J. S., Odell, E. A., Newkirk, E. S., Conrey, R. Y., Stenglein, J. L., Iannarilli, F., Erb, J. D., Brook, R. K., Davis, A. J., Lewis, J. S., Walsh, D. P., Beasley, J. C., Vercauteren, K. C., Clune, J., and Miller, R. S. (2020). Improving the accessibility and transferability

of machine learning algorithms for identification of animals in camera trap images: Mlwic2. *Ecology and Evolution*, 10:10374 – 10383.

[Thomas and Davison, 2022] Thomas, R. J. and Davison, S. P. (2022). Seasonal swarming behavior of myotis bats revealed by integrated monitoring, involving passive acoustic monitoring with automated analysis, trapping, and video monitoring. *Ecology and Evolution*, 12.

[Tödtmann et al., 2020] Tödtmann, H., Vahl, M., von Lukas, U., and Ullrich, T. (2020). Time-unfolding object existence detection in low-quality underwater videos using convolutional neural networks. In *VISIGRAPP*.

[Villon et al., 2018] Villon, S., Mouillot, D., Chaumont, M., Darling, E. S., Subsol, G., Claverie, T., and Villéger, S. (2018). A deep learning method for accurate and fast identification of coral reef fishes in underwater images. *Ecological informatics*, 48:238–244.

[von Arx and Carrer, 2014] von Arx, G. and Carrer, M. (2014). Roxas – a new tool to build centuries-long tracheid-lumen chronologies in conifers. *Dendrochronologia*, 32:290–293.

[von Arx and Dietz, 2005] von Arx, G. and Dietz, H. (2005). Automated image analysis of annual rings in the roots of perennial forbs. *International Journal of Plant Sciences*, 166:723 – 732.

[von der Ohe et al., 2004] von der Ohe, W., Oddo, L. P., Piana, M., Morlot, M., and Martin, P. (2004). Harmonized methods of melissopalynology. *Apidologie*, 35.

[W3C, 2023a] W3C (2023a). *Web Neural Network API Candidate Recommendation Snapshot*. W3C. 30 March 2023.

[W3C, 2023b] W3C (2023b). *WebGPU Working Draft*. W3C. 28 March 2023.

[Wang et al., 2019] Wang, T., Rostamza, M., Song, Z., Wang, L., McNickle, G. G., Iyer-Pascuzzi, A. S., Qiu, Z., and Jin, J. (2019). Segroot: A high throughput segmentation method for root image analysis. *Comput. Electron. Agric.*, 162:845–854.

[Wilber et al., 2013] Wilber, M. J., Scheirer, W. J., Leitner, P., Heflin, B., Zott, J. P., Reinke, D., Delaney, D. K., and Boult, T. E. (2013). Animal recognition in the mojave desert: Vision tools for field biologists. *2013 IEEE Workshop on Applications of Computer Vision (WACV)*, pages 206–213.

[Willi et al., 2018] Willi, M., Pitman, R. T., Cardoso, A. W., Locke, C. M., Swanson, A., Boyer, A., Veldthuis, M., and Fortson, L. (2018). Identifying animal species in camera trap images using deep learning and citizen science. *Methods in Ecology and Evolution*, 10:80 – 91.

[Wilmking et al., 2018] Wilmking, M., Buras, A., Lehejcek, J., Lange, J., Shetti, R., and van der Maaten, E. (2018). Influence of larval outbreaks on the climate reconstruction potential of an arctic shrub. *Dendrochronologia*.

[Xu et al., 2021] Xu, W., Yu, G., Cui, Y., Gloaguen, R., Zare, A., Bonnette, J., Reyes-Cabrera, J., Rajurkar, A. B., Rowland, D., Matamala, R., Jastrow, J. D., Juenger, T., and Fritschi, F. (2021). PRMI: A dataset of minirhizotron images for diverse plant root studies. In *AI for Agriculture and Food Systems*.

[Yasrab et al., 2019] Yasrab, R., Atkinson, J. A., Wells, D. M., French, A. P., Pridmore, T. P., and Pound, M. P. (2019). Rootnav 2.0: Deep learning for automatic navigation of complex plant root architectures. *GigaScience*, 8.

[Young et al., 2017] Young, T., Hazarika, D., Poria, S., and Cambria, E. (2017). Recent trends in deep learning based natural language processing [review article]. *IEEE Computational Intelligence Magazine*, 13:55–75.

[Yu et al., 2019] Yu, G., Zare, A., Sheng, H., Matamala, R., Reyes-Cabrera, J., Fritschi, F. B., and Juenger, T. E. (2019). Root identification in minirhizotron imagery with multiple instance learning. *Machine Vision and Applications*, 31:1–13.

[Yu et al., 2020] Yu, G., Zare, A., Xu, W., Matamala, R., Reyes-Cabrera, J., Fritschi, F. B., and Juenger, T. E. (2020). Weakly supervised minirhizotron image segmentation with mil-cam. In *Proceedings of the European Conference on Computer Vision (ECCV) Workshops*.

[Zhang et al., 2023] Zhang, J., Inkawhich, N., Linderman, R., Chen, Y., and Li, H. H. (2023). Mixture outlier exposure: Towards out-of-distribution detection in fine-grained environments. *2023 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 5520–5529.

[Zhang and Suen, 1984] Zhang, T. Y. and Suen, C. Y. (1984). A fast parallel algorithm for thinning digital patterns. *Commun. ACM*, 27:236–239.

# Own Publications

- **Gillert, A.**, & Lukas, U.V. (2021). *Towards Combined Open Set Recognition and Out-of-Distribution Detection for Fine-grained Classification.* VISIGRAPP.
- Resente, G., **Gillert, A.**, Trouillier, M., Anadon-Rosell, A., Peters, R.L., von Arx, G., von Lukas, U.F., & Wilmking, M. (2021). *Mask, Train, Repeat! Artificial Intelligence for Quantitative Wood Anatomy.* Frontiers in Plant Science, 12.
- **Gillert, A.**, Peters, B., Lukas, U.V., & Kreyling, J. (2021). *Identification and Measurement of Individual Roots in Minirhizotron Images of Dense Root Systems.* 2021 IEEE/CVF International Conference on Computer Vision Workshops (ICCVW), 1323-1331.
- Schwieger, S., Kreyling, J., Peters, B., **Gillert, A.**, Freiherr von Lukas, U., Jurasinski, G., Köhn, D., & Blume-Werry, G. (2022). *Rewetting prolongs root growing season in minerotrophic peatlands and mitigates negative drought effects.* Journal of Applied Ecology.
- **Gillert, A.**, Peters, B., Lukas, U.V., Kreyling, J., & Blume-Werry, G. (2023). *Tracking Growth and Decay of Plant Roots in Minirhizotron Images.* 2023 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV), 3688-3697.
- Peters, B., Blume-Werry, G., **Gillert, A.**, Schwieger, S., von Lukas, U.F., & Kreyling, J. (2023). *As good as human experts in detecting plant roots in minirhizotron images but efficient and reproducible: the convolutional neural network "RootDetector"*. Scientific Reports, 13.

- **Gillert, A.**, Resente, G., Anadon-Rosell, A., Wilmking, M., & Lukas, U.V. (2023). *Iterative Next Boundary Detection for Instance Segmentation of Tree Rings in Microscopy Images of Shrub Cross Sections.* In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR).

- Krivek, G., **Gillert, A.**, Harder, M., Fritze, M., Frankowski, K., Timm, L., Meyer-Olbersleben, L., von Lukas, Uwe, Kerth, G., & van Schaik, J. (2023). *BatNet: a deep learning based tool for automated bat species identification from camera trap images. Remote Sensing in Ecology and Conservation.*