Image-based deep learning approaches for plant phenotyping

by

Chaoxin Wang

B.A., Capital Normal University, China, 2011

M.A., Kent State University, U.S., 2014

M.S., Kansas State University, U.S., 2020

AN ABSTRACT OF A DISSERTATION

submitted in partial fulfillment of the
requirements for the degree

Doctor of Philosophy

Department of Computer Science
Carl R. Ice College of Engineering

KANSAS STATE UNIVERSITY
Manhattan, Kansas

2022

Abstract

The genetic potential of plant traits remains unexplored due to challenges in available phenotyping methods. Deep learning could be used to build automatic tools for identifying, localizing and quantifying plant features based on agricultural images. This dissertation describes the development and evaluation of state-of-the-art deep learning approaches for several plant phenotyping tasks, including characterization of rice root anatomy based on microscopic root cross-section images, estimation of sorghum stomatal density and area based on microscopic images of leaf surfaces, and estimation of the chalkiness in rice exposed to high night temperature based on images of rice grains.

For the root anatomy task, anatomical traits such as root, stele and late metaxylem were identified using a deep learning model based on Faster Region-based Convolutional Neural Network (Faster R-CNN) with the pre-trained VGG-16 as backbone. The model was trained on root cross-section images of roots, where the traits of interest were manually annotated as rectangular bounding boxes using the LabelImg tool. The traits were also predicted as rectangular bounding boxes, which were compared with the ground truth bounding boxes in terms of intersection over union metric to evaluate the detection accuracy. The predicted bounding boxes were subsequently used to estimate root and stele diameter, as well as late metaxylem count and average diameter. Experimental results showed that the trained models can accurately detect and quantify anatomical features, and are robust to image variations. It was also observed that using the pre-trained VGG-16 network enabled the training of accurate models with a relatively small number of annotated images, making this approach very attractive in terms of adaptations to new tasks.

For estimating sorghum stomatal density and area, a deep learning approach for instance segmentation was used, specifically a Mask Region-based Convolutional Neural Network (Mask R-CNN), which produces pixel-level annotations of stomata objects. The pre-trained

ResNet-101 network was used as the backbone of the model in combination with the feature pyramid network (FPN) that enables the model to identify objects at different scales. The Mask R-CNN model was trained on microscopic leaf surface images, where the stomata objects have been manually labeled at pixel level using the VGG Image Annotator tool. The predicted stomata masks were counted, and subsequently used to estimate the stomatal area. Experimental results showed a strong correlation between the predicted counts/stomatal area and the corresponding manually produced values. Furthermore, as for the root anatomy task, this study showed that very accurate results can be obtained with a relatively small number of annotated images.

Working on the root anatomy detection and stomatal segmentation tasks showed that manually annotating data, in terms of bounding boxes and especially pixel-level masks, can be a tedious and time-consuming job, even when a relatively small number of annotated images are used for training. To address this challenge, for the task of estimating chalkiness based on images of rice grains exposed to high night temperatures, a weakly supervised approach was used, specifically, an approach based on Gradient-weighted Class Activation Mapping (Grad-CAM). Instead of performing pixel-level segmentation of the chalkiness in rice images, the weakly supervised approach makes use of high-level annotations of images as chalky or not-chalky. A convolutional neural network (e.g., ResNet-101) for binary classification is trained to distinguish between chalky and not-chalky images, and subsequently the gradients of the chalky class are used to determine a heatmap corresponding to the chalkiness area and also a chalkiness score for a grain. Experimental results on both polished and unpolished rice grains using standard instance classification and segmentation metrics showed that Grad-CAM can accurately identify chalky grains and detect the chalkiness area. The results also showed that the models trained on polished rice cannot be transferred between polished and unpolished rice, suggesting that new models need to be trained and fine-tuned for other types of rice grains and possibly images taken under different conditions.

In conclusion, this dissertation first contributes to the field of deep learning by introducing new and challenging tasks that require adaptations of existing deep learning models. It also contributes to the field of agricultural image analysis and plant phenotyping by introducing

fully automated high-throughput tools for identifying, localizing and quantifying plant traits that are of significant importance to breeding programs. All the datasets and models trained in this dissertation have been made publicly available to enable the deep learning community to use them and further advance the state-of-the-art on the challenging tasks addressed in this dissertation. The resulting tools have also been made publicly available as web servers to enable the plant breeding community to use them on images collected for tasks similar to those addressed here.

Future work will focus on the adaptation of the models used in this dissertation to other similar tasks, and also on the development of similar models for other tasks relevant to the plant breeding community, to the agriculture community at large.

Image-based deep learning approaches for plant phenotyping

by

Chaoxin Wang

B.A., Capital Normal University, China, 2011

M.A., Kent State University, U.S., 2014

M.S., Kansas State University, U.S., 2020

A DISSERTATION

submitted in partial fulfillment of the
requirements for the degree

Doctor of Philosophy

Department of Computer Science
Carl R. Ice College of Engineering

KANSAS STATE UNIVERSITY
Manhattan, Kansas

2022

Approved by:

Major Professor
Dr. Doina Caragea

Copyright

Abstract

The genetic potential of plant traits remains unexplored due to challenges in available phenotyping methods. Deep learning could be used to build automatic tools for identifying, localizing and quantifying plant features based on agricultural images. This dissertation describes the development and evaluation of state-of-the-art deep learning approaches for several plant phenotyping tasks, including characterization of rice root anatomy based on microscopic root cross-section images, estimation of sorghum stomatal density and area based on microscopic images of leaf surfaces, and estimation of the chalkiness in rice exposed to high night temperature based on images of rice grains.

For the root anatomy task, anatomical traits such as root, stele and late metaxylem were identified using a deep learning model based on Faster Region-based Convolutional Neural Network (Faster R-CNN) with the pre-trained VGG-16 as backbone. The model was trained on root cross-section images of roots, where the traits of interest were manually annotated as rectangular bounding boxes using the LabelImg tool. The traits were also predicted as rectangular bounding boxes, which were compared with the ground truth bounding boxes in terms of intersection over union metric to evaluate the detection accuracy. The predicted bounding boxes were subsequently used to estimate root and stele diameter, as well as late metaxylem count and average diameter. Experimental results showed that the trained models can accurately detect and quantify anatomical features, and are robust to image variations. It was also observed that using the pre-trained VGG-16 network enabled the training of accurate models with a relatively small number of annotated images, making this approach very attractive in terms of adaptations to new tasks.

For estimating sorghum stomatal density and area, a deep learning approach for instance segmentation was used, specifically a Mask Region-based Convolutional Neural Network (Mask R-CNN), which produces pixel-level annotations of stomata objects. The pre-trained

ResNet-101 network was used as the backbone of the model in combination with the feature pyramid network (FPN) that enables the model to identify objects at different scales. The Mask R-CNN model was trained on microscopic leaf surface images, where the stomata objects have been manually labeled at pixel level using the VGG Image Annotator tool. The predicted stomata masks were counted, and subsequently used to estimate the stomatal area. Experimental results showed a strong correlation between the predicted counts/stomatal area and the corresponding manually produced values. Furthermore, as for the root anatomy task, this study showed that very accurate results can be obtained with a relatively small number of annotated images.

Working on the root anatomy detection and stomatal segmentation tasks showed that manually annotating data, in terms of bounding boxes and especially pixel-level masks, can be a tedious and time-consuming job, even when a relatively small number of annotated images are used for training. To address this challenge, for the task of estimating chalkiness based on images of rice grains exposed to high night temperatures, a weakly supervised approach was used, specifically, an approach based on Gradient-weighted Class Activation Mapping (Grad-CAM). Instead of performing pixel-level segmentation of the chalkiness in rice images, the weakly supervised approach makes use of high-level annotations of images as chalky or not-chalky. A convolutional neural network (e.g., ResNet-101) for binary classification is trained to distinguish between chalky and not-chalky images, and subsequently the gradients of the chalky class are used to determine a heatmap corresponding to the chalkiness area and also a chalkiness score for a grain. Experimental results on both polished and unpolished rice grains using standard instance classification and segmentation metrics showed that Grad-CAM can accurately identify chalky grains and detect the chalkiness area. The results also showed that the models trained on polished rice cannot be transferred between polished and unpolished rice, suggesting that new models need to be trained and fine-tuned for other types of rice grains and possibly images taken under different conditions.

In conclusion, this dissertation first contributes to the field of deep learning by introducing new and challenging tasks that require adaptations of existing deep learning models. It also contributes to the field of agricultural image analysis and plant phenotyping by introducing

fully automated high-throughput tools for identifying, localizing and quantifying plant traits that are of significant importance to breeding programs. All the datasets and models trained in this dissertation have been made publicly available to enable the deep learning community to use them and further advance the state-of-the-art on the challenging tasks addressed in this dissertation. The resulting tools have also been made publicly available as web servers to enable the plant breeding community to use them on images collected for tasks similar to those addressed here.

Future work will focus on the adaptation of the models used in this dissertation to other similar tasks, and also on the development of similar models for other tasks relevant to the plant breeding community, to the agriculture community at large.

Table of Contents

List of Figures

List of Tables

Chapter 1

Introduction

Object detection and segmentation are two fundamental tasks in deep learning and computer vision. They have many real-life applications in domains such as autonomous driving, healthcare monitoring, video surveillance, anomaly detection, or robot vision. In crop science, object detection and segmentation are used for plant phenotyping, a process of measuring and analyzing observable plant characteristics. Usually, plant phenotyping is done using manual low-throughput methods or traditional computer vision tool kits. However, manual methods are slow, laborious and expensive, while tool kits face several challenges. The first challenge relates to the fact that existing computer vision tools come with a range of inherent biases and limitations (e.g., assumptions of artificial plant growth conditions), with none of the techniques currently available clearly standing out as a promising "blanket fit" approach (Clark et al., 2011; Durham Brooks et al., 2010; Sozzani et al., 2014).The second challenge comes from the fact that existing tools are not fully automated and require significant human effort to produce accurate results. The third challenge is that, in the current genomics era, phenotyping of traits has been identified as a substantial bottleneck compared to generating large genome sequence datasets (Hudson, 2008). To derive extensive benefit from the genetic progress achieved, deep learning tools that facilitate high-throughput phenotyping are needed.

Deep learning is an area of machine learning focused on (deep) multi-layer artificial neural networks, which can learn non-linear dependencies, and implicitly capture complex and diverse patterns in the input data (Goodfellow et al., 2016). Deep learning has been successfully used in many application domains, including computer vision, natural language processing, speech recognition, autonomous driving, among others (LeCun et al., 2015). In recent years, applications of advanced deep learning techniques to challenging problems in crop analysis have led to state-of-the-art results that outperform the results of traditional machine learning and image analysis techniques (Kamilaris and Prenafeta-Boldú, 2018). Among many successful applications of deep learning techniques in bioinformatics and computational biology, deep learning techniques have expanded the ability to accurately predict a plant phenotype (Mohanty et al., 2016). Such achievements have enabled researchers to capture a wide range of genetic diversity, a task which has been hardly possible in the past, given the amount of time and effort involved in manual analysis (Singh et al., 2016). Thus, this research aims to contribute to the plant phenotyping area and focuses on developing and applying deep learning approaches to identify, localizing, measuring the plant characteristics.

More specifically, this research focuses on the analysis of agricultural plant images and aims to build automated tools for crop physiology and plant breeding, and ultimately for farmers that may need to extract phenotype data from agricultural images.

An agricultural image is a general concept identifiable in many different formats and obtained using a variety of image acquisition tools, such as common scanners, microscopes, x-ray scanning, etc. While all image types and formats can be useful for agricultural image analysis, this research is focused on common scanner and microscopic images. The reason for focusing on images acquired by those types of equipment as opposed to x-ray or other more sophisticated equipment include: 1) the cost of image acquisition can be expensive for more sophisticated tools; 2) employ the most common formats to ensure broader applicability of the tools not just by science labs, but also by regular users that may not have access to high-end image acquisition technologies; 3) create pipelines that can be easily deployed and

used by larger groups; 4) minimize the overall pipeline development and usage time).

Leveraging images acquired using common scanners and microscopic images, this research aims to address the following problems using deep learning approaches:

1. Identify, localize, and quantify objects of interest in agricultural images.
2. Develop pipelines that can perform high-throughput automated analysis of agricultural images acquired using common imaging equipment.
3. Deploy user-friendly deep learning tools as web servers for the research community and general public.

The specific contributions of this dissertation are the following:

- **Chapter 2:** *Proposed an approach for performing root anatomy based on root cross-section image analysis using Faster R-CNN deep learning networks (Wang et al., 2019).*

  Aboveground plant efficiency has improved significantly in recent years, and the improvement has led to a steady increase in global food production. The improvement of belowground plant efficiency has potential to further increase food production. However, belowground plant roots are harder to study, due to inherent challenges presented by root phenotyping. Several tools for identifying root anatomical features in root cross-section images have been proposed. However, the existing tools are not fully automated and require significant human effort to produce accurate results. To address this limitation, we use a fully automated approach, specifically, the Faster Region-based Convolutional Neural Network (Faster R-CNN), to identify anatomical traits in root cross-section images. By training Faster R-CNN models on root cross-section images, we can detect objects such as root, stele and late metaxylem, and predict rectangular bounding boxes around such objects. Subsequently, the bounding boxes can be used to estimate the root diameter, stele diameter, late metaxylem number, and average diameter. Experimental evaluation using standard object detection metrics, such as

intersection-over-union and mean average precision, has shown that the Faster R-CNN models trained on rice root cross-section images can accurately detect root, stele and late metaxylem objects. Furthermore, the results have shown that the measurements estimated based on predicted bounding boxes have small root mean square error when compared with the corresponding ground truth values, suggesting that Faster R-CNN can be used to accurately detect anatomical features.

- **Chapter 3:** *Proposed an approach for finding the stomatal density and area in sorghum leaf images using Mask R-CNN deep networks (Bheemanahalli et al., 2021).*

  Stomatal density (SD) and stomatal complex area (SCA) are important traits that regulate gas exchange and abiotic stress response in plants. Despite sorghum (Sorghum bicolor) adaptation to arid conditions, the genetic potential of stomata-related traits remains unexplored due to challenges in available phenotyping methods. Identifying loci that control stomatal traits is fundamental to designing strategies to breed sorghum with optimized stomatal regulation. We implemented both classical and deep learning methods to characterize genetic diversity in 311 grain sorghum accessions for stomatal traits at two different field environments. Nearly 12,000 images collected from abaxial (Ab) and adaxial (Ad) leaf surfaces revealed substantial variation in stomatal traits. The study demonstrated significant agreement between manual and deep learning methods for predicting SD and SCA.

- **Chapter 4:** *Proposed a deep learning based high-throughput phenotyping of chalkiness in rice exposed to high night temperature (Wang et al., 2022).*

  Rice is a major staple food crop for more than half the world's population. As the global population is expected to reach 9.7 billion by 2050, increasing the production of high-quality rice is needed to meet the soaring demand. However, global environmental changes, especially increasingly high temperatures, can affect grain yield and quality. Heat stress is one of the major causes of an increased proportion of chalkiness in rice,

which compromises quality and, in turn, reduces the market value. Researchers have identified 140 quantitative trait loci linked to chalkiness mapped across 12 chromosomes of the rice genome. However, the available genetic information quantified by employing advances in genetics has not been adequately exploited due to a lack of a reliable, rapid and high-throughput phenotyping tool to capture chalkiness. To derive extensive benefit from the genetic progress achieved, tools that facilitate high-throughput phenotyping of rice chalkiness are needed. We use a fully automated approach based on convolutional neural networks (CNNs) augmented with Gradient-weighted Class Activation Mapping (Grad-CAM) to detect chalkiness in rice grain images. Specifically, we train a CNN model to distinguish between chalky and non-chalky grains and subsequently use Grad-CAM to identify the area of a grain that is indicative of the chalky class. The area identified by the Grad-CAM approach takes the form of a smooth heatmap that can be used to quantify the degree of chalkiness. Experimental results on both polished and unpolished rice grains using standard instance classification and segmentation metrics have shown that the Grad-CAM approach can accurately identify chalky grains and detect the chalkiness area. We have successfully demonstrated the application of a Grad-CAM based tool to accurately capture high night temperature induced chalkiness in rice. The models trained will be made publicly available. They are easy-to-use, scalable and can be readily incorporated into ongoing rice breeding programs, without rice researchers requiring computer science or machine learning expertise.

Chapter 2

Root anatomy based on root cross-section image analysis with deep learning

Abstract: Aboveground plant efficiency has improved significantly in recent years, and the improvement has led to a steady increase in global food production. The improvement of belowground plant efficiency has potential to further increase food production. However, belowground plant roots are harder to study, due to inherent challenges presented by root phenotyping. Several tools for identifying root anatomical features in root cross-section images have been proposed. However, the existing tools are not fully automated and require significant human effort to produce accurate results. To address this limitation, we use a fully automated approach, specifically, the Faster Region-based Convolutional Neural Network (Faster R-CNN), to identify anatomical traits in root cross-section images. By training Faster R-CNN models on root cross-section images, we can detect objects such as root, stele and late metaxylem, and predict rectangular bounding boxes around such objects. Subsequently, the bounding boxes can be used to estimate the root diameter, stele diameter, late metaxylem number, and average diameter. Experimental evaluation using standard

object detection metrics, such as intersection-over-union and mean average precision, has shown that the Faster R-CNN models trained on rice root cross-section images can accurately detect root, stele and late metaxylem objects. Furthermore, the results have shown that the measurements estimated based on predicted bounding boxes have small root mean square error when compared with the corresponding ground truth values, suggesting that Faster R-CNN can be used to accurately detect anatomical features. Finally, a comparison with Mask R-CNN, an instance segmentation approach, has shown that the Faster R-CNN network produces overall better results given a small training set. A webserver for performing root anatomy using the Faster R-CNN models trained on rice images, and a link to a GitHub repository containing a copy of the Faster R-CNN code are made available to the research community. The labeled images used for training and evaluating the Faster R-CNN models are also available from the GitHub repository.

Keywords: Image Analysis, Deep Learning, Object Detection, Faster R-CNN, Root Anatomy

2.1 Introduction

The crop scientific community has made significant strides in increasing global food production through advances in genetics and management, with majority of the progress achieved by improving aboveground plant efficiency (Araus et al., 2008; Bishopp and Lynch, 2015; Khush, 2013). The belowground plant roots, which provide water and nutrients for plant growth, are relatively less investigated. This is primarily because of the difficulty in accessing the roots, and the complexity of phenotyping root biology and function (Jung and Mccouch, 2013; Schmidt and Gaudin, 2017). Hence, root potential has largely been untapped in crop improvement programs (Jung and Mccouch, 2013; Schmidt and Gaudin, 2017). Over the past decade, different root phenotyping approaches have been developed for studying root architecture, including basket method for root angle (Uga et al., 2013), rhizotron method

for tracking root branching, architecture and growth dynamics (Bucksch et al., 2014), shovelomics, a.k.a., root crown phenotyping (Colombi et al., 2015), among others. Recent advances in magnetic resonance imaging and X-ray computed tomography detection systems have provided the opportunity to investigate root growth dynamics in intact plants at high temporal frequency (Mooney et al., 2012; Pfeifer et al., 2015; Schulz et al., 2013; Topp et al., 2013; van Dusschoten et al., 2016). However, each of these techniques comes with a range of inherent biases or limitations (such as artificial plant growth conditions), with none of the techniques currently available clearly standing out as a promising "blanket fit" approach (Clark et al., 2011; Durham Brooks et al., 2010; Sozzani et al., 2014). Recent non-destructive technologies, such as X-ray computed tomography, are extremely expensive, and thus beyond the reach of common crop improvement programs, in addition to not having the bandwidth to capture large genetic diversity.

Machine learning, in general, and deep neural networks (a.k.a., deep learning), in particular, are expanding the ability to accurately predict a plant phenotype (Aich and Stavness, 2017; Dobrescu et al., 2017; Kamilaris and Prenafeta-Boldú, 2018; Khan et al., 2018; Namin et al., 2017; Pound et al., 2017a; Singh et al., 2016; Tardieu et al., 2017; Ubbens and Stavness, 2017). These technological advances have enabled researchers to capture a wide range of genetic diversity, a task which has been hardly possible in the past, given the amount of time and effort involved in manual analysis. Several recent studies have used deep learning approaches for identifying and quantifying aboveground plant traits, such as the number of leaves in rosette plants, based on high-resolution RBF images (Aich and Stavness, 2017; Dobrescu et al., 2017; Ubbens and Stavness, 2017). Other investigations have focused on identifying plant diseases (Mohanty et al., 2016) or on stress phenotyping (Singh et al., 2016).

Furthermore, several prior studies have focused on data-driven approaches and tools for belowground plant phenotyping, including identifying and quantifying root morphological parameters, such as changes in root architecture, or branching and growth (Betegón-Putze et al., 2018; Delory et al., 2018; Pound et al., 2017b; Reeb et al., 2018). Such approaches

rely on standard image analysis techniques as opposed to state-of-the-art deep learning.

Both root morphological and anatomical traits are important in relation to the efficiency of soil moisture absorption by the root system. Large genetic variation in root related traits has positioned rice to uptake water and increase yields under a range of ecological conditions, including flooded and dryland conditions (Gowda et al., 2011). Root anatomical traits such as nodal root diameter (RD) (Henry et al., 2012), late metaxylem diameter (LMXD) and number (LMXN) (Comas et al., 2013; Lynch et al., 2014; Richards and Passioura, 1989), and stele diameter (SD) and its proportion to root diameter (SD:RD) (Kadam et al., 2015) have been proposed as key traits for optimized acquisition of water and productivity under water-limited conditions (Henry et al., 2012). Thin SD:RD has been used as a surrogate measure of cortex tissue area/width, which helps in the improvement of water flow and retention in vascular tissue (Kadam et al., 2015; Rieger and Litvin, 1999). Late metaxylem number and diameter along the root influence the hydraulic conductivity (Kadam et al., 2015; Richards and Passioura, 1989). These parameters mentioned above help to determine effective water use throughout the crop growth period (Lynch et al., 2014; Wasson et al., 2012).

Innovations in image acquisition technologies have made it possible to gather relatively large sets of root cross-section images, enabling studies on root anatomy. Several approaches and tools for quantifying root anatomical variation based on cross-section images have been proposed in recent years (Burton et al., 2012; Chopin et al., 2015; Lartaud et al., 2015). However, the existing tools are only partially automated, as they require user input and fine-tuning of the parameters for each specific image or for a batch of images. Fully automated tools exist for the analysis of hypocotyl cross-sections (i.e., the region in between seed leaves and roots) in the context of secondary growth (Hall et al., 2016; Sankar et al., 2014), but they are not directly applicable to the analysis of root cross-section images. Thus, there is a pressing need for automated root cross-section image analysis tools that can be used to perform root anatomy at a low cost.

To address this limitation, we have taken advantage of recent advances in deep learning

and image analysis, and used a modern, fully-automated deep learning approach, the Faster R-CNN network (Ren et al., 2015), to identify and quantify root anatomical parameters indicative of physiological and genetic responses of root anatomical plasticity in field crops. Specifically, as a proof-of-concept, we have focused on the following parameters: root diameter (RD), stele diameter (SD), late metaxylem diameter (LMXD) and late metaxylem number (LMXN), which were found important in relation to water-deficit stress in our prior work (Kadam et al., 2015, 2017). A graphical illustration of these parameters is shown in Figure 2.1.

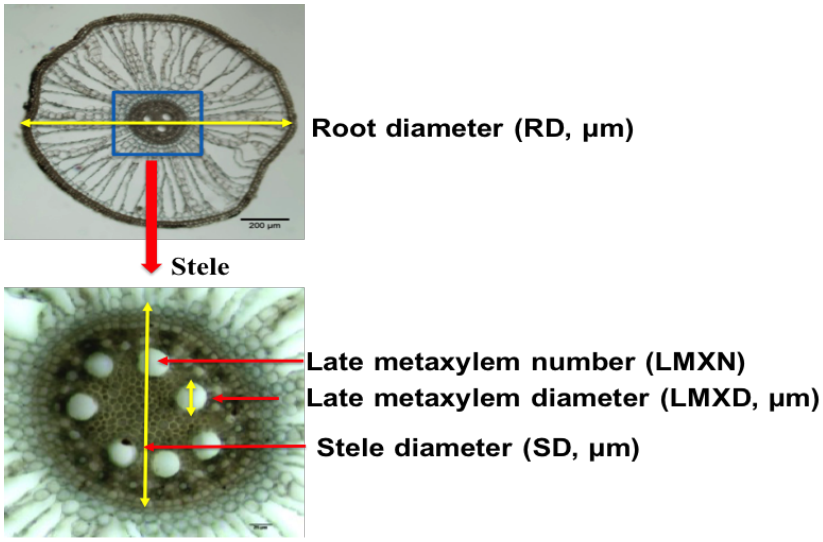


Figure 2.1: *Root anatomical traits. (Top) Root cross-section with highlighted root diameter and stele. Image taken at 50x magnification. (Bottom) Enlarged stele with highlighted stele diameter, and late metaxylem diameter. The late metaxylem number is also a trait of interest. The image was taken at 100x magnification.*

The existing Faster R-CNN model was trained on rice root cross-section images. The trained model was used to detect objects of interest in a root cross-section image (i.e., root, stele and late metaxylem), together with their corresponding bounding boxes. Subsequently, the bounding boxes were used to estimate anatomical parameters such as RD, SD, LMXD, LMXN. The Faster R-CNN model generalizes well to unseen images, thus eliminating the need for the end-user to hand-draw a stele border or manually choose or correct the

metaxylem cells, tasks that are time-consuming, and also prone to noise and errors.

To summarize, our main contributions are as follows:

- We have used the Faster R-CNN network trained on root cross-section images to detect root, stele and late metaxylem objects, and their corresponding bounding boxes.
- We have investigated the Faster R-CNN model with respect to the number of instances needed to accurately detect objects of interest, and their corresponding bounding boxes.
- We have evaluated the ability of the predicted bounding boxes to produce accurate estimates for anatomical properties, and performed error analysis to identify sources of errors.
- We have compared the results of the Faster R-CNN network (an object detection model) with results obtained using Mask R-CNN network (an instance segmentation model), and showed that the Faster R-CNN model produces better results overall, given a small training set.
- We have identified advantages and disadvantages of Faster R-CNN approach for root anatomy by comparison with existing approaches for this task.

2.2 Related work on root anatomy

There are several approaches and tools for quantifying root anatomical variation based on cross-section images (Burton et al., 2012; Chopin et al., 2015; Lartaud et al., 2015). Approaches in this category can be roughly categorized as manual, semi-automated, and automated approaches. Manual analysis of root images relies heavily on subjective assessments, and is suitable only for low throughput analysis. ImageJ (Schneider et al., 2012) is an image analysis tool that has been extensively used to manually identify and quantify root anatomical traits (Kadam et al., 2015, 2017; Yamauchi et al., 2013), given that it enables researchers

to mark objects of interest and obtain their measurements. In particular, the ImageJ software was used to acquire the ground truth (in terms of quantitative annotations) for the images used in this study, specifically, RD, SD, LMXD and LMXN measurements.

Semi-automated tools require user feedback to tune parameters for individual images in order to get accurate results. *RootScan* (Burton et al., 2012) and *PHIV-RootCell* (Lartaud et al., 2015) are semi-automated tools that identify and quantify anatomical root traits. RootScan was originally designed for analyzing maize root cross-section images. The analysis of each image involves several steps. RootScan starts by isolating the cross-section from the background using a global thresholding technique (Otsu, 1979). Subsequently, the stele is segmented based on the contrast between pixel intensities within and outside the stele. Different cells within the stele (e.g., late metaxylem) are classified based on their area according to background knowledge on root anatomy for a particular species. RootScan can detect several types of objects (including lucunae, metaxylem and protoxylem), and also a broad range of parameters for each detected object. After each step, the user has to "approve" the automated detection or alternatively correct it, before moving to the next step. The tool can be run on a set of images in batch mode, but the user still needs to provide input for each step of the analysis for each image, as explained above.

The *PHIV-RootCell* tool for root anatomy is built using the ImageJ software (Schneider et al., 2012), and provides options for selecting regions of interest (ROI) such as root, stele, xylem, and for measuring properties of these regions. It was designed for analyzing rice root cross-section images. Similar to RootScan, domain knowledge is used to identify ROIs. The PHIV-RootCell tool uploads and analyzes one image at a time, and does not have an option for batch uploading or processing. Furthermore, it requires user's supervision at each segmentation and classification step (Lartaud et al., 2015). For example, it requires the user to validate the root selection, stele selection, central metaxylem selection, among others.

As opposed to semi-automated tools that require user feedback, a fully automated approach should involve "a single click" and should produce accurate results without any

human intervention during the testing and evaluation phases. However, human input and supervision in the form of background knowledge or labeled training examples may be provided during the training phase. In this sense, *RootAnalyzer* (Chopin et al., 2015) is an automated tool, which incorporates background knowledge about root anatomy. The first step in RootAnalyzer is aimed at performing image segmentation to distinguish between root pixels (corresponding to boundaries of individual root cells) and background pixels. To achieve this, RootAnalyzer utilizes a local thresholding technique to analyze each pixel's intensity by comparing it with the mean pixel intensity in a small square neighborhood around that pixel (defined by a width parameter, W). Subsequently, RootAnalyzer constructs a difference image, and classifies pixels as root or background pixels based on a threshold, T, used on the difference image. The next step is focused on detecting root cells and closing small leaks in cell boundaries, using an interpolation approach. Finally, cells are classified in different categories, such as stele cells, cortex cells, epidermal cells, etc. based on size, shape, and position. Two thresholds are used to classify cells as small or large: a threshold, A_s, for small cells, and a threshold, A_l, for large cells. Furthermore, stele cells are classified based on an additional threshold, N, on the maximum distance from a cell to any of its nearest neighbor cells. The RootAnalyzer tool can be used for both single image processing and batch processing. Single image processing allows the user to adjust and tune parameters, and also to interact with the tool at each stage of the segmentation and classification. Batch processing requires the user to provide the parameters to be used with a specific batch of plant images. Similar to RootScan, RootAnalyzer outputs a table of area measurements and counts for regions of interest. This tool was designed for wheat and was shown to work also for maize (Chopin et al., 2015).

2.3 Methods

While there are many anatomical traits that can be identified, and measured or counted (e.g., RootScan outputs more than 20 anatomical parameters), as a proof-of-concept, we have focused on measuring the root diameter (RD), stele diameter (SD), and late metaxylem diameter (LMXD), and counting the number of late metaxylem inside the stele (LMXN). Our choice was motivated by studies by Kadam et al. Kadam et al. (2015, 2017), who showed the importance of these traits in relation to water-deficit stress, and provided the ground truth dataset for our study. The tasks that we target can be achieved with modern object detection techniques, such as Faster R-CNN (Ren et al., 2015) or Mask R-CNN (He et al., 2017), as described below. In addition to the traits of interest (RD, SD, LMXD and LMXN), other traits can be estimated based on the objects detected with our trained models (e.g., stele area, average area of the late metaxylem). Furthermore, Faster R-CNN or Mask R-CNN models can be trained to detect other objects (e.g., protoxylem objects), and their parameters, if data annotated with such objects becomes available.

2.3.1 Overview of the approach

We have used Faster R-CNN, a network for object detection, to detect objects of interest (i.e., root, stele, late metaxylem), and subsequently mark each object with a bounding box. More precisely, we have trained a Faster R-CNN model to identify the root and stele within a cross-section image, and another Faster R-CNN model to identify the late metaxylem within the stele region of a cross-section. Given the bounding box of an object, identified by the Faster R-CNN models trained on root cross-section images, we have calculated its diameter by averaging the width and height of the bounding box. The count of late metaxylem was obtained by counting the number of late metaxylem objects detected by the Faster R-CNN network.

The Faster R-CNN model architecture is shown in Figure 2.2. As can be seen, the

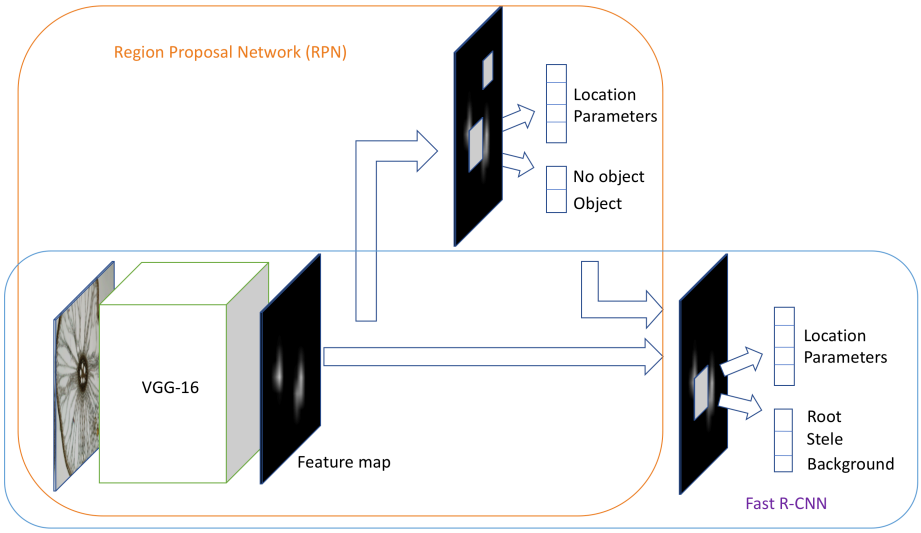


Figure 2.2: *Faster R-CNN model architecture (Ren et al., 2015), which has two main components: 1) a region proposal network (RPN), which identifies regions that may contain objects of interest and their approximate location; and 2) a Fast R-CNN network, which classifies objects as root or stele, and refines their location, defined using bounding boxes. The two components share the convolutional layers of the pre-trained VGG-16 Simonyan and Zisserman (2014a).*

Faster R-CNN has two main components. The first component consists of a Region Proposal Network (RPN), which identifies Regions of Interest (i.e., regions that may contain objects of interest), and also their location. The second component consists of a Fast R-CNN (Girshick, 2015), which classifies the identified regions (i.e., objects) into different classes (e.g., root and stele), and also refines the location parameters to generate an accurate bounding box for each detected object. The two components share the convolutional layers of the VGG-16 network (Simonyan and Zisserman, 2014a), which is used as the backbone of the Faster R-CNN model. More details on convolutional neural networks, VGG-16 and Faster R-CNN approach, which we used to detect objects and generate bounding boxes, are provided below.

In addition to the Faster R-CNN network, which focuses on object detection, we have

also explored the Mask R-CNN network, which performs more precise instance segmentation. More precisely, Mask R-CNN identifies the pixels belonging to objects of interest (in our case, root, stele and LMX). It does this by enhancing the Faster R-CNN with additional convolutional layers trained to predict instance masks for RoIs, in parallel with the object classification and bounding box regression tasks. The resulting segmentation masks identified by Mask R-CNN are used to estimate the area of the objects, and subsequently the area is used to estimate the diameter of the objects.

2.3.2 Convolutional neural networks and VGG-16

Convolutional Neural Networks (CNNs) (LeCun et al., 1989) are widely used in image analysis. While originally designed for image classification, the features extracted by CNNs are informative for other image analysis tasks, including object detection. A CNN consists of convolutional layers followed by non-linear activations, pooling layers and fully connected layers, as seen in Figure 2.3 (which shows a specific CNN architecture called VGG-16 (Simonyan and Zisserman, 2014a)).

A convolutional layer employs a sliding window approach to apply a set of filters (low-dimensional tensors) to the input image. The convolution operation captures local dependencies in the original image, and it produces a feature map. Different filters produce different feature maps, consisting of different features of the original image (e.g., edges, corners, etc.). A convolution layer is generally followed by a non-linear activation function, such as the Rectified Linear Unit (i.e., ReLU), applied element-wise to generate a rectified feature map. The ReLU activation replaces all the negative pixels in a feature map with zero values. A pooling layer is used to reduce the dimensionality of the rectified feature map. Intuitively, the pooling operation retains the most important information in a feature map by taking the maximum or the average pixel in each local neighborhood of the feature map. As a consequence, the feature map becomes invariant to scale and translation (LeCun et al., 2015).

After a sequence of convolutional layers (together with non-linear activations) and pooling

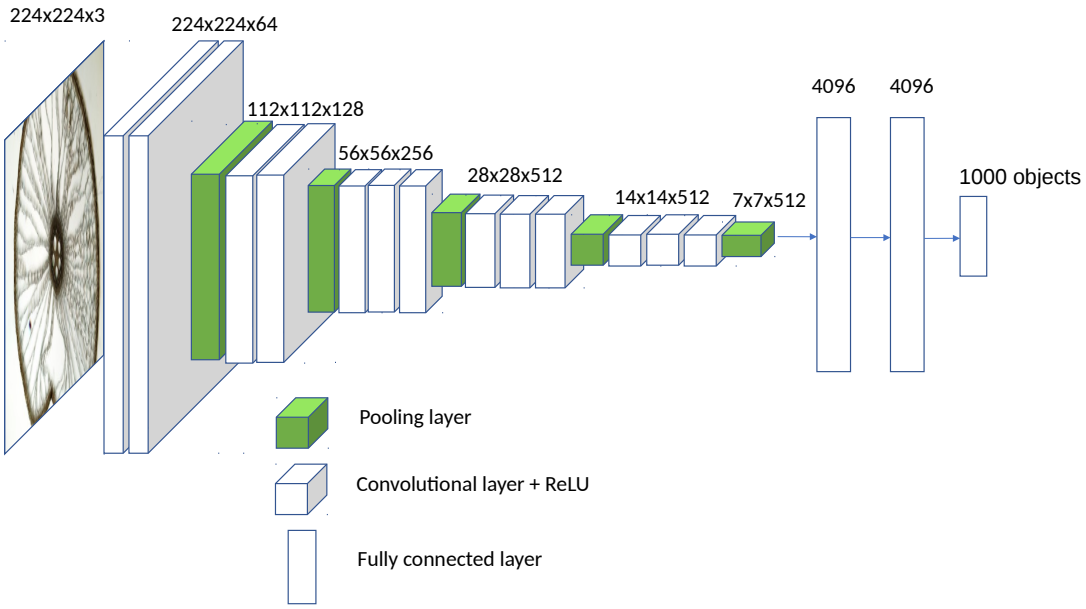


Figure 2.3: *VGG-16. The original VGG-16 architecture consists of 13 convolution+ReLU layers, five pooling layers, and three fully connected layers. A convolution+ReLU layer produces a feature map, while a pooling layer reduces the dimensionality of the feature map. The last fully connected layer uses a softmax activation function to predict one of the 1000 categories. The dimensions corresponding to each layer are also shown.*

layers, a CNN has one or more fully connected layers. In a fully connected layer, all neurons are connected to all neurons in the subsequent layer. The first fully connected layer is connected to the last downsized feature map. The fully connected layers are used to further reduce the dimensionality and to capture non-linear dependencies between features (LeCun et al., 2015). The last fully connected layer uses a softmax activation function, and has as many output neurons as the number of targeted classes.

There are several pre-trained CNN architectures available, including VGG-16 (Simonyan and Zisserman, 2014a), shown in Figure 2.3. A VGG type network, trained on 1.3 million images with 1000 categories, had the second best top-5 error (specifically, 7.3%) in ILSVRC (ImageNet Large Scale Visual Recognition Competition) in 2014. Furthermore, VGG-16 was used with good results in the original Faster R-CNN study (Ren et al., 2015), which motivated us to use it also in our study. As can be seen in Figure 2.3, VGG-16 has 13 convolutional+ReLU layers, 5 pooling layers, and 3 fully connected layers. The dimensions

corresponding to each layer are also shown in Figure 2.3.

2.3.3 Region proposal network (RPN)

As mentioned above, the region proposal network identifies regions that could potentially contain objects of interest, based on the last feature map of the pre-trained convolutional neural network that is part of the model, in our case VGG-16 (Simonyan and Zisserman, 2014a). More specifically, using a sliding window approach, k regions are generated for each location in the feature map. These regions, are represented as boxes called *anchors*. The anchors are all centered in the middle of their corresponding sliding window, and differ in terms of scale and aspect ratio (Ren et al., 2015), to cover a wide variety of objects. The region proposal network is trained to classify an anchor (represented as a lower-dimensional vector) as containing an object of interest or not (i.e., it outputs an "objectness" score), and also to approximate the four coordinates of the object (a.k.a., location parameters). The ground truth used to train the model consists of bounding boxes provided by human annotators. If an anchor has high overlap with a ground truth bounding box, then it is likely that the anchor box includes an object of interest, and it is labeled as positive with respect to the *object* versus *no object* classification task. Similarly, if an anchor has small overlap with a ground truth bounding box, it is labeled as negative. Anchors that don't have high or small overlap with a ground truth bounding box are not used to train the model. During training, the positive and negative anchors are passed as input to two fully connected layers corresponding to the classification of anchors as containing *object* or *no object*, and to the regression of location parameters (i.e., four bounding box coordinates), respectively. Corresponding to the k anchors from a location, the RPN network outputs $2k$ scores and $4k$ coordinates.

2.3.4 Fast R-CNN

Anchors for which the RPN network predicts high "objectness" scores are passed to the last two layers (corresponding to object classification and location parameter refinement, respectively) of a network that resembles the original Fast R-CNN network (Girshick, 2015), except for how the proposed regions are generated. Specifically, in the original Fast R-CNN, the regions were generated from the original image using an external region proposal method (e.g., selective search).

As opposed to the original Fast R-CNN (Girshick, 2015), in the Fast R-CNN component of the Faster R-CNN model, the external region proposal method is replaced by an internal RPN trained to identify regions of interest (Ren et al., 2015). Highly overlapping regions, potentially corresponding to the same object, can be filtered using a non-maximum suppression (NMS) threshold. A pooling layer is used to extract feature vectors of fixed length for the regions of the interest proposed by RPN. Subsequently, the feature vectors are provided as input to two fully connected layers, corresponding to the classification of the object detected and the regression of its location, respectively.

The object classification layer in Fast R-CNN uses the softmax activation, while the location regression layer uses linear regression over the coordinates defining the location as a bounding box. All parameters of the network are trained together using a multi-task loss (Girshick, 2015).

Mask R-CNN network

Mask R-CNN is a network for instance segmentation, which identifies masks enclosing the pixels that belong to instances of an object of interest, e.g., it identifies masks for instances of root, stele or LMX objects. Mask R-CNN extends the Faster R-CNN network by including additional convolutional layers trained for the task of predicting instance masks for RoIs, in parallel with the tasks performed by Faster R-CNN, specifically object classification and bounding box regression tasks. Another innovation in Mask R-CNN is to use a Feature

Pyramid Network (FPN) (Lin et al., 2017) to enable the identification of objects at different scales. Finally, Mask R-CNN replaces the RoI Pool layer in Faster R-CNN, which extracts a fixed-length feature vector from a feature map, with a RoI Align layer, which performs pixel-to-pixel alignment between network input and output, to enable the generation of precise instance masks.

Faster R-CNN and Mask R-CNN implementation and training

The original, publicly available implementation of the Faster R-CNN network Ren (2015) uses MATLAB as the programming language, and Caffe Jia) as the backend deep learning framework. Chen and Gupta Chen and Gupta (2017) provided an implementation of the Faster R-CNN network, which uses Python as the programming language and TensorFlow TensorFlow) as the backend deep learning framework. This publicly available implementation Chen (2017), allows the user to train a model from scratch and also to reuse one of several pre-trained models as the backbone of the network. In particular, the user can select the VGG-16 network, pre-trained on the ImageNet dataset with 1000 categories.

We used the Python/TensorFlow implementation of the Faster R-CNN network, with the pre-trained VGG-16 model as its backbone, and trained the network to identify objects such as root, stele and late metaxylem. More precisely, the parameters of the VGG-16 convolutional layers, which are shared by the Fast R-CNN and RPN networks in Faster R-CNN, were initialized using the pre-trained VGG-16 network. As many image features are highly transferable between different datasets, this initialization based on VGG-16 allowed us to train accurate models from a relatively small number of root cross-section labeled images. In our preliminary experimentation, we found that it is difficult to accurately detect late metaxylem at the same time with root and stele. To address this issue, we trained a Faster R-CNN model to detect root and stele from background (i.e., everything else in the image), and another Faster R-CNN model to detect late metaxylem from background. To achieve this, we changed the output layer of the original Faster R-CNN network to reflect

our classes (corresponding to the objects detected).

Given that the RPN and Fast R-CNN networks share 13 convolutional layers (initialized based on VGG-16), they were co-trained using an iterative process that alternates between fine-tuning the RPN and fine-tuning the Fast R-CNN network (with fixed proposed regions produced by RPN) (Ren et al., 2015). All the model parameters were updated using stochastic gradient descent (SGD).

For Mask R-CNN, we used a popular implementation (Matterport-Inc., 2017), based on Python 3, Keras, and TensorFlow. This implementation has the pre-trained ResNet101 (He et al., 2016) network as its default backbone. We trained three separate models to detect objects (root, stele, and LMX, respectively), and identify objects' masks (pixel-level segmentation). We changed the output layer of the original Mask R-CNN network to reflect our classes (corresponding to the objects detected). The parameters of the ResNet convolutional layers were initialized using the pre-trained ResNet network. Feature maps produced by ResNet were provided as input to the RPN network, which produced RoIs. The RoIs were subsequently provided as input to the Fast R-CNN network, extended with a component for predicting a segmentation mask for an object instance at the pixel level. Similar to the Faster R-CNN training, training of the Mask R-CNN network was based on an iterative process that alternates between fine-tuning the RPN and fine-tuning the extended Fast R-CNN network.

2.4 Dataset

Twenty-five accessions of Oryza species were grown in plastic pots (25 cm in height; 26 and 20 cm diameter at the top and bottom, respectively), filled with 6 kg of clay loam soil. Three replications per each accession were maintained under well-watered conditions and roots were sampled 60 days after sowing, to ensure fully mature roots. The roots were harvested and washed thoroughly. To obtain the cross-section images used in this study,

root samples stored in 40% alcohol were hand sectioned with a razor blade using a dissection microscope. For each of the 25 rice accessions, and for each of the three biological replicates, root samples from root-shoot junction and 6 cm from the root tip were obtained. Images of root sections were acquired with the Axioplan 2 compound microscope (Zeiss, Germany) at 50x and 100x magnification. Specifically, for each accession and each replicate, 2-3 images were taken at root-shoot junction, and 2-3 images at 6 cm from the tip of the root, at 50x and 100x magnification. Thus, an image may have two versions: a 50× magnification version, which captures the whole root diameter (top image in Figure 2.1), and a 100× magnification version, which captures only the stele diameter (bottom image in Figure 2.1). However, not all 50× images have a 100× correspondent. Precisely, there are 388 images at 50× magnification, and 339 images at 100× magnification.

For each root image, we manually measured root anatomical parameters, such as root cross-section diameter, stele diameter, late metaxylem average diameter and late metaxylem number, using the ImageJ software (Schneider et al., 2012). Specifically, root diameters were estimated using the 50× magnification images. The stele diameter, and late metaxylem average diameter and count were estimated using the 100× magnification images, if available (otherwise, the 50× magnification images were used). The manual measurements and counts constitute our ground truth to which we compared the measurements produced based on the bounding boxes detected by our trained Faster R-CNN models. Statistics about the dataset, including the minimum, maximum, average and standard deviation for RD, SD, LMXD and LMXN, are presented in Table 2.1.

Statistics	RD	SD	LMXD	LMXN
Min	354	115	15	1
Max	1352	419	65	12
Avg ± std	869± 194	216 ± 55	36 ± 8	5.4 ± 1.8

Table 2.1: *Ground Truth Statistics: minimum (Min), maximum (Max), and average together with standard deviation (Avg ± std) are shown for the ground truth measurements of RD, SD, LMXD (expressed in micrometers, μm) and LMXN (which is the count of late metaxylem objects).*

In addition to measuring root anatomical parameters, each 50× magnification image was also manually labeled by independent annotators with bounding boxes that represent root, stele, and late metaxylem, respectively, and each 100× magnification image was labeled with boxes that represent late metaxylem.

We used the LabelImg tool (Tzutalin, 2015) to perform the bounding box labeling. This tool produces annotations in the Pascal Visual Object Classes (VOC) XML format (Everingham et al., 2015), a standard format used for annotating images with rectangular bounding boxes corresponding to objects. An example of a root cross-section image annotated using the LabelImg tool (zoomed in on stele) is shown in Figure 2.4 (a), where each target object is marked using four coordinates, which determine a bounding box. The bounding boxes annotated with the LabelImg tool in the 50× and 100× magnification images constitute the ground truth to which we compared the bounding boxes of the objects detected by our models. Corresponding to the ground truth image in Figure 2.4 (a) annotated with LabelImg, Figure 2.4 (b) shows the bounding box annotations produced by our models.

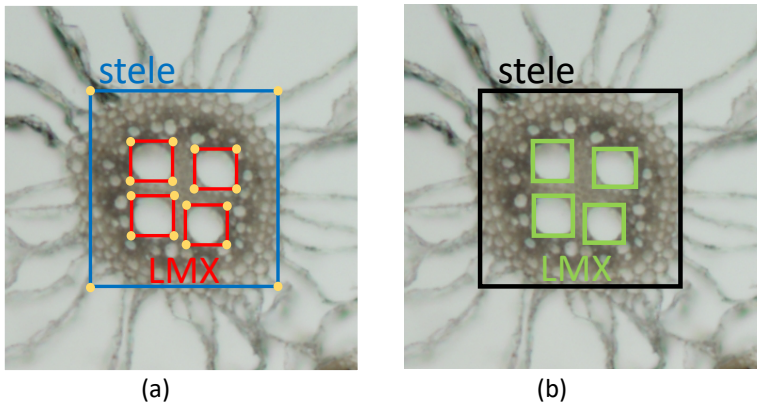


Figure 2.4: *Objects of interests as bounding boxes: (a) Ground truth image annotated using LabelImg, where each object is marked using four coordinates, which determine a bounding box. (b) The annotation of the same image by the root/stele and late metaxilem models, where the detected objects are also shown using bounding boxes.*

To produce ground truth data for Mask R-CNN, we used the VGG Image Annotator (VIA) tool (Dutta et al., 2017) with 50x magnification images. Specifically, we enclosed instances of objects of interest within polygons. The pixels contained in a polygon represent

the ground truth mask corresponding to an object instance. An example of a root cross-section image annotated using the VIA tool (zoomed in on stele) is shown in Figure 2.5 (a), where each target object is marked with a polygon representing a mask. Corresponding to the ground truth annotated with VIA tool, Figure 2.4 (b) shows the masks produced by our stele and LMX Mask R-CNN models, respectively.

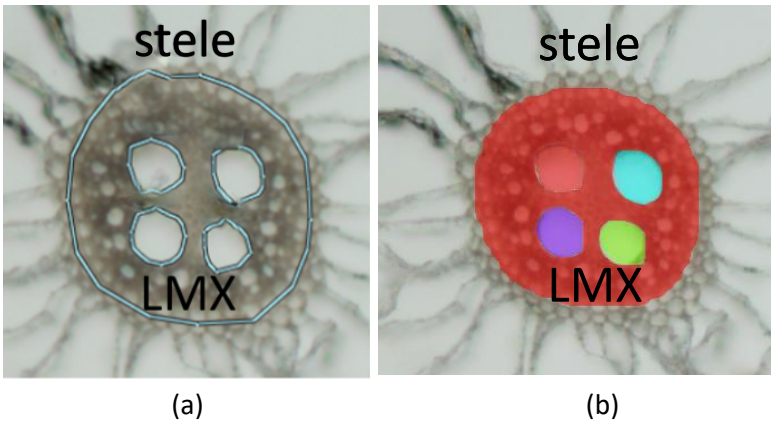


Figure 2.5: *Objects of interests as polygons/masks: (a) Ground truth stele object and LMX objects annotated using the VIA tool: the objects are marked using polygons, enclosing masks. (b) The annotation of the same objects by our models, where the detected stele and LMX objects are masked.*

We would like to emphasize that the 50× magnification images contain all the anatomical features that we target in this study, and are sufficient for training the proposed deep learning models. However, for the Faster R-CNN network, we also trained models on the 100× magnification images, independently, to understand how much the identification of the LMX objects and their measurements may be improved by using images with a higher resolution. In general, any resolution can be used for training, as long as all the features that need to be identified are contained in the image.

2.5 Experimental setup

2.5.1 Training, development and test datasets

We performed a set of experiments using 5-fold cross-validation. Specifically, we split the set of 50× magnification images into five folds, based on accessions, such that each fold contained 5 accessions out of the 25 accessions available. The exact number of 50× magnification images (instances) in each fold is shown in Table 2.2. For each fold, Table 2.2 also shows the number of corresponding 100× magnification images (instances) available (as mentioned before, not every 50× magnification image has a corresponding 100× magnification image). In each 5-fold cross-validation experiment, four folds were used for training, and the fifth fold was used for test. To tune hyper-parameters, we used one of the training folds as the development dataset. The results reported represent averages over the 5 folds. The reason for splitting the set of images based on accessions was to avoid using images from the same plant or the same replicate both in the training and test datasets.

Table 2.2: *Number of instances in each of the 5 folds used to perform cross-validation for the* 50× *and* 100× *magnification images, respectively. The total number of instances in the dataset is also shown.*

Fold	Fold 1	Fold 2	Fold 3	Fold 4	Fold 5	Total
Instances (50×)	71	79	86	77	75	388
Instances (100×)	62	60	80	69	68	339

2.5.2 Evaluation metrics

We used three standard metrics in our evaluation, driven by preliminary observations. First, given that there exist exactly one root and one stele in an image, we observed that these objects are always detected in the 50× magnification images. We used the Intersection-over-Union (IoU) metric to measure how well the predicted bounding boxes overlap with the ground truth bounding boxes. Second, given that the number of LMX objects varies between 1 and 12, and these objects are relatively small, the corresponding object detection

models are prone to both false positive and false negative mistakes. Thus, we used mean average precision (mAP), a standard metric in object detection, to evaluate the ability of our models to accurately identify the LMX objects. Both IoU and mAP metrics range between 0 and 1, and higher values are better. Finally, we used the root mean square error (RMSE) and relative root mean square error (rRMSE) (i.e., percentage error) metrics to measure the ability of the Faster R-CNN/Mask R-CNN networks to detect objects and corresponding bounding boxes that lead to root/stele/LMX diameter measurements and LMX counts close to those available as ground truth. For RMSE and rRMSE, smaller values are better.

2.5.3 Hyper-parameter tuning

Deep learning models, in general, and the Faster R-CNN/Mask R-CNN models, in particular, have many tunable hyper-parameters. We tuned several hyper-parameters shown to affect the performance of the Faster R-CNN models Zhang et al. (2016), and used the values suggested by Ren et al. Ren et al. (2015) for the other hyper-parameters. More specifically, we tuned the IoU threshold used in the RPN network to identify anchors that could potentially include an object of interest (i.e., positive instances/anchors). Furthermore, we tuned the non-maximum suppression (NMS) threshold which is used to filter region proposals produced by the trained RPN network (specifically, if two proposals have IoU larger than the NMS threshold, the two proposals will be considered to represent the same object). At last, we tuned the fraction of positive instances in a mini-batch.

The specific values that we used to tune the IoU threshold were 0.4, 0.5 and 0.6; the values used to tune the NMS threshold were 0.6, 0.7 and 0.8; and the values used to tune the fraction of positive instances in a mini-batch were 1:5 and 1:4. To observe the variation of performance with the tuned parameters, and select the values that gave the best performance, we trained a model corresponding to a particular combination of parameters on three training folds, and evaluated the performance of the model on the development fold. The performance of the models for root and stele detection was measured using the IoU metric (by comparing the

predicted bounding boxes with the ground truth bounding boxes), while the performance of the models for LMX detection was measured using the mAP metric (by comparing the detected LMX objects with the ground truth LMX objects) to ensure that the Faster R-CNN models can accurately detect all the LMX objects.

Our tuning process revealed that the performance did not vary significantly with the parameters for our object detection tasks. However, the best combination of parameters for the root/stele models consisted of the following values: 0.4 for the IoU threshold, 0.8 for the NMS threshold and 1:4 for the fraction of positive anchors in a mini-batch. The best combination of parameters for the LMX models was: 0.5 for the IoU threshold, 0.8 for the NMS threshold, and 1:4 for the fraction of positive anchors in a mini-batch. We used these combinations of values for the root/stele and LMX models, respectively, in our experiments described in the next section.

Based on our observation that Faster R-CNN performance does not vary significantly with the model hyper-parameters, we used the default values for the Mask R-CNN models.

2.6 Results

In this section, we present and discuss the results of our experiments using the Faster R-CNN models trained on rice root cross-section images. We also compare the results of the Faster R-CNN models with the results of the Mask R-CNN models. Finally, we outline time requirements for Faster R-CNN and discuss the availability of the Faster R-CNN model for root anatomy as a tool.

2.6.1 Performance of Faster R-CNN with the number of training instances

As opposed to the existing tools for identifying anatomical parameters in root cross-section images, which incorporate background knowledge about the root anatomy of a particular

species and the types of images used, the automated Faster R-CNN approach is easily generalizable to various species and types of images, given that a representative set of annotated images is provided as training data. Under the assumption that data annotation is expensive and laborious, we aim to understand how many images are necessary for good performance on roots from a particular species. Intuitively, the number of required images should be relatively small, given that our model relies on a VGG-16 network pre-trained to detect a large number of objects, generally more complex than root, stele and late metaxylem objects.

To validate our intuition, we have performed an experiment where we varied the number of images used for training, while keeping the number of test images fixed. Specifically, we used 5, 10, 25, 50, 75, 100, 150, 200, 250, and all available training images in a split, respectively, to train models for detecting the root, stele and LMX in an image. The 50× magnification images were used to train the models for root/stele/LMX. The 100× magnification images were also used to train models for LMX, with the goal of understanding the benefits provided by higher resolution images. The trained models were subsequently used to detect root, stele, and LMX objects in test images.

The performance of the models was measured by comparing the predicted objects with the ground truth objects. We used the IoU metric to evaluate the predicted bounding boxes for root/stele by comparison with the corresponding ground truth bounding boxes. We used the mAP metric to measure the ability of the models to accurately detect LMX objects. The variation of performance with the number of training images is shown in Figure 2.6 for root/stele (Left plot) and LMX (Right plot).

For models trained on the 50× magnification images, the performance increases with the number of training images. Furthermore, the left plot in the figure shows that the IoU values for both root and stele objects are around 0.95, when all the training images are used, and that the root bounding boxes are slightly better than the stele bounding boxes. Similarly, the LMX objects are detected with high accuracy, as shown in the right plot of Figure 2.6, where the mAP values are close to 0.9 consistently for models trained with smaller or larger

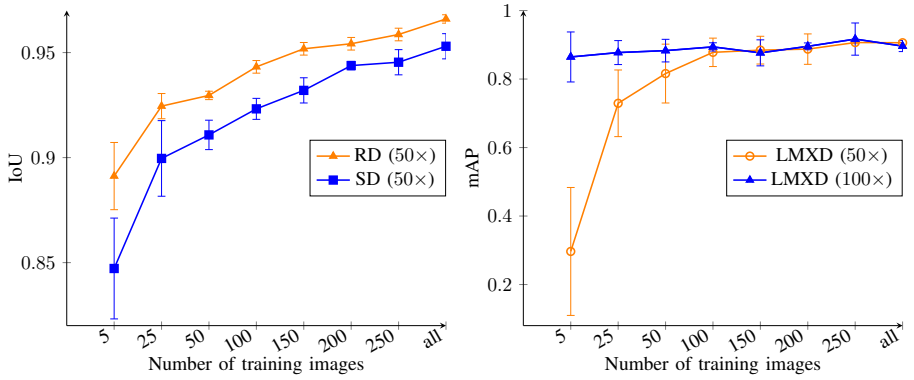
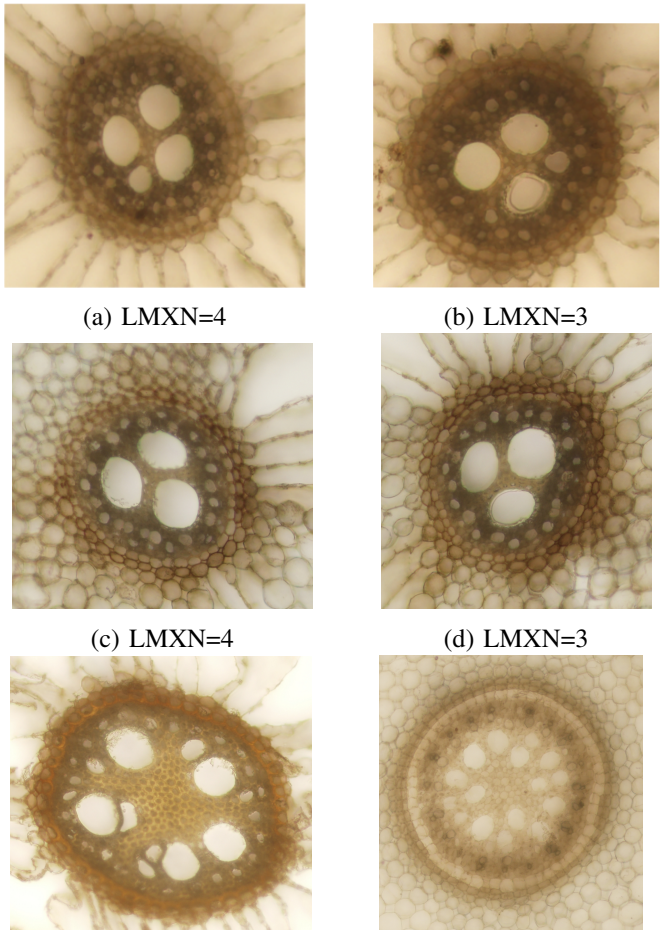


Figure 2.6: *Variation of the Faster R-CNN performance with the number of training images for root/stele detection model (Left plot), and for the LMX detection model (Right plot), respectively. We used* 50× *magnification images to detect root and stele objects, and both* 50× *and* 100× *magnification images to detect LMX. The performance of the root/stele detection model was measured using the IoU metric (which shows how accurately the predicted bounding boxes match the ground truth), while the performance of the LMX detection model was measured using the mAP metric (which shows how accurately LMX objects were detected). The plots show average values over 5 splits together with standard deviation.*

number of 100× magnification images. Similar performance is obtained with the models trained from all 50× magnification images. The plots for both root/stele and LMX also show that generally the variance decreases with the size of the data. The slow decrease in performance that is observed sometimes between two training set sizes can be explained by the addition of some inconsistently labeled images present in the original dataset. Examples of inconsistently labeled images as shown in Figure 2.7. Overall, these results support our hypothesis that only a small number of labeled images is needed to learn accurate models for the problem at hand.

2.6.2 Evaluation of Faster R-CNN performance using RMSE/rRMSE

The Faster R-CNN models trained on root images were used to detect root/stele/LMX objects in the test data. Subsequently, the detected objects were further used to calculate RD, SD, LMXD and LMXN. To evaluate the models in terms of their ability to produce accurate root/stele/LMX diameter and LMX number, we have used the RMSE error computed by



(a) LMXN=4 (b) LMXN=3

(c) LMXN=4 (d) LMXN=3

(e) LMXN=12 (f) LMXN=11

Figure 2.7: *Examples of inconsistent human annotations that are included in our ground truth dataset. Specifically, image (a) was manually labeled as having LMXN=4 (the smaller LMX was included in the count), while image (b) was labeled as having LMXN=3 (the smaller LMX was not included in the count although it has size comparable with the smaller LMX counted in (a)). Our models consistently identified 4 LMX objects in both (a) and (b) images. Similarly, image (c) was incorrectly labeled manually as having LMXN=4, while the similar image in (d) was properly labeled as having LMXN=3. Our models correctly identified 3 LMX objects in both (c) and (d) images. Finally, images (e) and (f) show a larger number of LMX which have variable size, but it is not very clear which LMX were counted by the human annotator and which were not counted to get the 12 and 11 counts, respectively. Our models identified 7 LMX objects in image (e) and 10 LMX objects in image (f).*

comparing the measurement/count estimates obtained from the predicted bounding boxes with the ground truth measurements/counts. The RD and SD measurements were evaluated based on models trained/tested with the 50× magnification images, while LMXD and LMXN were evaluated based on models trained/tested with 50× and 100× magnification images, respectively. Intuitively, the LMXD/LMXN results obtained with the models trained on the 100× magnification images should be more accurate, as those images have higher resolution. The RMSE/rRMSE results of the experiments corresponding to the five splits, together with the average over the five splits, are shown in Table 2.3. In addition, Table 2.3 shows the expected human error estimated by performing an additional manual annotation using ImageJ (similar to how the original ground truth annotation was done), and comparing the second manual annotation against the first manual annotation.

Table 2.3: *Faster R-CNN Results: RMSE (μm) and rRMSE (i.e., percentage error) results for root diameter (RD), stele diameter (SD), late metaxylem diameter (LMXD) and late metaxylem number (LMXN) for 5 splits, together with the average over the 5 splits, and also the estimates for the human error. The number of* 50× *magnification images used in these experiments is 388, while the number of* 100× *magnification images is 339. For each measurement, the magnification of the images used to train the model that produced that measurement (i.e.,* 50× *or* 100×*) is also shown.*

Split	RD(50×)		SD(50×)		LMXD(50×)		LMXD(100×)		LMXN(50×)		LMXN (100×)	
	RMSE	rRMSE	RMSE	rRMSE	RMSE	rRMSE	RMSE	rRMSE	RMSE	rRMSE	RMSE	rRMSE
Split 1	62.77	6.78	21.93	9.16	3.67	9.50	2.45	6.54	0.81	22.34	1.37	24.55
Split 2	32.18	3.94	17.54	8.32	3.77	10.53	3.13	8.18	0.71	16.55	0.45	9.17
Split 3	61.19	6.90	21.96	9.16	3.53	9.07	3.22	7.87	0.91	17.35	0.83	15.53
Split 4	33.12	3.74	20.01	9.18	3.58	11.70	3.56	10.34	1.90	30.98	0.63	11.33
Split 5	43.67	3.26	20.94	10.26	2.43	7.51	1.61	4.61	0.74	16.39	0.25	5.02
Average	46.59	4.92	20.39	9.21	3.40	9.66	2.79	7.51	1.02	20.72	0.71	13.12
Human error	48.14	5.46	25.17	11.29	3.39	9.13	3.39	9.13	0.21	3.89	0.21	3.89

As can be seen from Table 2.3, the average RMSE error for RD over the 5 splits is $46.59\mu m$, while the average rRMSE is 4.92%. Given that root diameter for the images in our dataset varies between $354\mu m$ and $1352\mu m$ (see Table 2.1), and that the RMSE estimate for human error for RD is $48.14\mu m$ (with the corresponding rRMSE being 5.46%), these results suggest that the Faster R-CNN models trained on rice images can accurately learn

to predict RD. Similarly, the average RMSE error for SD over the five splits is 20.39μm and the corresponding rRMSE is 9.21%, while the stele diameter varies between 115μm and 419μm. As for RD, the RMSE/rRMSE errors for the SD predictions are smaller than the estimates for human error, which are 25.17μm and 11.29%, respectively. As opposed to root and stele, the LMXD is significantly smaller, varying between 15μm and 65μm. In this case, the average RMSE error is 3.40μm and 2.79μm for models trained using 50× and 100× magnification images, respectively. The rRMSE for the model trained on the 50× magnification images is 9.66%, and decreases to 7.51% for the model trained on the 100× magnification images. Compared with the SD estimates for human error (which are based on the 100× magnification images, when available, or the 50× magnification images, otherwise), the results of the models trained on the 50× magnification images are slightly worse (rRMSE is 9.66% versus 9.13%), while the results of the models trained on the 100× magnification images are slightly better (7.51% versus 9.13%).

In terms of LMXN, the ground truth numbers vary between 1 and 12, with an average of 5 LMX objects per image. The average RMSE error for LMXN is 1.02 for models trained on 50× magnification images and 0.71 for models trained on 100× magnification images. Correspondingly, the rRMSE is 20.70% for models trained on 50× magnification images, and down to 13.12% for models trained on 100× magnification images. While the Faster R-CNN models trained with the 100× magnification images reduce the rRMSE error by approximately 7.5%, their average error is still higher than the estimate for human error by approximately 10%, showing that these models could be further improved with more training data.

We performed error analysis to gain insights into the usefulness of these results in practice. Specifically, we analyzed images where our models made mistakes in terms of LMXN, and observed that some of those images were annotated in an inconsistent way by the human annotators, as can be seen in Figure 2.7, where some smaller LMX objects are sometimes counted and other times not counted. This observation is not surprising, as human annotators are

prone to mistakes and inconsistencies. As opposed to that, the automated Faster R-CNN models produce more consistent results (i.e., consistently count or not count a smaller LMX). More training images are necessary to learn well in the presence of noise/inconsistencies. Nevertheless, our results suggest that the Faster R-CNN approach to root anatomy has the potential to replace the labor-intensive manual annotations of root cross-section images.

2.6.3 Comparison of Faster R-CNN with Mask R-CNN

While the results obtained using Faster R-CNN models suggest that these models can be used to replace labor-intensive manual annotations, intuitively, Mask R-CNN models can potentially improve the results further, as they perform precise instance segmentation at pixel level. To verify this intuition, we used 50x magnification images to train Mask R-CNN models for root, stele and LMX, respectively, and further used the models to identify object masks in the corresponding test images. We used the results of the LMX Mask R-CNN models to count the number of LMX in an image. The masks were used to estimate the area of each object. Finally, assuming round objects, the area was used to estimate the diameter of root, stele and LMX, respectively. We evaluated the performance on each split using the RMSE/rRMSE metrics, and compared the average RMSE/rRMSE values obtained with Mask R-CNN models over five splits, against the average RMSE/rRMSE values obtained with Faster R-CNN. Table 2.4 shows the average RMSE/rRMSE results of the Mask R-CNN and Faster R-CNN models. In addition, Table 2.4 shows the expected human error (estimated as before). The Mask R-CNN network produces better results for RD. Specifically, the rRMSE value obtained with the Mask R-CNN network is 4.14%, while the rRMSE value obtained with the Faster R-CNN value is 4.92% (and the human error is 5.46%). However, for the other measurements, SD, LMXD and LMXN, the results of the Faster R-CNN network are better. For example, the rRMSE value produced by Faster R-CNN for SD is 9.21% and the corresponding value with Mask R-CNN is 10.34%. While this result is counter-intuitive, there are several possible explanations for the observed behavior.

First, we used the VIA tool (Dutta et al., 2017) to annotate objects of interest as polygons. The polygon annotation is more labor-intensive than the bounding box annotation and also more prone to inconsistencies in annotation, especially for the smaller objects, as it is hard to mark the smaller objects very precisely as polygons. Second, to estimate the diameter from the area, we assume that all objects are round, although this not always the case. Finally, the Mask R-CNN network is an extension of the Faster R-CNN network, and thus has more parameters that need to be estimated. A larger training set may be needed to improve the results, and potentially some hyper-parameter tuning.

Table 2.4: *Mask R-CNN versus Faster R-CNN Results: average RMSE (μm) and rRMSE (i.e., percentage error) results for root diameter (RD), stele diameter (SD), late metaxylem diameter (LMXD) and late metaxylem number (LMXN) over 5 splits. In addition, the estimates for the human error are also included. The results are obtained using the 388* 50× *magnification images.*

Approach	RD(50×)		SD(50×)		LMXD(50×)		LMXN(50×)	
	RMSE	rRMSE	RMSE	rRMSE	RMSE	rRMSE	RMSE	rRMSE
Mask R-CNN	36.74	4.14	23.02	10.34	8.35	22.56	1.50	27.64
Faster R-CNN	46.59	4.92	20.39	9.21	3.40	9.66	1.02	20.72
Human error	48.14	5.46	25.17	11.29	3.39	9.13	0.21	3.89

2.6.4 Faster R-CNN robustness to image variations

We further studied the ability of the Faster R-CNN models to "adapt" to other types of root cross-section images. To do this we identified 14 images that have been used to demonstrate RootAnalyzer and 10 images that have been used to demonstrate PHIV-RootCell. In addition, we also searched the Web for root cross-section images, and identified 15 more images from rice, 9 images from maize, and 9 images labeled as monocot root cross-section images. Together, our dataset of *external* images consists of 57 heterogeneous images, which came from different species, were taken with different imaging technologies under different conditions, had different sizes and resolutions, different backgrounds, different luminosity, etc. We randomly split each category of images into training/validation and test subsets. Specifically, 42 images were used for training/validation and 15 images were used for test.

We initially used the Split 1 models (trained on 50× magnification images) to identify RD, SD, LMXD and LMXN traits for the external test images. Subsequently, we fine-tuned the Split 1 models with the external training images, and used the fine-tuned models to identify the RD, SD, LMXD and LMXN traits for the external test images. The results of these experiments are shown in Table 2.5.

Table 2.5: *Faster R-CNN model robustness to image variations. The training and test internal images correspond to the training and test subsets of Split 1. The external images are collected from the Web. We used RMSE(μm)/rRMSE(%) to compare models trained on internal images with models trained on internal and external images in terms of their ability to detect RD/SD/LMX objects (and derived their diameter) in a variety of images.*

Experiment	RD (50x)		SD (50x)		LMXD (100x)		LMXN (100x)	
	RMSE	rRMSE	RMSE	rRMSE	RMSE	rRMSE	RMSE	rRMSE
Train on internal images Test on external images	480.99	57.14	301.46	100.28	45.02	91.04	3.78	53.96
Train on internal/external images Test on external images	24.85	2.95	13.67	4.55	3.85	7.79	0.58	8.25
Train on internal images Test on internal images	62.77	6.78	21.93	9.14	3.67	9.50	0.81	22.34
Train on internal/external images Test on internal images	59.79	6.46	20.18	8.41	2.84	7.56	0.96	17.46

As can be seen in the table, out-of-the-box, the Faster R-CNN models trained on our original rice images were not very accurate on the external images. In fact, the original models could not even detect the root in 4 out of 15 images, and could not detect the stele in 7 out of 15 images, due to the differences between the external images and our internal images used for training (if an object was not detected, a 0 diameter was assigned to it). However, the fine-tuned models significantly improved the results of the original models, with rRMSE dropping from 57.14% to 2.95% for RD, from 100.28% to 4.55% for SD, from 91.04% to 7.79% for LMXD, and from 53.96% to 8.25% for LMXN. We emphasize that the high errors of the original models are generally due to the models not being able to detect some objects at all (although the error for the objects detected was relatively small). These results show that the Faster R-CNN models fine-tuned with a small number of images (specifically, 42) can learn to predict the new types of images accurately.

To ensure that the performance of the fine-tuned models was not worse than the performance of the original models on our internal images, we also tested the fine-tuned models

on the test fold corresponding to Split 1 (which was used for training). We recorded both the results of the original models and the results of the fine-tuned models in Table 2.5 (the last two rows, respectively). As can be seen, the results on our internal images improved slightly when using the fine-tuned models, as those models are more robust to variations. Specifically, rRMSE dropped from 6.78% to 6.46% for RD, from 9.14% to 8.41% for SD, from 9.50% to 7.56% for LMXD, and from 22.34% to 17.46% for LMXN. It is also interesting to note that the results of the models on the external images are better than the overall results on the internal images. One possible reason for this may be that the images found online are generally clearer images, used to illustrate root anatomy, despite the fact that they are different from our internal images.

2.7 Discussion

Given the experimental results presented in the previous section, we now discuss advantages and disadvantages of the Faster R-CNN approach by comparison with the existing approaches for root anatomy. We also discuss the potential of the Faster R-CNN network as a tool for practical annotation of cross-section root images.

2.7.1 Advantages and disadvantages of the Faster R-CNN approach

While a direct comparison between the Faster R-CNN model (trained on rice root cross-section images) and existing approaches (e.g., RootScan and RootAnalyzer) is not possible, given that each approach is trained on different species, in this section, we first outline several advantages of the Faster R-CNN model by comparison with existing models (similar advantages can be also observed for Mask R-CNN), and then emphasize several disadvantages.

Regarding the advantages, the following points can be made:

(1) For an existing tool, it is hard to find parameters that are universally good for a set of images. For example, for a given set of parameters, the segmentation result from the RootAnalyzer in Figure 2.8 shows that the parameters are appropriate for the left rice image (a) where the LMX are reasonably well identified, but not appropriate for the right rice image (b) where no LMX is identified. As opposed to that, our experiments have shown that the performance of the Faster R-CNN model does not vary much with hyper-parameters. Once a model is properly trained, it performs accurately on a variety of images.

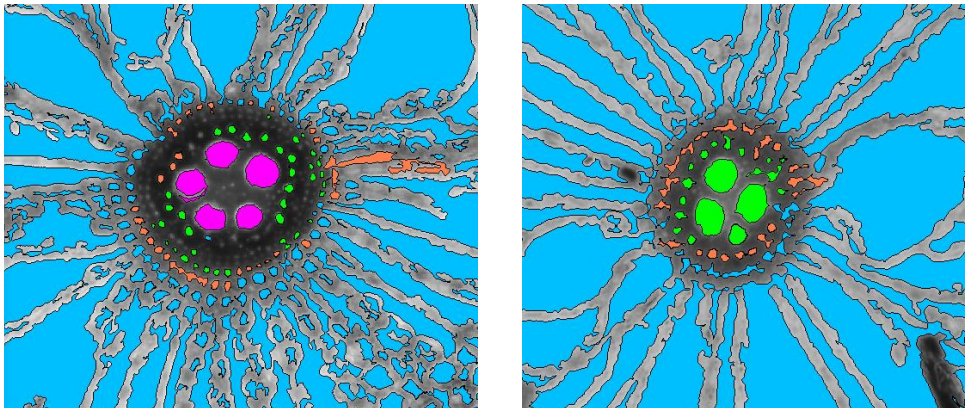


Figure 2.8: *RootAnalyzer Annotations: With the same set of parameters, in the left image the stele border (orange), endodermis (green) and late-metaxylem (purple) are detected reasonably well, while in the right image, only half of the stele border is detected. Given that the tool fails to properly detect the stele border, it also fails to detect the late metaxylem.*

(2) Plant samples used for imaging are grown in different conditions, for example in hydroponic (water based nutrient supply) or in soil, and root cross-section images are collected using different techniques (e.g., hand sectioning or sectioning using tools like vibratomes). Plant growing conditions or image acquisition differences lead to differences in image's color, contrast and brightness. As opposed to other tools, the Faster R-CNN model is not very sensitive to the light conditions or to the structure of the root cross-section images (including the epidermis thickness, epidermis transparency, and distorted cross-sections), assuming the models are trained with a variety of root

cross-section images.

(3) Each existing tool is designed based on certain pre-determined image characteristics, and may not work on images that do not exhibit those characteristics. For example, RootAnalyzer assumes a clear cell boundary and does not work for images that contain a solid boundary where the cells are not clearly identifiable. The Faster R-CNN models simply reflect the broad characteristics of the images that they are trained on, instead of being built based on pre-determined characteristics. No specific background knowledge is provided, except for what is inferred automatically from training images.

(4) Each tool is designed for a particular species, and incorporates background knowledge for that particular species. As different species may have different root anatomy, a tool designed for a species may not work for other species. For example, RootAnalyzer is designed to automatically analyze maize and wheat root cross-section images, and "may work" for other species (Chopin et al., 2015). However, the Faster R-CNN model can be easily adapted to other species, assuming some annotated training images from those species are provided. No other background knowledge is required. Along the same lines, the Faster R-CNN model can be easily adapted to images with different resolutions, assuming those images include the features of interest.

While the Faster R-CNN model presents several advantages as compared to existing approaches that incorporate background knowledge, it also has some disadvantages, as outlined below:

- We found that smaller LMX objects are not detected by the Faster R-CNN models, most probably due to inconsistencies in the training data, as illustrated in Figure 2.7. To better handle noise and improve the performance, more training data might be needed. Alternatively, more consistent ground truth should be provided.

- While the bounding boxes which mark detected objects can produce accurate results, they are not always perfectly enclosing the detected object, as it can be seen in Figure

2.4. Thus, the diameter measurements can be sometimes slightly biased, and could potentially be improved. Unfortunately, we found that the Mask R-CNN models, which can more precisely mask objects of interest, do not always provide better results possibly due to the fact that polygon annotations are time consuming, imperfect and potentially inconsistent.

- The Faster R-CNN models can only detect objects that can be marked with a bounding box. For other types of objects (e.g., aerenchyma), Mask R-CNN models are more appropriate.

2.7.2 Faster R-CNN approach as a tool for root anatomy

The Faster R-CNN model trained on our images can be used as a tool from a terminal or through a web-based application, which is also mobile friendly. The web-based application is publicly available Wang (2019b). This site is linked to a GitHub repository that contains the source code, the pre-trained Faster R-CNN models and the ground truth data. The web-based application is user-friendly and does not require any programming skills. It can be run with one of our sample images displayed on the site, or with an image uploaded by the user.

2.7.3 Time requirements

In terms of time/image requirements, our experiments have shown that accurate Faster R-CNN models can be trained from scratch with 150 to 250 images. The average time for labeling an image with LabelImg is approximately 2 minutes. The average time for training a model on an EC2 p2-xlarge instance available from Amazon Web Services (AWS) is approximately 10 hours, and does not require any human intervention during that time. Once the model is trained, the average time to annotate a new image is less than one second (using an EC2 p2-xlarge instance). If using our webserver (hosted on a local machine), the

running time for annotating a new image is approximately 9 seconds, as this includes the time to setup the virtual environment, the time to retrieve the input image from the server, the time to perform the annotation, and the time to download the image to the user's browser. Given these time requirements, assuming that a relatively large number of images need to be annotated for genetically diverse mapping populations (on the order of thousands), the human time can be potentially reduced from days or weeks (the time would take to manually annotate all images) to hours (the time may take to manually label images for training) or minutes (the time for automatically annotating images with our tool).

Furthermore, the human time for labeling images for training could be dramatically reduced to less than an hour, if one is fine-tuning the Faster R-CNN model trained on our images as opposed to training a model from scratch.

2.8 Conclusions

In this paper, we trained Faster R-CNN models on rice root cross-section images and used the trained model to perform root anatomy. The Faster R-CNN approach to root anatomy is fully automated and does not need any background knowledge, except for the implicit knowledge in images that the model is trained on. More specifically, we trained Faster R-CNN models to detect root, stele and LMX objects, and to predict bounding boxes for each detected object. Subsequently, the bounding boxes were used to obtain anatomical properties, specifically, root diameter, stele diameter, LMX diameter and LMX number. The Faster R-CNN models used had VGG-16 as a backbone, to take advantage of the extensive training of the VGG-16 network, and were fine-tuned on root cross-section images.

As part of future work, we plan to thoroughly study domain adaptation approaches that allow the transfer of knowledge from the trained rice Faster R-CNN models to models for other plant species (or for other traits), without labeling a large number of images from the other species of interest. We also aim to improve the Mask R-CNN models by

preparing more consistent ground truth annotations, larger training datasets and performing hyper-parameter tuning to understand the variation of the performance with the model hyper-parameters. Finally, we have focused mostly on performance in this research, as the bottleneck in the root anatomy pipeline lies in the image acquisition step, rather than the network training/inference steps. However, it is of interest to explore efficient approaches, such as MobileNet (Howard et al., 2017) and its variants, and NasNet (Zoph et al., 2018), and compare them with Faster R-CNN/Mask R-CNN models, both in terms of efficiency and performance.

The image datasets used in this study is available on GitHub Wang (2019a).

Chapter 3

Classical phenotyping and deep learning concur on genetic control of stomatal density and area in sorghum

Abstract: Stomatal density (SD) and stomatal complex area (SCA) are important traits that regulate gas exchange and abiotic stress response in plants. Despite sorghum (Sorghum bicolor) adaptation to arid conditions, the genetic potential of stomata-related traits remains unexplored due to challenges in available phenotyping methods. Identifying loci that control stomatal traits is fundamental to designing strategies to breed sorghum with optimized stomatal regulation. We implemented both classical and deep learning methods to characterize genetic diversity in 311 grain sorghum accessions for stomatal traits at two different field environments. Nearly 12,000 images collected from abaxial (Ab) and adaxial (Ad) leaf surfaces revealed substantial variation in stomatal traits. Our study demonstrated significant accuracy between manual and deep learning methods in predicting SD and SCA.

3.1 Introduction

Stomatal characteristics, including SD and SCA have been studied using manual low throughput methods in crops exposed to different environments (Gitz and Baker, 2009). However, the genetic architecture controlling stomatal traits and their responses to different environments is not known in sorghum. In addition, the diversity in stomatal traits is largely unexplored or utilized in breeding programs due to a cumbersome phenotyping protocol, which requires substantial investment of resources. For example, manual phenotyping of stomatal count involves obtaining stomatal imprints, imaging of the specimen, and manual counting of stomatal numbers, with the latter requiring most time and effort (Fetter et al., 2019; Sakoda et al., 2019). Sorghum is generally grown in arid and semi-arid regions, and hence its productivity depends on timing and amount of rainfall. This poses a crucial challenge to sorghum grown in USA, Sub-Saharan Africa, India, and other regions in the world (Leff et al., 2004). Despite their adaptation to arid conditions, sorghum hybrids are shown to be susceptible to harsh environments during different stages of the crop growth (Tack et al., 2017). Given that C_4 crops including sorghum have evolved and adapted to hot and arid conditions (Osborne and Freckleton, 2009), they provide an excellent opportunity to investigate natural variability in SD and area under field conditions. To date, there has not been an attempt to map the genetic loci associated with stomatal traits using the grain sorghum association panel (SAP). Thus, we hypothesized that integration of physiology, deep learning, and genomic approaches would help us understand the genetic architecture of stomatal traits in grain sorghum.

In this study, we characterized the genetic variation for stomatal traits using SAP in two environments in Kansas, USA. Additionally, we integrated the high-throughput deep learning tools and classical phenotyping methods to map genomic regions associated with stomatal number and area. Specific objectives were to (1) develop, test, and validate a fully automated deep learning tool for high-throughput phenotyping of Ab and Ad SD and SCA on a diversity panel; (2) comparative assessment of the stomatal traits obtained with deep

learning (predicted) and manual methods;

The ultimate goal of this study was to characterized the genetic variation for stomatal traits using SAP in two environments in Kansas, USA. Towards this goal, we integrated the high-throughput deep learning tools and classical phenotyping methods to map genomic regions associated with stomatal number and area. Specific objectives were to (1) develop, test, and validate a fully automated deep learning tool for high-throughput phenotyping of Ab and Ad SD and SCA on a diversity panel; (2) comparative assessment of the stomatal traits obtained with deep learning (predicted) and manual methods.

3.2 Related work

In the current genomic era, phenotyping of traits has been identified as a substantial bottleneck compared to generating large genome sequence datasets (Hudson, 2008). Recently, several computer vision-based automated phenotyping tools have been developed to overcome this challenge by automated detection of stomata, including Cascade object detection algorithm (Duarte et al., 2017; Higaki et al., 2014; Jayakody et al., 2017; Laga et al., 2014), AlexNet-based deep convolutional neural network (Fetter et al., 2019) and You Only Look Once (Casado and Heras, 2018). Several approaches and tools for quantifying stomatal variations based on images have been proposed (Dittberner et al., 2018; Sakoda et al., 2019). However, previous methods have followed the object detection approach instead of the more precise semantic object segmentation. To address this limitation, we trained the Mask Region-based Convolutional Neural Network (Mask R-CNN) algorithm to automatically predict labels for future images, to segment the stomata in an image to identify and count stomata, and to determine the SCA.

3.3 Methodology

3.3.1 Mask R-CNN model

We used a fully automated deep learning method, called Mask R-CNN, to perform stomata instance segmentation for each input image, i.e. to identify the pixels corresponding to stomata in an image. Mask R-CNN (Figure 3.1 C) is an extension of the Faster R-CNN approach (Ren et al., 2015). Similar to the Faster R-CNN network, Mask R-CNN can be trained to detect objects of interest (e.g., stomata) in an image and to localize the objects detected using bounding boxes. In addition, Mask R-CNN generates a precise segmentation mask for each object instance. The Faster R-CNN network has two main components, which share a backbone feature extractor CNN, such as ResNet (He et al., 2016). The first component, called a Region Proposal Network (RPN), uses the last feature map produced by the backbone CNN to identify regions of interest (RoI), i.e. fragments of the image (called anchors) that may contain target objects and initial approximate bounding boxes for those objects. The second component consists of fully connected layers that classify RoI proposed by the RPN network into specific categories (an object classification task) and refine the corresponding bounding box coordinates (a bounding box regression task). Mask R-CNN extends the Faster R-CNN network by including additional convolutional layers trained to predict instance masks for RoI (an instance segmentation task), in parallel with the object classification and bounding box regression tasks. Furthermore, Mask R-CNN uses a Feature Pyramid Network (Lin et al., 2017) together with ResNet as the architectural backbone to enable the identification of objects at different scales. It also replaces the RoI Pool layer in Faster R-CNN, which extracts a fixed-length feature vector from a feature map, with a RoI Align layer, which performs pixel-to-pixel alignment between network input and output, to enable the generation of precise instance masks. We used the implementation of Mask R-CNN, available at https://github.com/matterport/Mask_RCNN, with ResNet101 as the backbone network (together with FPN). We changed the original Mask RCNN architecture

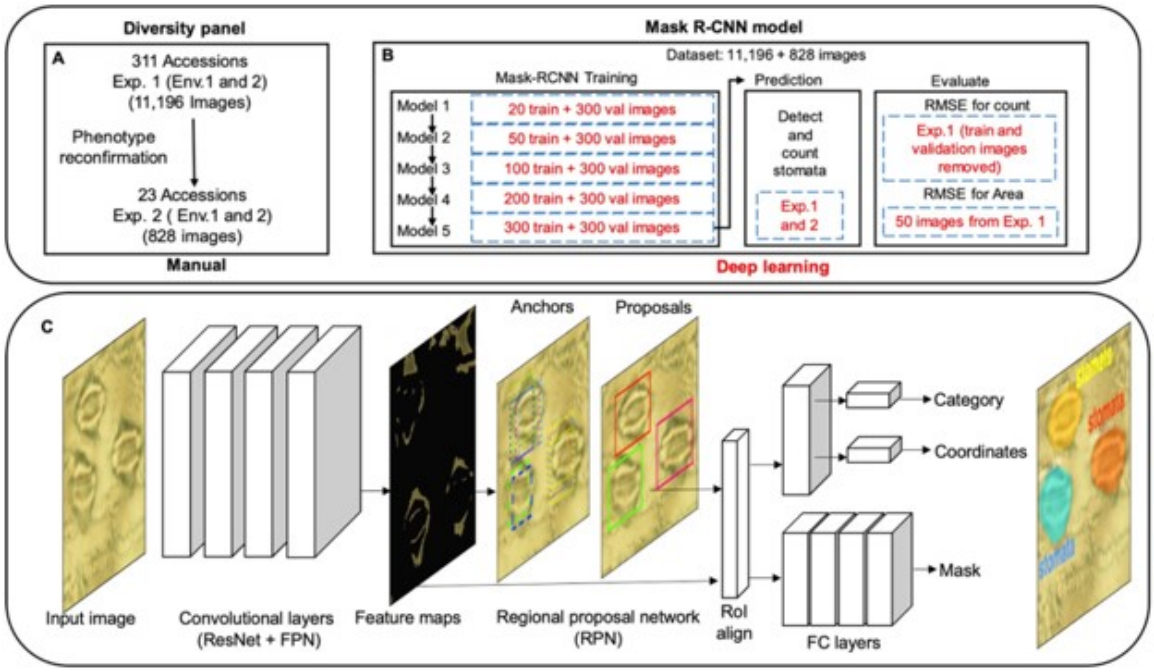


Figure 3.1: *Schematic overview of the study. A, Phenotyping of the SAP for SD and SCA in two environments (Env. 1—Manhattan and Env. 2—Hays) for two years (Exp. 1 in 2017 and Exp. 2 in 2018). B, Mask R-CNN models trained for predicting Ab and Ad stomatal number and complex area. Train and validate (val) images indicate the number of images used for training and validating the Mask R-CNN model trained. C, Mask R-CNN, a deep learning framework for stomata instance segmentation and stomata count. The network architecture contains convolutional layers (left) and fully connected layers (right), shown as rectangular cuboids in the figure. The size of each cuboid indicates the dimensionality of the corresponding layer. The connections between layers are represented through arrows.*

to customize it to our categories (stomata and background) used for the object classification and instance segmentation tasks.

3.3.2 Phenotypic data analyses

All the phenotypic traits collected were analyzed using analysis of variance (ANOVA) to test the effect of genotype (G), environment (E), and their interaction using GenStat (18th Edition, http://www.vsni.co.uk). The PCA was performed in XLSTAT. The chart.Correlation () function within the R package "Performance Analytics" was used to generate the correlation scatter plot. The H2 of all the measured traits was estimated considering the proportion of phenotypic variance that is due to the genetic variance.

3.3.3 Data collection

The SAP consisting of 311 accessions was assembled from 25 countries representing major sorghum growing regions of the world (Casa et al., 2008; Morris et al., 2013). The SAP consisted of five grain sorghum races, (namely caudatum, bicolor, guinea, durra, and kafir), intermediate races, converted lines, and elite accessions of historical and geographic importance (Harlan and Wet, 1972). In experiment 1 (Exp. 1 in 2017), the SAP was grown in two different environments (Env. 1: Kansas State University, North Farm, Manhattan and Env. 2: Agricultural Research Centre at Hays, Kansas) in a randomized block design with two replications per accession per environment. All 311 accessions were planted in a single row plot of 6.1-m long, with 0.7-m spacing between rows. Approximately 50 seeds were sown per row for each accession. Three representative plants in the middle of the row, for each accession, were tagged for studying the natural variation in SD, SCA and SinLA. All measurements were recorded 62 d after planting in both environments (Env. 1 and 2). In experiment 2 (Exp. 2 in 2018), to reconfirm the expression of the trait, candidate accessions carrying the contrasting allelic haplotypes for Ab SD with similar SCA were planted in the same environments (Env. 1 and Env. 2) in 2018. Sixty-eight days after planting, we measured gs, the effective quantum yield (QY) of PS II, including the SD, SCA, and SinLA. The crop management and protocol for obtaining stomatal imprints and other data were the same across both experiments as detailed below. A schematic overview of the experiments in the study are visualized in Figure 3.1 A. Specifically, the figure shows information about phenotyping of SAP for SD and SCA in two environments (Env. 1 - Manhattan and Env. 2 - Hays) for two years (Exp. 1 in 2017 and Exp. 2 in 2018). More details about the experiments are found in (Bheemanahalli et al., 2021).

3.4 Stomatal density

To capture the natural variation in stomatal number and SCA, the Ab and Ad leaf surfaces were carefully smeared with a thin layer of transparent nail polish in the mid-por- tion of the fully opened leaf. Care was taken to identify the second leaf from the top that was fully open and completely developed, from which the imprints were obtained. After 3–5 min, thin imprints ($\approx 25 \times 17$ mm^2) were peeled off from both the leaf surfaces using tape (Scotch Transparent Clear Tape), and mounted on glass slides ($\approx 75 \times 25$ mm^2) following the procedure of Rowland-Bamford et al (Rowland-Bamford et al., 1990). Three random field of view images per slide were captured at magnification using the compound microscope (Olympus BX51 with DP 70 camera). From each image, the number of stomata was counted and divided by $0.24mm^2$ (area of each field) to estimate SD. In brief, number of stomata (N) was manually counted per field of view ($S = \pi r^2$, r = view radius) and SD was estimated as N/S (N = number of stomata mm^{-2}), as described in another study (Drake et al., 2013). A total of 11,196 images (311 accessions $\times$ 3 plants $\times$ 2 environments $\times$ 2 leaf surfaces $\times$ 3 images per slide) were used to record stomatal traits. Three leaves that we used for taking stomata imprints were harvested separately to determine the SinLA, using a leaf area meter (LI-3000; LI-COR, Lincoln, Nebraska, USA). Later, stomatal number per leaf was estimated to normalize the density on a whole leaf area basis, using the Ad and Ab SD per mm^2.

3.5 Experimental setup

The pretrained Mask R-CNN network was fine-tuned on datasets of increasingly larger sizes (specifically, 20, 50, 100, 200, and 300 images) and validated on a separate dataset consisting of 300 images. Using the training and validation loss curves, we selected the model trained on 300 images (280 images from Exp. 1 and 20 images from Exp. 2) to perform the stomata instance segmentation on the remaining images (i.e. images not included in the training and validation subsets), and subsequently produced the predic-

tions (i.e. deep learning dataset) used in this study. All images used for training and validation had stomata labeled using the VGG Image Annotator (1.0.6) tool, available at http://www.robots.ox.ac.uk/∼vgg/software/via/. The number of stomata in an image was obtained from the segmentation result and used to calculate SD, which was compared to the density obtained based on manual counting. Subsequently, the instance masks were used to calculate SCA, and the results were validated based on 50 images where stomatal area was manually measured using ImageJ (https://imagej.nih.gov/ij/). Finally, SCA was calculated for all images from Exp. 1 and Exp. 2 using the predicted stomata masks. The experimental design is summarized in Figure 3.1 B. Evaluation is performed in terms of root mean square error (RMSE) and correlation coefficient of model predictions by comparison with the manual annotated data. Evaluation is performed on the whole dataset, with the exception of training and validation subsets.

3.6 Results

To extract SD and SCA, in addition to manual counting of stomata on 11,196 images, we used the deep learning tool that we developed, specifically, the Mask R-CNN model (He et al., 2017), to extract SD and SCA automatically. The Mask R-CNN model was developed by experimenting with datasets of different sizes and identifies, classifies, and counts the number of stomata and measures SCA of all stomata in an image. Figure 3.2 shows the a comparison of the models trains with different dataset sizes and illustrate the process used to select the final model (A & B). Specifically, Figure 3.2 A shows the training loss graph obtained as described in what follows. Models were trained on our images by fine-tuning a Mask R-CNN model (with ResNet101 as the backbone network) pre-trained on the COCO dataset, using a learning rate of 0.001, and weight decay of 0.0001. We selected 20, 50, 100, 200, and 300 images, respectively, from the combined data of Env. 1 (2017 dataset) and Env. 2 (2018 dataset) as training data. We also selected 300 images from Env. 1 and Env.

2 as validation data. We trained models on 20, 50, 100, 200 and 300 images, respectively, to identify the number of images needed to produce accurate models. The x-axis shows the number of epochs a model is trained, while the y-axis shows the loss at each epoch. Given that we are interested in identifying accurate masks for stomata, the mask loss of the network on the training data is plotted. The graph shows that the mask training loss decreases when the number of the epochs increases. The graph also shows that the loss is smaller when the number of images used for training is smaller, suggesting that the models learned from smaller numbers of images may overfit the training data.

Figure 3.2 B shows the validation loss graph. Specifically, the graph shows the mask validation loss corresponding to the models trained with 20, 50, 100, 200, and 300 images, respectively. All models were validated with the same validation data consisting of 300 images. The graph shows that the validation loss decreases with an increase in the number of images used to train the models, and increases when the number of images used is small. In particular, for the model trained with 20 images, the validation loss substantially increases while the training loss decreases, suggesting overfitting. The validation loss for the model trained on 300 images is slightly better than the one for the model trained on 200 images. The graph also shows that for all models, the validation loss first decreases with the number of epochs, and then slightly increases after approximately 300 epochs. Based on this observation, we fixed the number of epochs to 300 in the models that are used to estimate the RMSE and correlation coefficient. Furthermore, the number of training and validation images used in the final model was 300 and 300, respectively, given the observed lowest validation loss.

A strong correlation was observed between the human measured and predicted values of the images in our dataset with r = 0.98 and Root Mean Square Error (RMSE) = 1.76 as shown in Figure 3.2 C and D, respectively. Specifically, Figure 3.2 C shows the RMSE graph: The x-axis represents the number of epochs on which a model is trained, while the y-axis shows the loss at each epoch. The RMSE is calculated based on the combined Env. 1 and Env. 2 dataset, from which we removed the 300 training and 300 300 validation images.

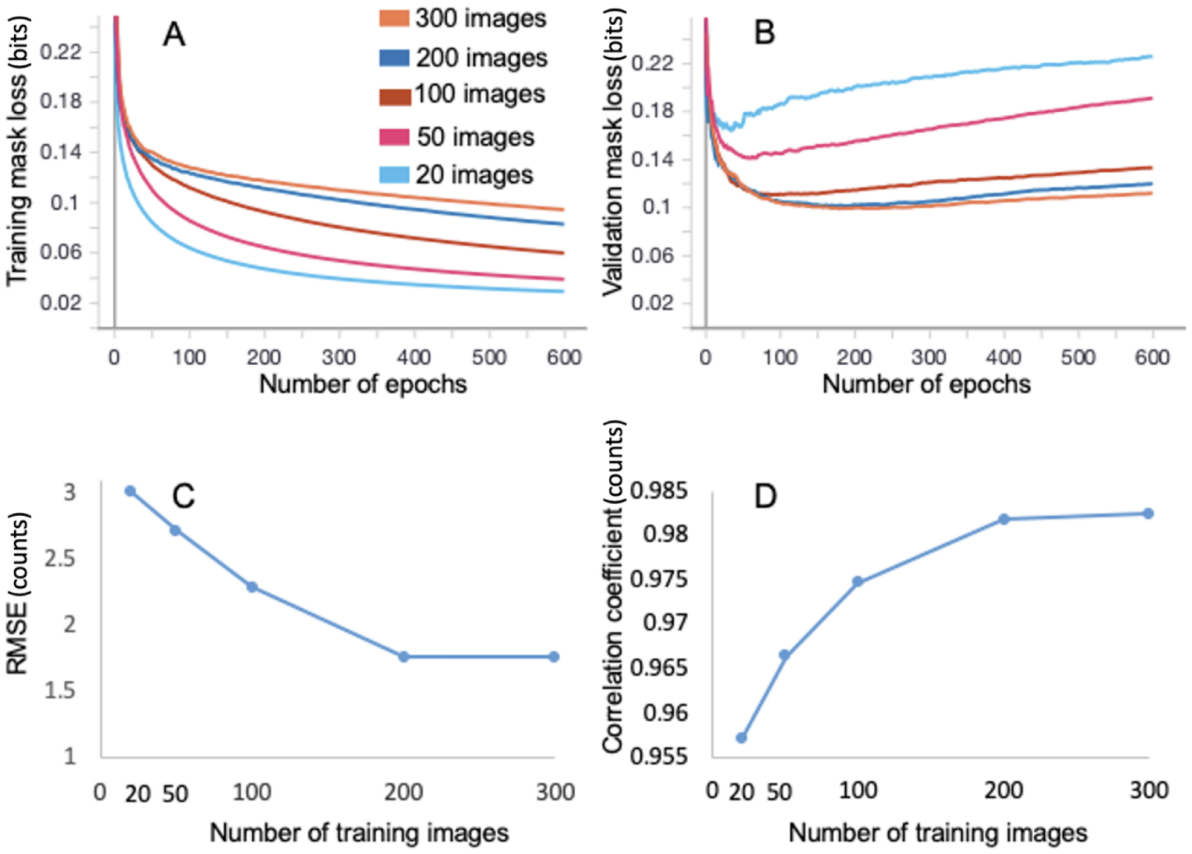


Figure 3.2: *Comparison of models trained with different dataset sizes. (A) Comparison of training loss curves of models trained with different dataset sizes, and (B) the corresponding validation loss curves. (C) Graph showing the root mean square errors (RMSE) corresponding to models trained with different dataset sizes (the RMSE is obtained by comparing the manually annotated data with the predicted data on the whole dataset, excluding the training and validation data). (D) The correlation coefficient of models trained with different dataset sizes (obtained by comparing manually annotated data with predicted data on the whole dataset, excluding the training and validation data).*

Trait	Acronym	G	E	G × E	Environment 1			Environment 2			H^2
					Minimum	Maximum	h^2	Minimum	Maximum	h^2	
Manual (Classical method)											
Ab SD (mm^{-2})	SDAb	<0.001	<0.001	<0.001	53.7	180.56	0.44	64.81	174.07	0.43	0.72
Ad SD (mm^{-2})	SDAd	<0.001	<0.001	<0.001	41.2	118.06	0.49	41.20	128.70	0.28	0.72
Ab stomatal number ($\times 10^6$ per leaf)	SNAb_LA	<0.001	<0.001	<0.001	0.89	9.33	0.39	1.31	8.15	0.43	0.79
Ad stomatal number ($\times 10^6$, per leaf)	SNAd_LA	<0.001	0.005	<0.001	0.66	5.70	0.36	1.03	5.21	0.10	0.78
Single leaf area (cm^2)	SinLA	<0.001	<0.001	<0.001	72.64	813.79	0.38	117.88	673.70	0.28	0.84
Predicted (deep learning method)											
Ab SD (mm^{-2})	SDAb	<0.001	<0.001	<0.001	52.78	177.78	0.46	66.20	168.52	0.45	0.72
Ad SD (mm^{-2})	SDAd	<0.001	<0.001	<0.001	41.67	118.98	0.52	38.89	125.46	0.28	0.72
Ab stomatal number ($\times 10^6$ per leaf)	SNAb_LA	<0.001	<0.001	<0.001	0.90	9.36	0.38	1.30	7.89	0.43	0.79
Ad stomatal number ($\times 10^6$ per leaf)	SNAd_LA	<0.001	NS	<0.001	0.66	5.71	0.36	1.05	5.09	0.08	0.78
Ab SCA (μm^2)	SCAAb	<0.001	NS	<0.001	538.01	879.11	0.45	550.16	885.21	0.33	0.74
Ad SCA (μm^2)	SCAAd	<0.001	<0.001	<0.001	574.77	1008.65	0.56	560.69	932.92	0.36	0.76

Probability values of the effects of genotype (G), environment (E), and their interaction (G × E) for all of the traits measured by ANOVA. NS indicates nonsignificant.
h^2 indicates marker-based narrow sense heritability using Genome Association and Prediction Integrated Tool.
H^2 indicates the broad-sense heritability estimated considering the proportion of phenotypic variance that is due to genetic variance. Mask R-CNN, a framework of deep learning method was used to predict the SD and SCA.

Figure 3.3: *ANOVA and variation in phenotypic traits using classical phenotyping and deep learning methods in SAP in environments 1 and 2.*

The graph shows that the models trained with 200 and 300 images have similar RMSE, and they have substantially smaller error than the model trained with 20, 50 and 100 images.

Finally, Figure 3.2 D shows the correlation coefficient graph: The x-axis represents the number of the training images used to train each model, while the y-axis shows the corresponding correlation coefficient for each model. The correlation coefficient is calculated based on the combined Env. 1 and Env. 2 dataset, from which we removed the 300 training and 300 validation images. The graph shows that the model trained with 300 images has the best correlation coefficient, followed by the model trained with 200 images.

As can be seen from Figure 3.2 C & D, the model trained with 300 images and validated with 300 additional images gave the lowest error and the highest correlation coefficient, and was hence later considered to explore the genetic diversity in the SD and SCA (Figure 3.3).

A comparison between manual (observed; Figure 3.4, A and B) and automated (prediction; Figure 3.4, C and D) stomata counts recorded a significant positive association between methods for Ab (R2 = 0.96 and R2 = 0.96; Figure 3.4, E and G) and Ad SD (R2 = 0.97 and R2 = 0.96; Figure 3.4, F and H) in Env. 1 and Env. 2, respectively. The broad-sense heritability (H2) values of the Ab (0.72) and Ad (0.72) SD were the same between methods

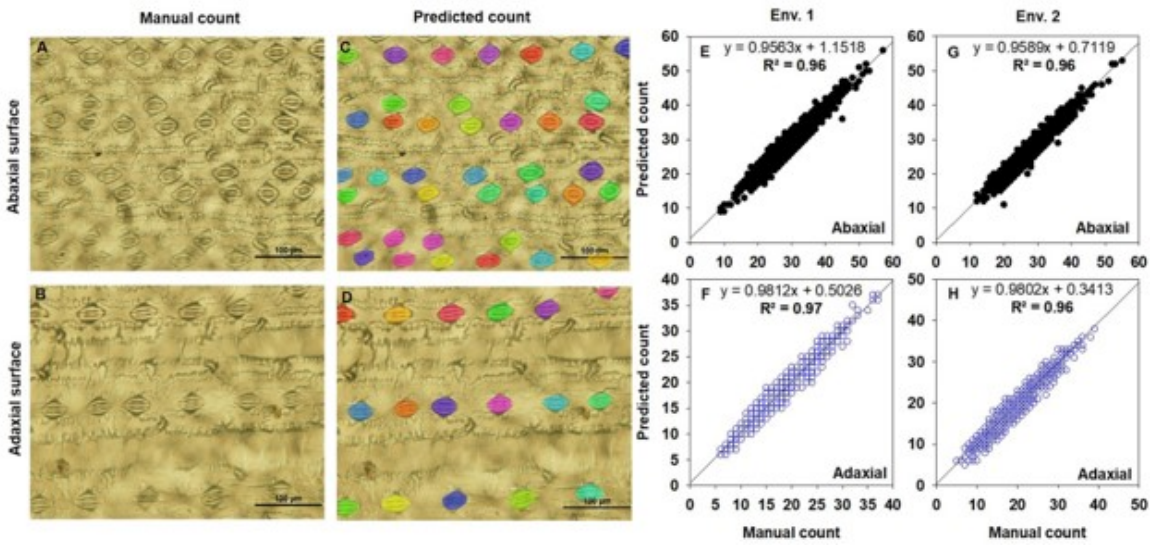


Figure 3.4: *Results of SD (per image) following manual and deep learning methods. Comparison of ground-truth images (A and B) and deep learning segmentation results (C and D, predicted stomata highlighted in colors). Relationship of the SD obtained from manual count with predicted count obtained from the deep learning method (E and G-Ab; F and H-Ad). SAP was characterized in two environments (Env. 1 and Env. 2). A total of 11,196 (in Exp. 1) and 828 (Exp. 2) images were used to manually count stomata and generate the observational ground-truth SD data. The same sets of images were used to predict the SD with the deep learning method, as illustrated in Figure 1. A-D, bars = 100 μm.*

(Figure 3.3).

There was a strong relationship between the predicted and human measured SCA for Ab (R2 = 0.91) and Ad (R2 = 0.90) leaf surfaces (Figure 3.5 A). Based on this strong relationship between the manual and predicted values for Ab and Ad SCA, the predicted data on the entire diversity panel were considered for further analysis (Figure 3.5).

3.7 Conclusions

Considerable evidence in field crops has shown the importance of stomatal characteristics and their association with photosynthesis and productivity (Farquhar and Sharkey, 1982), including rice (Bertolino et al., 2019; Buckley et al., 2020; Ohsumi et al., 2007), barley (Hughes et al., 2017), wheat(Dunn et al., 2019), and sorghum(Muchow and Sinclair, 1989). Previous studies have characterized the stomatal traits manually, either from a single en-

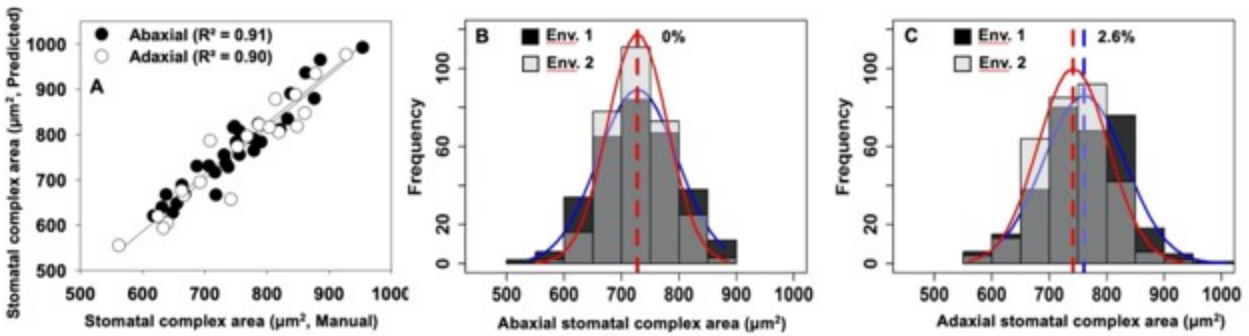


Figure 3.5: *Relationship of observed SCA (μm^2) with the corresponding data obtained using deep learning method (A). SCA was predicted using the deep learning approach on the entire SAP grown in Env. 1 and Env. 2 in 2017. Panels "B and C" show the distribution (Env. 1-blue line, dark gray bars; Env. 2-red line, light gray bars; intermediate gray bars indicate the overlap between the environments) of Ab and Ad stomata complex area, respectively. The vertical dotted lines on the histograms show population mean values in Env. 1 (blue) and Env. 2 (red). Values represent the positive percentage change in mean phenotypic value with respect to Env. 1 = [(mean trait value of Env. 1 - mean trait value of Env. 2)/mean trait value of Env. 1] × 100.*

vironment or under controlled environments using limited genetic diversity, due to challenges associated with phenotyping. Phenotyping of diversity panels for stomatal traits following the classical approach is cumbersome, with reproducibility of results from large-scale studies posing a substantial bottleneck (Furbank and Tester, 2011; Hudson, 2008). To bridge this knowledge gap, we characterized the genetic diversity in sorghum stomatal traits from two different environments by developing and integrating deep learning-based high-throughput phenotyping. We targeted the middle portion of the second fully developed leaf from the top, which is known to have the highest SD at the 14 leaves stage in sorghum, to collect stomatal imprints (Liang et al., 1975). The integration of the automated deep learning method (https://github. com/matterport/Mask_RCNN) helped overcome the time-consuming manual method of stomata counting and stomata complex area measurement, both in terms of speed and accuracy. Following the classical manual phenotyping approach, it took approximately 150 working days (≈ 3 min × 11, 196 images) to obtain the SD, while it took ≈ 7 days to obtain both SD and SCA by adopting the deep learning

method. A web server for detecting and counting stomata using our best model is available at https://rootanatomy.cs.ksu.edu/html_stomata/. Our work has the potential to contribute to stomata-targeted breeding as it can help uncover the molecular mechanisms that control stomatal regulation in sorghum to enhance adaptation under arid conditions with minimal to no yield penalty.

Chapter 4

Deep learning based high-throughput phenotyping of chalkiness in rice exposed to high night temperature

Background: Rice is a major staple food crop for more than half the world's population. As the global population is expected to reach 9.7 billion by 2050, increasing the production of high-quality rice is needed to meet the soaring demand. However, global environmental changes, especially increasingly high temperatures, can affect grain yield and quality. Heat stress is one of the major causes of an increased proportion of chalkiness in rice, which compromises quality and, in turn, reduces the market value. Researchers have identified 140 quantitative trait loci linked to chalkiness mapped across 12 chromosomes of the rice genome. However, the available genetic information quantified by employing advances in genetics has not been adequately exploited due to a lack of a reliable, rapid and high-throughput phenotyping tool to capture chalkiness. To derive extensive benefit from the genetic progress achieved, tools that facilitate high-throughput phenotyping of rice chalkiness are needed.

Results: We use a fully automated approach based on convolutional neural networks (CNNs) augmented with Gradient-weighted Class Activation Mapping (Grad-CAM) to de-

tect chalkiness in rice grain images. Specifically, we train a CNN model to distinguish between chalky and non-chalky grains and subsequently use Grad-CAM to identify the area of a grain that is indicative of the chalky class. The area identified by the Grad-CAM approach takes the form of a smooth heatmap that can be used to quantify the degree of chalkiness. Experimental results on both polished and unpolished rice grains using standard instance classification and segmentation metrics have shown that the Grad-CAM approach can accurately identify chalky grains and detect the chalkiness area.

Conclusions: We have successfully demonstrated the application of a Grad-CAM based tool to accurately capture high night temperature induced chalkiness in rice. The models trained will be made publicly available. They are easy-to-use, scalable and can be readily incorporated into ongoing rice breeding programs, without rice researchers requiring computer science or machine learning expertise.

4.1 Background

Rice (*Oryza sativa*) is a staple food crop for nearly half the world population (Federation, 2020). In 2019, the world produced over 750 million tonnes of rice (N.A., a), which placed rice as the third highest amongst cereals, only trailing wheat (*Triticum Aestivum*) (765 million tonnes) and maize (*Zea Mays*) (1.1 billion tonnes). As the global population is expected to reach 9.7 billion by 2050 (N.A., c), agricultural production must be doubled in order to meet this demand (N.A., b). As of 2008, rice yields are increasing on average by 1% annually and, at this rate, the production will only increase by 42% by 2050 which falls well short of the desired target (Ray et al., 2013).

In addition to the required increase in production, climate variability threatens future rice grain yields and quality attributes (Dabi and Khanna, 2018; Stuecker et al., 2018). Temperatures above 33°C during anthesis can cause significant spikelet sterility (Bheemanahalli et al., 2016; Jagadish et al., 2007, 2008, 2010). It is predicted that approximately 16% of

the global harvested area of rice will be exposed to at least five days of elevated temperature during the reproductive period by 2030s (Gourdji et al., 2013). In addition to yield losses, heat stress during the grain-filling period is shown to increase grain chalkiness in rice (Lisle et al., 2000; Lyman et al., 2013; Shi et al., 2017). Disaggregating the mean increase in global temperature has resulted in identifying a more rapid increase in the average minimum night temperature than the average maximum day temperature (Wang et al., 2017a). High night temperature stress during the grain-filling period can lead to severe yield and quality penalties, primarily driven by increased night respiration (Bahuguna et al., 2017; Impa et al., 2021; Sadok and Jagadish, 2020). An increased rate of night respiration during grain-filling ultimately impairs grain yield and grain quality through reduction in 1000 grain weight, grain width, reduced sink strength with lowered sucrose and starch synthase activity resulting in reduced grain starch content, and an increase in rice chalkiness (Bahuguna et al., 2017; Impa et al., 2021; Lanning et al., 2011; Shi et al., 2017).

Chalkiness is an undesirable trait and an increased proportion of chalk leads to linear decrease in market value of rice (Lyman et al., 2013). In addition, high levels of chalk leads to increased breakage during milling and degrades cooking properties, which can affect rice's palatability (Ashida et al., 2009; Fitzgerald et al., 2009; Lisle et al., 2000; Lyman et al., 2013). Chalkiness in grains is the visual appearance of loosely packed starch granules (Ashida et al., 2009; Shi et al., 2017). The poor packaging of starch granules leads to larger air pockets within the grain. The air pockets prevent reflection, giving the grains an opaque appearance (Tashiro and Wardlaw, 1991). Three different processes have been considered to explain the cause of increased chalkiness under heat stress: 1) a reduced source potential or a shortened grain-filling duration inhibits the plant's ability to provide a sufficient amount of assimilates to the seed, 2) reduced activity of starch metabolism enzymes, which are used to convert sugars to starch, and 3) hormonal imbalance between ABA and ethylene as a high ABA-to-ethylene ratio is vital during grain-filling (Jagadish et al., 2015). Physiologically, the level of chalkiness is dependent on the source-sink relationships, with the primary tillers in rice

having greater advantage of accessing the carbon pool compared to later formed tillers. We tested the hypothesis that, under higher night temperatures, increased carbon loss due to higher respiration would lead to different levels of grain chalkiness among the tillers with the least chalkiness from primary panicles and the highest chalkiness in the later formed tillers. Regardless of the cause or differential chalkiness among tillers, the ability to quickly and accurately identify and quantify the chalkiness in rice is extremely important to help not only to understand the cause of chalkiness, but also to breed for heat tolerant nutritional rice varieties.

Traditional grain phenotyping has been performed by manual inspection (Komyshev et al., 2017). As such, it is subjective, inefficient, tedious, and error-prone despite the fact that it is performed by a highly skilled workforce (Elmasry et al., 2019). Over the past decade, interest has grown in applying image-based phenotyping to provide quantitative measurements of plant-environment interactions with a higher accuracy and lower labor-cost than previously possible (Walter et al., 2015).

In particular, several automated approaches for rice grain chalkiness classification, segmentation and/or quantification have been developed. For example, the K-means clustering approach performs instance segmentation (i.e., identifies the pixels that belong to each instance of an object of interest, in our case "chalkiness") by grouping pixels based on their values (Sethy et al., 2018). One advantage of the K-means clustering approach is that it works in an unsupervised manner and does not require manually labeled ground truth (Alfred and Lun, 2019). However, one disadvantage is that it involves extensive parameter tuning to identify good clusters corresponding to objects of interest in an image. Furthermore, the final clusters depend on the initial centroids and the algorithm needs to be run several times with different initial centroids to achieve good results (N.A., d).

In addition to the K-means clustering approach, threshold based approaches have been used for chalkiness identification and quantification. For example, a multi-threshold approach based on maximum entropy was used for chalky area calculation (Yao et al., 2009),

and another threshold-based approach was used to detect broken, chalky and spotted rice grains (Payman et al., 2018). However, such approaches need extensive fine-tuning to identity the right thresholds and are not easily transferable to seeds of different types or to images taken under different conditions. Support vector machine (SVM) approaches have been used to classify grains according to the location of the chalkiness (Sun et al., 2014), and to estimate rice quality by detecting broken, chalky, damaged and spotted grains in red rice based on infrared images (Chen et al., 2019). Similar to the threshold-based approaches, the SVM classifiers are not easily transferable to images containing different types of seeds or taken under different illumination conditions. Furthermore, they require informative image features to be identified and provided as inputs to produce accurate results. Rice chalkiness has also been addressed using specially designed imaging instruments. For example, Armstrong et al. used a single-kernel near-infrared (SKNIR) tube instrument and a silicon-based light-emitting diode (SiLED) high-speed sorter to classify single rice grains based on the percentage of chalkiness (Armstrong et al., 2019). Unfortunately, the single-kernel approach is limited in scope and cannot be used to develop a high-throughput phenotyping method. More recently, volume based quantification technologies, such as X-ray microcomputed tomography, have been used to quantify rice chalkiness (Su and Xiao, 2020). However, such technologies are extremely expensive and, thus, are beyond the reach of routine crop improvement programs and for traders and millers who regularly estimate chalkiness and establish a fair market price.

In recent years, the use of deep learning approaches for image classification and segmentation crop science tasks, different from chalkiness classification and segmentation, have led to state-of-the-art high-throughput tools that outperform the results from traditional machine learning and image analysis techniques (Jones et al., 2017; Tardieu et al., 2017), thus enabling researchers to capture a wide range of genetic diversity (Singh et al., 2016). To the best of our knowledge, deep learning approaches have not been used to detect chalkiness despite the fact that they have been used to address other challenging problems in crop

science. To fill the gap, we investigated modern deep learning techniques to create a tool that facilitates high-throughput phenotyping of rice chalkiness to support genetic mapping studies and enable

development of rice varieties with minimal chalkiness under current and future warming scenarios. One possible solution to rapidly and accurately phenotype chalkiness is provided by Mask R-CNN (He et al., 2017). Mask R-CNN is a widely used instance detection and segmentation approach, which employs a convolutional neural network (CNN) as its backbone architecture. One limitation of the Mask R-CNN approach is that it requires pixel-level ground truth with respect to the concept of interest, in our case, chalkiness. Acquiring pixel-level ground truth is laborious and expensive (Xiao et al., 2019). Furthermore, the Mask R-CNN segmentation approach labels the pixels of a rice grain as chalky or non-chalky, while sometimes it maybe preferred to characterize the pixels based on the chalkiness intensity, i.e., on a continuous scale as opposed to a binary scale.

To address the limitations of the Mask R-CNN approach, we framed the problem of detecting chalkiness as a binary classification problem (i.e., a grain is chalky or non-chalky), and used CNNs combined with class activation mapping, specifically Grad-CAM (Rs et al., 2019), to identify the chalkiness area in an image. Grad-CAM works on top of a CNN model for image classification. It makes use of the gradients of a target category to produce a heatmap that identifies the discriminative regions for the target category (i.e., regions that explain the CNN model prediction) and implicitly localizes the category in the input image. By framing the problem as an image classification task, Grad-CAM can help reduce the laborious pixel-level labeling task to a relatively simpler image labeling task, i.e., an image is labeled as chalky or non-chalky. Furthermore, the heatmaps produced by Grad-CAM have soft boundaries showing different degrees of chalkiness intensity. The values of the pixels in a heatmap can be used to calculate a chalkiness intensity score corresponding to an image. Similar approaches to segmentation have been used in other application domains (Li et al., 2019; Schumacher et al., 2020; Vinogradova et al., 2020; Wang et al., 2017b; Yang et al.,

2020), including in the agriculture domain for segmentation of remote sensing imagery (Wang et al., 2020). Such approaches are generally called weakly supervised semantic segmentation approaches, given that they only require image-level labels as opposed to pixel-level labels.

The Grad-CAM based approach to rice chalkiness detection has the potential to help rice phenomics catch up with the developments in rice genomics (Yang et al., 2013) as well as help implementing new advances in achieving the target of nutritious food production goals by 2050 (of Public Information, 2009). To summarize, the contributions of this research are:

- We proposed to use a weakly supervised approach, Grad-CAM, to classify rice grains as chalky or non-chalky and subsequently detect the chalkiness area in chalky grains.
- We experimented with the Grad-CAM approach (with a variety of CNN networks as backbone) on polished rice seeds and evaluated the performance using both instance classification and segmentation metrics as well as time and memory requirements.
- We compared the weakly supervised Grad-CAM approach with the Mask R-CNN segmentation approach on polished seeds and studied its transferability to unpolished rice seeds (i.e., to rice seeds that have not been polished after the removal of the husk).
- We tested the applicability of the tool in determining the level of chalkiness in rice plants exposed to high night temperature (HNT) and quantified the differential level of chalkiness among tillers within a plant exposed to HNT stress.

4.2 Methods and materials

4.2.1 Deep learning methods for rice chalkiness segmentation

We address the rice chalkiness segmentation problem using a weakly supervised Grad-CAM approach, which requires binary (chalky or non-chalky) image-level labels as opposed to more expensive pixel-level labels.

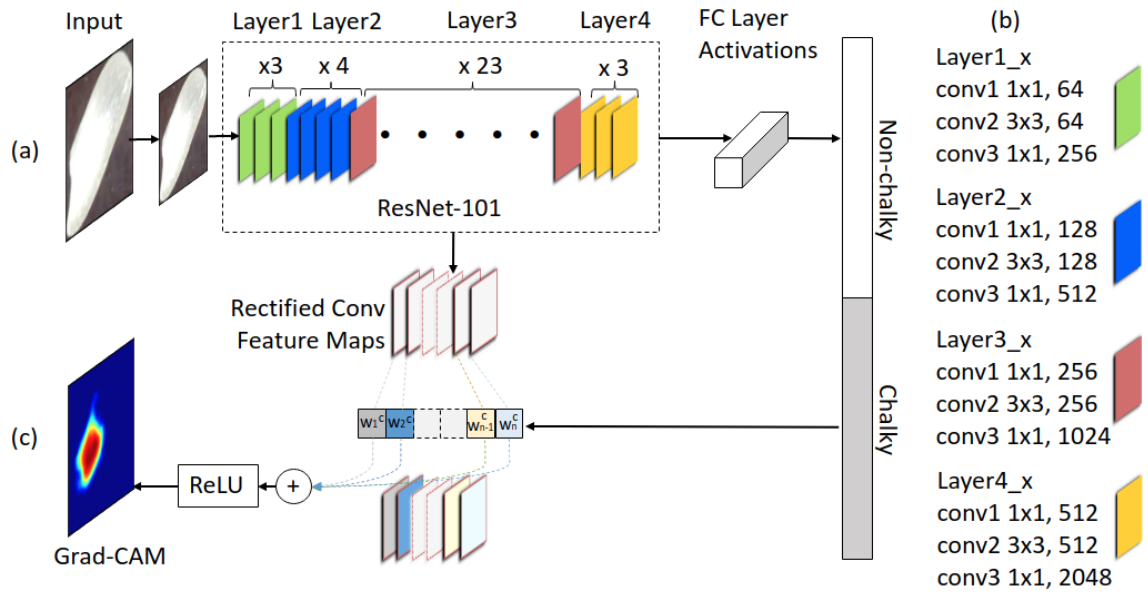


Figure 4.1: *Model Architecture. (a) A backbone CNN (e.g., ResNet-101) is trained to classify (resized) input grain images as chalky or non-chalky. ResNet-101 has four main groups of convolution layers, shown as Layer1, Layer2, Layer3, and Layer4, consisting of 3, 4, 23 and 3 bottleneck blocks, respectively. (b) Each bottleneck block starts and ends with a* 1×1 *convolution layer, and has a* 3×3 *layer in the middle. The number of filters in each layer is shown after the kernel dimension. (c) Grad-CAM uses the gradients of the chalky category to compute a weight for each feature map in a convolution layer. The weighted average of the features maps, transformed using the ReLU activation, is used as the heatmap for the current image at inference time.*

Overview of the approach. The Grad-CAM approach includes two main components: (i) a deep CNN network (e.g., VGG or ResNet) that is trained to classify seed images into two classes, chalky or non-chalky; and (ii) a class activation mapping component, which generates a rice chalkiness heatmap as a weighted average of the feature maps corresponding to a specific layer in the CNN network. The chalkiness heatmap can be further used to calculate a chalkiness score, which quantifies the degree of chalkiness in each individual grain, and to estimate the chalkiness area for each grain. An overview of the approach is shown in Figure 4.1. Details for the components of the model are provided below.

CNNs. Models based on CNNs have been successfully used for many image classification and segmentation tasks (Goodfellow et al., 2016; LeCun et al., 1989, 2015). A CNN consists of convolutional layers (which apply filters to produce feature maps), followed by non-linear

activations (such as Rectified Linear Unit, or ReLU), pooling layers (used to reduce the dimensionality), and fully connected layers (that capture non-linear dependencies between features). The last fully connected layer in a classification network generally uses a softmax activation function and has as many output neurons as the number of target classes (in our case, two classes - chalky and non-chalky).

The ImageNet competition (where a dataset with 1.2 million images in 1000 categories was provided to participants) has led to many popular architectures, including highly competitive architectures in terms of performance, as well as cost-effective architectures designed to be run efficiently on low-cost platforms generally present in embedded systems (Russakovsky et al., 2015). We anticipate that our rice chalkiness detection models could be useful in both environments with rich computational resources and also environments with more limited resources. Thus, given the trade-off between model performance (i.e., accuracy) and model complexity (e.g., number of parameters, memory and time requirements), we consider a variety of networks (and variants) published between 2012 and 2019 including AlexNet (Krizhevsky et al., 2012), Very Deep Convolutional Networks (VGG) (Simonyan and Zisserman, 2014b), Deep Residual Networks (ResNet)(He et al., 2016), SqueezeNet (Iandola et al., 2016), Densely Connected Convolutional Networks (DenseNet) (Huang et al., 2017), and EfficientNet(Tan and Le, 2019).

Grad-CAM approach. The Grad-CAM approach was originally proposed by Selvaraju et al. Selvaraju et al. (2019) in the context of understanding the predictions of a CNN model. In recent years, this approach and its variants have been frequently used for weakly supervised object localization (Zhou et al., 2016). Given a trained CNN model and an input image at inference time, the Grad-CAM approach uses the gradients of a category of interest (specifically, the corresponding logit provided as input to the softmax function) to compute a category-specific weight for each feature map in a selected convolution layer. Formally, let f^k (with $k = 1, \ldots, K$) be a feature map in a particular convolutional layer, which consists of a total of K feature maps. Let y^c be the logit (i.e., input to the softmax

function) of the class of interest, c (e.g., chalky). Grad-CAM averages the gradients of y^c with respect to all N pixels f_{ij}^k of the features map f^k to calculate a weight w_k^c representing the importance of the feature map f^k. Specifically, $w_k^c = \frac{1}{N}\sum_{i,j}\frac{\partial y^c}{\partial f_{i,j}^k}$. The feature maps f^k of the selected convolutional layer are averaged into one final heatmap for the category of interest, c, according to the importance weights w_c^k, i.e., $H^c = F\left(\sum_k w_c^k f^k\right)$, where F is a non-linear activation function. Traditionally, ReLU has been used as the activation function to cancel the effect of the negative values while emphasizing areas that positively contribute to the category c. The heatmap, H^c, is resized to the original input size using linear interpolation. The resized heatmap, H_{final}^c, can be used to identify the discriminative regions for the category of interest, c, and implicitly to localize the category in the input image. The last convolutional layer was originally used by Selvaraju et al. (Selvaraju et al., 2019), under the assumption that the last layer captures the best trade-off between high-level semantic features and spatial information. However, in this study, we experimented with a variety of convolutional layers, from lower level convolutional layers (more general) to higher level convolutional layers (more specific), to identify sets of features maps that best capture chalkiness.

Application of Grad-CAM to rice chalkiness detection. We used the Grad-CAM approach to generate chalkiness heatmaps for rice grain images. The heatmaps show the concept of chalkiness using soft boundaries through a color gradient. This representation is very appropriate for localizing the concept of chalkiness, which exhibits different levels of intensity and, thus, has inherently soft boundaries that separate the chalky area from the non-chalky area. The heatmap, H_{final}^{chalky}, corresponding to a particular convolutional layer (determined using validation data) is the final rice chalkiness heatmap and can be used to visualize the area of a seed that is discriminative with respect to chalkiness. This heatmap can further be converted into a chalkiness score corresponding to a rice grain as follows: $ChalkyScore = \frac{1}{Z}\sum_i\sum_j(H_{final}^{chalky} \cap GrainArea)$, where Z represents the total number of pixels in the $GrainArea$ in the original image. The resulting chalkiness score has a numerical

value between 0 and 1, where 0 means that the grain shows no chalkiness and 1 means that the grain has severe chalkiness all over its surface. Finally, the heatmap is used to create a binary mask for the chalkiness area using a threshold on the intensity of the pixels (determined based on validation data). The masked area can be used to estimate the area of the chalkiness as a percentage of the total grain area. The numeric scores, including the chalkiness score and the chalkiness area, obtained from large mapping populations can be used in determining the genetic control of chalkiness in rice.

Baseline approach - Mask R-CNN. Mask R-CNN is an object instance segmentation approach, i.e., an approach that identifies instances of given objects in an image (in our case, the chalkiness concept) and labels their pixels accordingly. Mask R-CNN extends an object detection approach, specifically Faster R-CNN (Ren et al., 2015), to perform instance segmentation. The Faster R-CNN network first identifies Regions of Interest (ROI, i.e., regions that may contain objects of interest) and their locations (represented as bounding box coordinates) using a Region Proposal Network (RPN). Subsequently, the Faster R-CNN network classifies the identified regions (corresponding to objects) into different classes (e.g., chalkiness and background) and also refines the location parameters to generate an accurate bounding box for each detected object. In addition to the object classification and the bounding box regression components of the Faster R-CNN, the Mask R-CNN network has a component for predicting instance masks for ROIs (i.e., identifying all pixels that belong to an object of interest). One advantage of the Mask R-CNN approach is that it is specifically trained to perform instance segmentation and, thus, produces a precise mask for objects of interest. The main disadvantage of the Mask R-CNN baseline, as compared to the weakly supervised Grad-CAM approach, is that it requires expensive pixel-level annotation for training.We compared the weakly supervised Grad-CAM approach to chalkiness segmentation with Mask R-CNN in terms of performance and also time and memory requirements.

4.2.2 High night temperature stress experiment

In this section, we describe plant materials and the biological experiment that generated the data (i.e., rice grains) used in this study.

Plant materials. Six genotypes (CO-39, IR-22, IR1561, Oryzica, WAS-174, and Kati) with contrasting chlorophyll index responses to a 14-day drought stress initiated at the agronomic panicle-initiation stage were used in this study (Šebela et al., 2019). The experiment was carried out in controlled environment chambers (Conviron Model CMP 3244, Winnipeg, MB) at the Department of Agronomy, Kansas State University, Manhattan, KS, USA.

Crop husbandry and high night temperature stress imposition. Seeds obtained from the Germplasm Resources Information Network (GRIN) database were sown at a depth of 2 cm in pots (1.6-L, 24 cm tall and 10 cm diameter at the top, MT49 Mini-Treepot) filled with farm soil. Seedlings were thinned to one per pot at the three-leaf stage. Controlled-release Osmocote (Scotts, Marysville, OH, USA) fertilizer (19% N, 6% P2O5, and 12%K2O) was applied (5 g per pot) before sowing along with 0.5 g of Scotts Micromax micronutrient (Hummert International, Topeka, KS) at the three-leaf stage. The plants were well-watered throughout the experiment, and a 1-cm water layer was maintained in the trays holding the pots. Seventy-two plants were grown with at least 12 plants per genotype wherein 6 plants were used for control and the remainder for HNT. Plants were grown in controlled environment chambers maintained at control temperatures of 30/21°C (maximum day/minimum night temperatures; actual inside the chamber: 32.6°C [SD±1.0]/21.1°C [SD±0.3]) and relative humidity (RH) of 70% until treatment imposition. Both control and HNT chambers were maintained at a photoperiod of 11/13 h (light/dark; lights were turned on from 0700 to 1800 h, with a dark period from 1800 to 0700 h) with a light intensity of 850 $\mu mol\ m^{-2}s^{-1}$ above the crop canopy. Temperature and RH were recorded every 15 min using HOBO UX 100-011 temperature/RH data loggers (Onset Computer Corp., Bourne, Massachusetts) in all growth chambers. At the onset of the first spikelet opening, the main tiller, primary tillers and other tillers of the flowering genotype were tagged and readied for treatment

imposition. The same approach was followed for all six genotypes and replicates. Tagged replicate plants were moved to HNT (30/28°C) chambers and equal numbers of plants were similarly tagged and maintained in control conditions. Six independent plants for each genotype were subjected to HNT stress (30/28°C- day/night temperatures; actual: 31.8°C [SD±0.8]/27.9°C [SD±0.1]) after initiation of flowering on the main tiller until maturity to determine the impact of HNT on chalkiness while the other six plants were maintained under control conditions.

Data Collection. At physiological maturity, the plants were harvested from both the control and HNT treatments. The panicles were separated into main panicles (the panicle on the main tiller), two primary panicles (tillers that followed the main panicle), and other remaining panicles for each plant from each treatment and hand threshed separately. Subsequently, the grains were de-husked using the Kett, Automatic Rice Husker TR-250.

In addition to the unpolished grains, polished grains were also used in the initial model development and testing, as polished grains are easier to analyze and label with respect to chalkiness and could potentially be beneficial in terms of knowledge transfer to unpolished rice. The polished grains were obtained from Rice Research and Extension Center in Stuttgart Arkansas, University of Arkansas for preliminary testing and to establish the model. The polished rice grains composed of both medium and long grain rice. For each grain size, there are three degrees of grain chalkiness (roughly estimated by a domain expert): low, medium, and high chalkiness. Thus, based on grain size and degree of chalkiness, the grains were grouped into six categories: 1) long grain, low chalkiness; 2) long grain, medium chalkiness; 3) long grain, high chalkiness; 4) medium grain, low chalkiness; 5) medium grain, medium chalkiness; and 6) medium grain, high chalkiness.

4.2.3 Rice grain image acquisition and processing

Image acquisition. Both polished and unpolished grain samples were arrange in transparent 90 mm Petri-plates with three Petri-plates for each sample. A sample corresponds

to a size/chalkiness combination in the case of polished rice and a genotype/tiller/condition combination in the case of unpolished rice. Three replicates (i.e., sets of grains to be used in one scan) were randomly selected (without replacement) for each sample. The grains were scanned using an Epson Perfection V800 photo scanner attached to a computer (see Supplementary Figure S1). Images were scanned at a resolution of 800 dots per inch (dpi) and saved in the TIFF (.tif) file format for further image analysis. A total of 18 (i.e., $3 \times 2 \times 3$) images were acquired for polished rice, and 108 (i.e., $3 \times 6 \times 3 \times 2$) images for unpolished rice. The scanned images included all borders of the three Petri-plates but not excessive blank area outside of the dishes, as shown in Supplementary Figure S2.

Image preprocessing. Each scanned image (for both polished and unpolished rice grans) had size of approximately 6000×6000 pixels. This size is extremely large for deep learning approaches, which require GPU acceleration (He et al., 2018). Furthermore, as we aim to perform chalkiness detection at grain level using a weakly supervised approach, we need images that contain individual seeds. To reduce the size of the images and to enable grain level labeling and analysis, we resorted to cropping individual grains from the original Petri-plate images (which contain approximately 25-30 rice grains per plate). The following steps, illustrated in Figure 4.2, were used to crop individual grain images: (i) we first converted original images from .tif to .jpg format; (ii) converted RGB images to grayscale images; (iii) performed canny edge detection; (iv) identified bounding boxes corresponding to individual seeds; (v) extracted ROIs defined by the bounding boxes and saved each ROI/grain as an image into a file with unique file name.

The total number of individual seeds extracted from the images containing Petri-plates with polished rice grains was 1645 out of the total of 1654 grains in the original set of 18 images. Nine seeds got truncated and were removed from the dataset. The exact number of polished seeds in each image and the corresponding number of extracted seed images are shown in Supplementary Table T1 in columns 4 (Grains original) and 5 (Grains used), respectively. Similarly, the total number of individual seeds extracted from the images

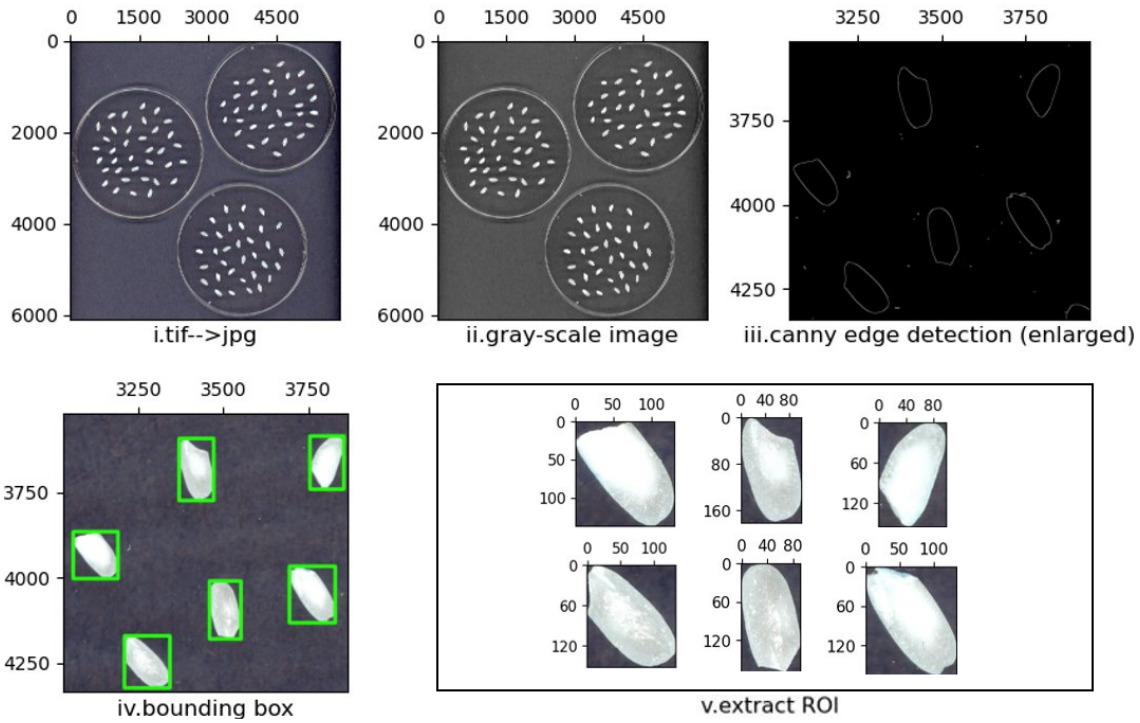


Figure 4.2: *Image preprocessing. Steps used to crop individual rice seeds from the original scanned images, each with approximately 25-30 seeds. Five steps (i. to v.) are depicted below each image that illustrate the action achieved in each respective step.*

containing Petri-plates with unpolished rice grains was 13,101 out of the total of 13,149 seeds in the original set of 108 high resolution images. In this case, 48 seeds got truncated and were not included in the final set. The exact number of unpolished seeds in each of the 108 images and the corresponding number of individual seed images extracted are shown in Supplementary Table T2 in columns 5 (Grains original) and 6 (Grains used), respectively.

4.2.4 Image annotation and benchmark datasets

Ground truth labeling. Two types of manual annotations were performed and used as ground truth in our study, as shown in Figure 4.3. First, for the Grad-CAM weakly supervised approach to chalkiness segmentation, we labeled each rice grain image as chalky or non-chalky. The labeling was done based on visual inspection of the images by a domain expert. Second, to train Mask R-CNN models, which inherently perform instance segmentation, and to evaluate the ability of the Grad-CAM approach to accurately detect

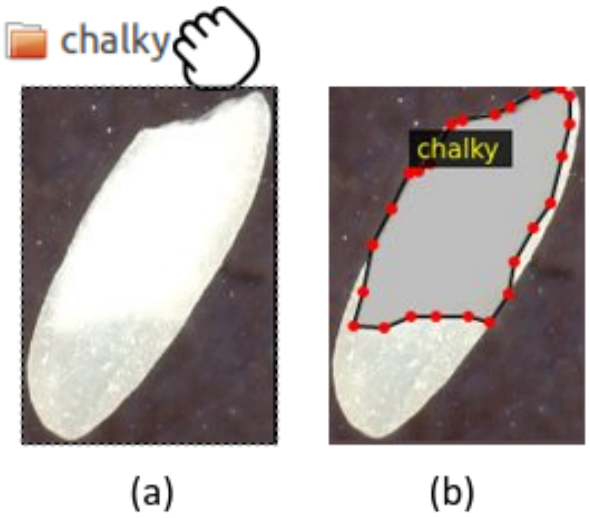


Figure 4.3: *Manual annotations. (a) Image-level annotation: each seed is labeled as chalky or non-chalky (technically, the label was created by dragging each rice seed image into chalky or non-chalky folder, respectively). (b) Specific chalkiness annotation: chalkiness area is marked with polygons using VGG Image Annotator (each red dot in the image represents a click). The dark white opaque region in panel "a" is the chalk portion while the non-chalky region is translucent.*

the chalkiness area in a rice grain, we manually marked the chalkiness area using polygons. The polygon annotation was performed by a domain expert using the VGG Image Annotator (Dutta and Zisserman, 2019), a web-based manual annotation software. Compared to the image-level labeling (i.e., chalky/non-chalky), the polygon annotation is significantly more expensive, as it requires 10 to 100 clicks to draw the polygons, given the irregular shape of the chalkiness area.

Out of 1645 polished grains used in our study, 660 grains were labeled as chalky and 985 seeds were labeled as non-chalky. The exact numbers of chalky and non-chalky seeds in each of the eighteen high-resolution images with polished rice are shown in Supplementary Table T1 in columns 6 (Chalky) and 7 (Non-chalky), respectively. To be able to evaluate segmentation performance and to compare the Grad-CAM approach with Mask R-CNN, we also labeled the 660 chalky grains in terms of chalkiness area (represented as a polygon).

Similarly, out of 13,101 unpolished grains, 4085 grains were labeled as chalky and 9016 grains were labeled as non-chalky. The exact numbers of chalky and non-chalky grains in

each of the 108 high-resolution images of unpolished rice are shown in Supplementary Table T2 in columns 7 (Chalky) and 8 (Non-chalky), respectively. We note that many of the 36 possible genotype/tiller/condition combinations have a small number of chalky grains (or do no have any chalky grain at all). Specifically, 12 combinations corresponding to genotypes CO-39 and Kati contain 4085 chalky grains and 1299 non-chalky grains, while the remaining 24 combinations contain 151 chalky grains and 7717 non-chalky grains. Thus, we used only the 12 chalky prevalent combinations for training, tuning and evaluating the models designed in this study. Twenty chalky grain images from each of these 12 combinations (for a total of 240 images) were used as test set. To estimate the chalkiness segmentation performance on unpolished rice, the 240 test images were labelled also in terms of chalkiness area using polygons. We did not label all the chalky images in terms of chalkiness area due to the cost associated with this annotation. The number of images labeled as chalky and non-chalky, and also the number of chalky images annotated in terms of chalkiness area are summarized in Table 4.1.

Table 4.1: *Statistics on manual image annotation, specifically, the number of images labeled as chalky and non-chalky, and also the number of chalky images annotated in terms of chalky area, for polished images, and unpolished images from 12 chalky combinations, respectively.*

Set of seeds	Chalky	Non-chalky	Total	Chalky Area
Polished	660	995	1,645	660
Unpolished (12)	3,934	1,299	5,233	240

Training, development and test datasets. To train, fine-tune and evaluate our models, we created training, development and test datasets for both polished and unpolished (12) grain images. In the case of polished grain images, for each grain size and chalkiness degree combination, we used one of the three replicates for testing while the other two replicates were split between training and development subsets. In the case of unpolished seed images, for each genotype, tiller and condition combinations, we used one of the three replicates for testing and the other two replicates were split between training and development subsets. The specific distribution of chalky/non-chalky grain images in the training/development/test

Table 4.2: *Distribution over Training/Development/Test subsets*

Set of seeds	Training		Development		Test		Total
	Chalky	Non-chalky	Chalky	Non-chalky	Chalky	Non-chalky	
Polished	326	497	168	243	166	245	1,645
Unpolished (12)	1,856	830	483	229	1,595	240	5,233

subsets is shown in Table 4.2. It should be noted that our splitting process ensures that the training/development/test subsets contain all types of grains considered and there is no grain that belongs to at least two subsets. We used the training subsets to train the models (both Grad-CAM networks for binary chalky/non-chalky classification and the Mask R-CNN networks for chalkiness segmentation). We used the development subsets to fine-tune hyper-parameters for the models. Finally, the performance of the models is evaluated on the test subsets. The subsets are made publicly available to ensure reproducibility and to enable further progress in this area.

4.2.5 Experimental Setup

In this subsection, we state several research questions that we aim to address and describe the experiments performed to answer these questions. We also discuss the metrics used to evaluate the models trained in our experiments and the hyper-parameters that were fine-tuned to obtain the most accurate models.

Research Questions. We aim to answer the following research questions (RQ):

RQ1 Among different CNN networks used as the backbone in the Grad-CAM models for polished rice, what network is the best overall in terms of chalky/non-chalky classification performance versus time and memory requirements? Also, what network is the best overall in terms of chalkiness segmentation?

RQ2 How does the Grad-CAM weakly supervised approach to chalkiness segmentation compare with the Mask R-CNN segmentation approach to chalkiness detection in polished rice?

RQ3 What is the performance of the Grad-CAM models for unpolished rice? What is the performance of the polished rice models when used to make predictions on unpolished rice? Does the performance improve if we fine-tune the polished rice models with unpolished rice?

Experiments. To answer RQ1, we trained Grad-CAM models with several CNN networks as backbone, including variants of AlexNet, DenseNet, ResNet, SqueezeNet, VGG and EfficientNet pre-trained on ImageNet. We compared the models in terms of classification performance, memory and time requirements. We also identified the best model/network for each type of architecture. Subsequently, we study the variation of those best models with respect to the layer used to generate the heatmaps and the threshold used to binarize the heatmaps when calculating the average Intersection-over-Union (IoU). The goal is to identify the best overall layer and threshold for each type of network. The best models (with the best layer and threshold) are used to evaluate the localization accuracy, both quantitatively and qualitatively, for chalkiness detection in polished rice. To answer RQ2, we also trained Mask R-CNN models (with the default ResNet-101 as backbone) and compared them with the best weakly supervised Grad-CAM approach. Finally, to answer RQ3, we first trained and evaluated a Grad-CAM model (with ResNet-101 as backbone) on unpolished rice. We compared the performance of the resulting model with the performance of a model trained on polished rice and also with the performance of the polished rice model fine-tuned on unpolished rice.

Evaluation metrics. We evaluated the performance of the Grad-CAM approach along two main dimensions. First, we evaluated the ability of the approach to correctly classify seeds as chalky and non-chalky using standard classification metrics such as accuracy, precision, recall and F1 measure. Second, we evaluated the ability of the approach to perform chalkiness segmentation (i.e., the ability to identify the chalky area in the chalky seed images) using standard segmentation metrics. Specifically, we calculated average IoU (Shelhamer et al., 2016), as well as localization accuracy and ground truth known (GT-known) localization

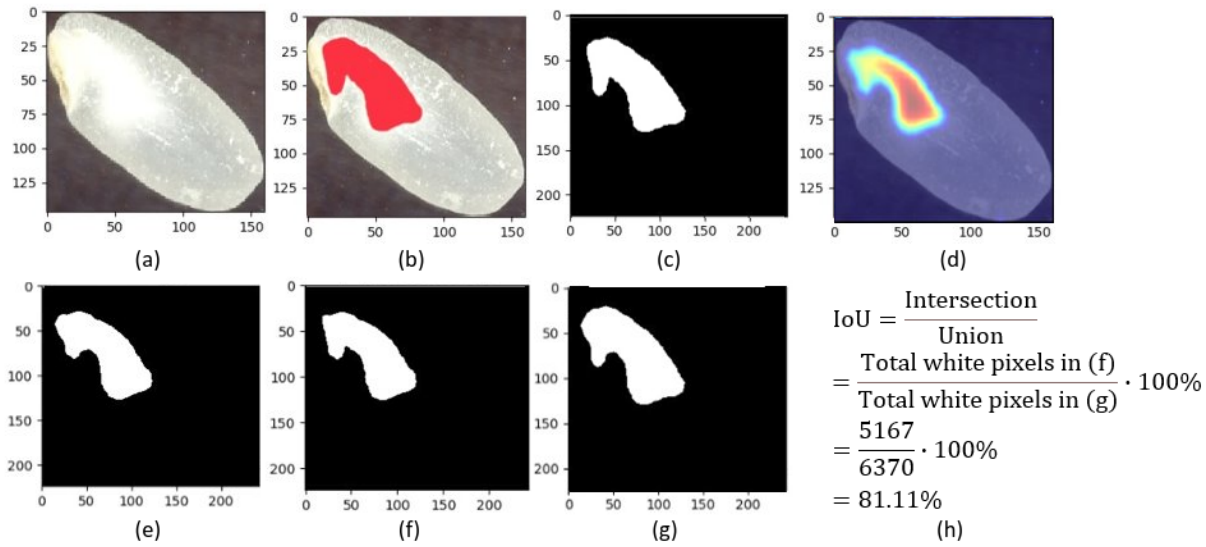


Figure 4.4: *Calculating the IoU between binarized ground truth and prediction: (a) chalky seed; (b) corresponding ground truth chalkiness area; (c) binarized ground truth area; (d) predicted chalkiness area; (e) corresponding predicted binarized area; (f) intersection between the binerized ground truth (c) and prediction (e): the number of white pixels in the intersection is 5167; (g) union between the binarized ground truth (c) and prediction (e): the number of white pixels in the union is 6370; (h) Calculation of IoU.*

accuracy (Russakovsky et al., 2015) for the chalky class. Figure 4.4 illustrates the process of calculating IoU between the ground truth mask for the chalkiness area and the predicted chalkiness mask. As opposed to classification accuracy, which considers a prediction to be correct if it exactly matches the ground truth label, the localization accuracy considers a prediction to be correct if both the image label and the location of the detected object are correct. For the location of the object to be correct, the object mask needs to have more than 0.5 overlap with the ground truth mask. The overlap is measured as the IoU. In our case, we calculated the localization accuracy for the chalky class as the fraction of seed images for which the predicted mask for the chalky area has more than 50% IoU with the ground-truth mask. We also calculated the GT-known localization accuracy, which eliminates the influence of the classification results, as it considers a prediction to be correct when the IoU between the ground truth mask and estimated mask (in our case, for the chalky class seed images) is 0.5 or more.

Hyper-parameter tuning. Deep learning models, in general, and the ResNet, VGG,

SqueezeNet, DenseNet EfficientNet networks, in particular, have many configurable hyper-parameters. We tuned several hyper-parameters shown to affect the performance of all models. More specifically, we tuned the batch size used in gradient descent to control the number of rice seeds processed before updating the internal model weights. Furthermore, we tuned the learning rate which controls how much we are adjusting the network weights with respect to the gradient of the loss function. The specific values that we used to tune the batch size were 16, 32 and 64. The values used to tune the learning rate were 0.1, 0.01, 0.001, 0.0001 and 0.00001. For each network, the best combination of parameters was selected based on the F1 score observed on the validation subset. Each model was run for 200 epochs and the best number of epochs for a model was also selected based on the validation subset. Overall, our hyper-parameter tuning process revealed that the performance did not vary too much with the parameters considered. All the models were trained on Amazon Web Services (AWS) p3.2xlarge instances.

As opposed to the models used as backbone for the Grad-CAM approach, the Mask R-CNN network with ResNet-101 as backbone, could only be trained with a batch size of 8 images on AWS p3.2xlarge instances. The same learning rate values as for the CNN networks were used for tuning. However, this network was trained for a total of 600 epochs, as opposed to just 200 epochs for the other models. No other hyper-parameters specific to Mask R-CNN network were fine-tuned.

4.3 Results and discussion

4.3.1 Chalkiness classification and detection in polished rice using Grad-CAM models

Chalkiness classification in polished rice. Table 4.3 shows classification results for a variety of network architectures (and variants within one type of architecture) that were used as backbone for the Grad-CAM models. Specifically, we experimented with variants of

the DenseNet, ResNet, SqueezeNet, VGG, and EfficientNet architectures. All the variants that we used have models pre-trained on ImageNet, which allowed us to perform knowledge transfer and train weakly supervised models for chalkiness detection with a relatively small number of chalky/non-chalky seed images. Only models that we could train on AWS p3.2xlarge instances were included in the table to allow for a fair comparison in terms of training time. Each model is trained and fine-tuned on the training and development subsets consisting of polished rice seed images. Performance is reported in terms of overall accuracy and also precision, recall and F1 measure for both the chalky and non-chalky classes. The best results for one type of architecture are highlighted with bold font. For each model included in Table 4.3, Table 4.4 shows the training time (seconds), number of parameters, and size (MB) of the models versus the classification accuracy of the model.

As can be seen from Table 4.3, the overall classification accuracy varies from 93.67% (for EfficientNetB2 and EfficientNetB5) to 95.61% (for DenseNet-121). The DenseNet-121 model, which has the highest classification accuracy, also has the highest F1 measure for both chalky and non-chalky classes, although there is at least one competitive variant for each architecture type, e.g., ResNet-101 for ResNet, SqueezeNet-1.0 for SqeezeNet, VGG-16 for VGG, and EfficientNetB4 for EfficientNet. Furthermore, the DenseNet-121 model has a relatively small size (28 MB) and average training time (approximately 1500 seconds). Surprisingly, the SqueezeNet architecture, which is highly competitive in terms of performance, has the smallest size (3.0/2.9 MB for SqueezeNet-1.0/SqueezeNet-1.1, respectively) and smallest training time (approximately 500 seconds). The VGG models have the largest size (more than 500 MB) and relatively large training time (in the range of 2400 to 3000 seconds), and the best EfficientNet variant (EfficientNetB4) has moderate size (approximately 140 MB) but relatively large training time (approximately 3500 seconds). Finally, the ResNet-101 variant, which is the best in the ResNet group, has moderate size (170 MB) and training time (close to 1700 seconds). Based on these results, we selected one model for each type of architecture and used those selected models for further analysis.

Table 4.3: *Classification results on polished rice with various networks as backbone in the weakly supervised Grad-CAM approach. The number following a network's name denotes the number of layers in the network (as in DenseNet-121 or ResNet-101) or the version of the network (as in SqueezeNet-1.0 or EfficientNetB0). Performance is reported in terms of Accuracy (Acc.), Precision (Pre.), Recall (Rec.) and F1 measure (F1). Precision, Recall and F1 measure values are reported separately for the Chalky and Non-Chalky classes. All models are trained/tuned/evaluated on the same training/development/test splits. The results reported are obtained on the test set. The best performance for each type of model for each metric is highlighted using bold font.*

Model	Acc.(%)	**Chalky**			**Non-Chalky**		
		Pre. (%)	Rec.(%)	F1(%)	Pre.(%)	Rec.(%)	F1(%)
DenseNet-121	**95.61**	**94.58**	94.58	**94.58**	96.31	**96.31**	**96.31**
DenseNet-161	95.12	92.44	**95.78**	94.08	**97.06**	94.67	95.85
DenseNet-169	94.63	92.86	93.98	93.41	95.87	95.08	95.47
ResNet-18	94.63	94.44	92.17	93.29	94.76	96.31	95.53
ResNet-34	94.15	93.29	92.17	92.73	94.72	95.49	95.10
ResNet-50	94.88	**95.03**	92.17	93.58	94.78	**96.72**	95.74
ResNet-101	**95.12**	93.45	**94.58**	**94.01**	**96.28**	95.49	**95.88**
ResNet-152	94.88	93.94	93.37	93.66	95.51	95.90	95.71
SqueezeNet-1.0	**95.12**	**93.45**	94.58	**94.01**	96.28	**95.49**	**95.88**
SqueezeNet-1.1	94.39	91.33	**95.18**	93.22	**96.62**	93.85	95.22
VGG-11	94.88	**93.94**	93.37	93.66	95.51	**95.90**	95.71
VGG-13	94.39	92.31	93.98	93.13	95.85	94.67	95.26
VGG-16	**95.12**	92.94	95.18	**94.05**	96.67	95.08	**95.87**
VGG-19	94.15	90.34	**95.78**	92.98	**97.01**	93.03	94.98
EfficientNetB0	95.13	93.98	93.98	93.98	95.92	95.92	95.92
EfficientNetB1	95.13	94.51	93.37	93.94	95.55	96.33	95.93
EfficientNetB2	93.67	90.23	**94.58**	92.35	**96.20**	93.06	94.61
EfficientNetB3	95.13	95.06	92.77	93.90	95.18	96.73	95.95
EfficientNetB4	**95.38**	**96.82**	91.57	**94.12**	94.49	**97.96**	**96.19**
EfficientNetB5	93.67	91.67	92.77	92.22	95.06	94.29	94.67
EfficientNetB6	94.16	92.77	92.77	92.77	95.10	95.10	95.10

Chalkiness detection in polished rice. To produce accurate detection of chalkiness area, we first studied the variation of the average IoU with respect to the layer used to generate the heatmaps and the threshold, T, used to binarize the heatmaps when calculating the IoU. The best layer/threshold combination was selected independently for each type of network using both qualitative and quantitative evaluations. Based on preliminary visual inspection of the heatmaps, we observed that heatmaps corresponding to lower level layers in a network result in better approximations of the chalkiness area, possibly because

Table 4.4: *Classification networks: training time and model size. The number following a network's name denotes the number of layers in the network (as in DenseNet-121 or ResNet-101) or the version of the network (as in SqueezeNet-1.0 or EfficientNetB0). All models are trained on AWS p3.2xlarge instances. The training time it took to train each model for 200 epochs is reported in seconds (sec). Model complexity is reported as the number of trainable parameters of the model, as well as the size of the model in MB. The accuracy of each model is also shown, and the best accuracy (Acc.) obtained for each type of model is highlighted in bold font.*

Model	Training time (sec)	Number of parameters	Size (MB)	Acc. (%)
DenseNet-121	1522.88	6955906	28.4	**95.61**
DenseNet-161	2157.04	26,476,418	107.1	95.12
DenseNet-169	1306.20	12,487,810	50.9	94.63
ResNet-18	546.77	11,177,536	44.8	94.63
ResNet-34	719.41	21,285,696	85.3	94.15
ResNet-50	1011.85	23,512,128	94.4	94.88
ResNet-101	1668.41	42,504,256	170.6	**95.12**
ResNet-152	2172.97	58,147,904	233.4	94.88
SqueezeNet-1.0	533.15	736,450	3.0	**95.12**
SqueezeNet-1.1	481.53	723,522	2.9	94.39
VGG-11	2382.44	128,774,530	515.1	94.88
VGG-13	2641.00	128,959,042	515.9	94.39
VGG-16	2745.00	134,268,738	537.1	**95.12**
VGG-19	3079.89	139,578,434	558.4	94.15
EfficientNetB0	1198.53	4,052,126	33.0	95.13
EfficientNetB1	2243.48	6,577,794	53.4	95.13
EfficientNetB2	1882.26	7,771,380	62.9	93.67
EfficientNetB3	2696.21	10,786,602	87.1	95.13
EfficientNetB4	3476.74	17,677,402	142.3	**95.38**
EfficientNetB5	3584.68	28,517,618	229.1	93.67
EfficientNetB6	4946.95	40,964,746	328.3	94.16
Mask R-CNN	14863.00	42,504,256	255.9	N/A

the progressive down-sampling along the convolutional layers of the backbone CNN makes it hard to precisely recover the chalkiness information from the higher level feature maps (Souibgui and Kessentini, 2020). Therefore, for each type of network, we evaluated a lower-level layer (e.g., layer1_2_conv2 for ResNet-101), two intermediate layers (e.g., layer2_0_conv2 and layer3_1_conv2 for ResNet-101), and one high-level layer (e.g., layer4_1_conv3 for ResNet-101). The threshold, T, varied from 10% to 80% in increments of 10. We focused our analysis on ResNet-101 moving forward as this network produced the best segmentation results over-

all. Table 4.5 shows the variation of performance (i.e., average IoU over the set of chalky seed images) with the layer and the threshold for ResNet-101.

Table 4.5: *Variation of the Average IoU (%) with the layer used to generate the heatmaps and the threshold T used to binarize the heatmaps (e.g., T = 20% means that only pixels with values at least 20% of the max pixel value in the image are included in the binary mask). The layers were sampled to include a low-level layer (layer1_2_conv2), a high-level layer (layer4_1_conv3) and two intermediate layers (layer2_0_conv2 and layer3_1_conv2) that showed good results based on a qualitative inspection of the maps. The threshold T is varied from 20% to 80% in increments of 10. The best result and the corresponding layer and threshold are highlighted in bold font.*

Layer	T=20%	T=30%	T=40%	T=50%	**T=60%**	T=70%	T=80%
layer1_2_conv2	0.20	9.90	18.41	26.08	37.53	18.55	18.55
layer2_0_conv2	3.81	19.86	31.53	44.90	**68.11**	18.55	18.55
layer3_1_conv2	1.77	9.59	18.92	28.22	41.59	18.55	18.55
layer4_1_conv3	0.15	10.26	15.43	21.10	29.68	18.55	18.55

As shown in Table 4.5 , we obtained better performance with a lower-intermediate layer (layer2_0_conv2) as opposed to a higher layer as reported in other studies (Selvaraju et al., 2019; **?**), and a threshold of $T = 60\%$ of the highest pixel value, which is larger than the standard $T = 15\%$ (Selvaraju et al., 2019) or $T = 20\%$ (Zhou et al., 2016) thresholds frequently used in prior studies. Similar results were obtained with the other networks.

To gain more insights into the heatmap layer and threshold, Figure 4.5 shows qualitative and quantitative results obtained with Grad-CAM using ResNet-101 as backbone for 10 sample seed images in the test dataset when considering three thresholds (20%, 40%, 60%) and four convolution layers. As can be seen in the figure, seeds with a larger chalky area (e.g., seeds 6 and 10) are less sensitive to the layer chosen, i.e., several layers produce heatmaps with high IoU scores. However, for seeds with a smaller or narrow chalky area, the results are more sensitive to the layer selected and the best results are obtained with the intermediate layer, layer2_0_conv2. Another observation that can be made from Figure 4.5 is that, overall, the lower layers tend to have sharper boundaries as opposed to the higher levels that have softer boundaries, making it harder to find a good threshold. This may be due to the fact that higher levels in the network correspond to lower dimensional feature maps, which no longer

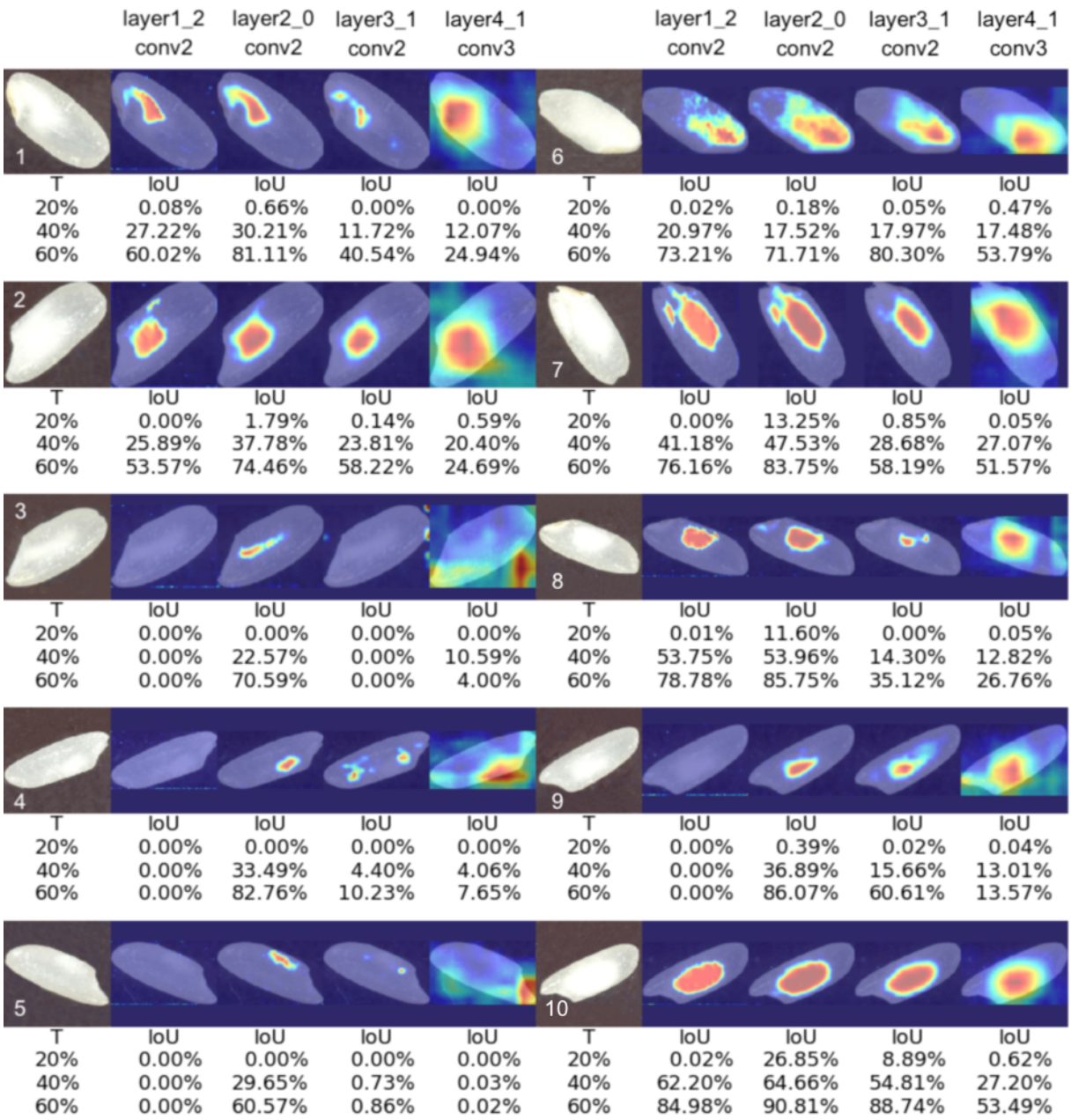


Figure 4.5: *Examples of Grad-CAM (ResNet-101) heatmaps for 10 sample chalky seed images (5 on the left side and 5 on the right side). For each seed, heatmaps corresponding to the following four layers are shown: (1) ResNet101 layer1_2_conv2; (2) ResNet101 layer2_0_conv2; (3) ResNet101 layer3_1_conv2; (4) ResNet101 layer4_1_conv3. The IoU values obtained for three thresholds T (20%, 40% and 60%, respectively) are shown under each heatmap.*

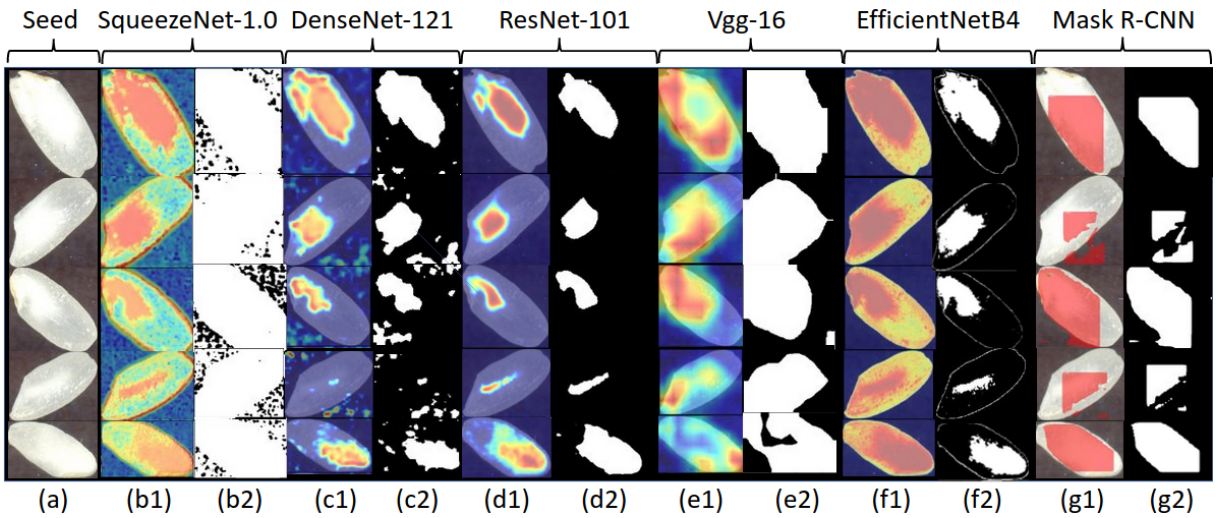


Figure 4.6: *Examples of Grad-CAM heatmaps and corresponding binarized chalkiness masks. (a) Five sample chalky seed images; (b1) SqueezeNet-1.0 Heatmaps; (b2) SqueezeNet-1.0 Masks; (c1) DenseNet-121 Heatmaps; (c2) DenseNet-121 Masks; (d1) ResNet-101 Heatmaps; (d2) ResNet-101 Masks; (e1) VGG-19 Heatmaps; (e2) VGG-19 Masks; (f1) EfficientNetB4 Heatmaps; (f2) EfficientNetB4 Masks; (g1) Mask R-CNN Original Masks ; (g2) Mask R-CNN Binary Masks.*

preserve boundary details when interpolated back to higher dimensions. Figures S3, S4, S5 show similar quantitative and qualitative results produced by SqueezeNet-1.0, DenseNet-121 and VGG-16 networks, respectively, on the same 10 seeds shown in Figure 4.5. Despite the good classification results obtained with these networks, the heatmaps show lighter colors and softer boundaries for the chalkiness area and overall poor chalkiness detection results as compared with the results of ResNet-101. A better understanding regarding this can be gained from Figure 4.6 which shows a side-by-side comparison of the heatmaps produced by different networks and the corresponding binarized chalkiness masks. The masks obtained with Mask R-CNN are also shown.

The same conclusions regarding the superiority of ResNet-101 for chalkiness segmentation are supported by a quantitative evaluation of the networks in terms of localization metrics computed over the whole test set. The results of this evaluation are shown in Table 4.6 for the best performing models for each type of architecture considered as backbone (DenseNet-121, ResNet-101, SqueezeNet-1.0, VGG-19, and EfficientNetB4). For each network, the specific

convolution layer and the threshold used to produce the results are shown in the last two columns of the table, respectively. The results obtained with the Mask R-CNN network, which has ResNet101 as its backbone, are also shown in Table 4.6. As can be seen, the best results are obtained using the ResNet-101 network (for all metrics considered), while the next best results are obtained with DenseNet-121. Among the weakly supervised Grad-CAM networks, the ones that have SqueezeNet-1.0 and VGG-16 as backbones, produce the worse results. The results of the Mask R-CNN network are extremely poor when compared with the results of the Grad-CAM with ResNet-101, DenseNet-121 and EfficientNetB4 backbones, but they are better than those of the Grad-CAM with SqueezeNet-1.0 and VGG-16 as backbones. This shows that the weakly supervised approach is more effective for the chalkiness detection/segmentation problem in addition to being less laborious in terms of data labeling, as compared to the Mask R-CNN segmentation approach.

4.3.2 Chalkiness Classification and Detection in Unpolished Rice

Another objective of this study is to explore the applicability of the Grad-CAM approach to unpolished rice seeds and to study the transferability of the models trained on polished rice to unpolished rice (as unpolished rice seeds can be harder to annotate manually). This is important as researchers working on large breeding populations involving hundreds of lines do not obtain large sample sizes and would not have access to polish a small amount of seeds, which requires models that can effectively operate on unpolished seeds. To address this objective, we performed experiments with three models that use ResNet-101 as their backbone: 1) a model trained on polished seed images, called *polished model*; 2) a model trained on unpolished seed images, called *unpolished model*; and 3) a model originally trained on polished seed images and subsequently fine-tuned on unpolished seed images, called *mixed model*. All models were evaluated on the 240 seed images in the unpolished test set, which were manually annotated in terms of chalkiness area. These images belong to one of the 12 combinations corresponding to the Kati and CO-39 genotypes, i.e., unpolished(12) set.

Table 4.6: *Chalkiness segmentation: results of the weakly supervised Grad-CAM approach with the best performing classification models as backbone. The results of Mask R-CNN with ResNet-101 as backbone are also shown. Only the 166 chalky seed images in the test set were used for chalkiness segmentation evaluation. Performance is reported using the following metrics (as applicable): Ground-Truth Localization Accuracy (GT-known Loc. Acc.), which represents the fraction of ground-truth chalky seed images with IoU* ≥ 0.5*; Localization Accuracy (Loc. Acc.), which represents the fraction of ground-truth chalky images, with IoU* ≥ 0.5*, correctly predicted by the model; Average IoU (Avg. IoU), which represents the average IoU for the set of chalky seed images. To calculate the IoU, the mask of the predicted chalkiness is obtained using a threshold* $T = 60\%$ *of the maximum pixel intensity. The last two columns show the layer that was used for generating the heatmap and the threshold used to binarize the heatmap when calculating IoU, respectively.*

Model	GT-known Loc. Acc. (%)	Loc. Acc. (%)	Avg. IoU (%)	Layer	T (%)
Grad-CAM (DenseNet-121)	51.20 = 85/166	51.20 = 85/166	47.44	features_ denseblock2_ denselayer7_ conv2	60
Grad-CAM (ResNet-101)	84.34 = 140/166	83.13 = 138/166	68.11	layer2_0_ conv2	60
Grad-CAM (SqueezeNet-1.0)	7.83 = 13/166	7.83 = 13/166	20.25	features_4 _expand1x1	60
Grad-CAM (VGG-16)	7.23 = 12/166	7.23 = 12/166	24.92	features_ module_5	60
Grad-CAM (EfficientNetB4)	28.92 = 48/166	28.92 = 48/166	35.40	stem _conv	50
Mask R-CNN (ResNet-101)	18.67 = 31/166	N/A	29.63	N/A	N/A

The training and developments sets used to train the unpolished and mixed models belong to the unpolished(12) set as well (see Table 4.2). Classification results for the three models are shown in Table 4.7, while segmentation results are shown in Table 4.8. As can be seen in Table 4.7, the mixed model performs the best overall in terms of classification metrics, although the unpolished model has similar performance for the chalky class. However, as Table 4.8 shows, the unpolished model is by far the most accurate in terms of segmentation metrics, while the polished model is the worst.

To visually illustrate the output of each model, Figure 4.7 shows the chalkiness prediction masks of the polished, unpolished and mixed models for four unpolished seeds. The polished

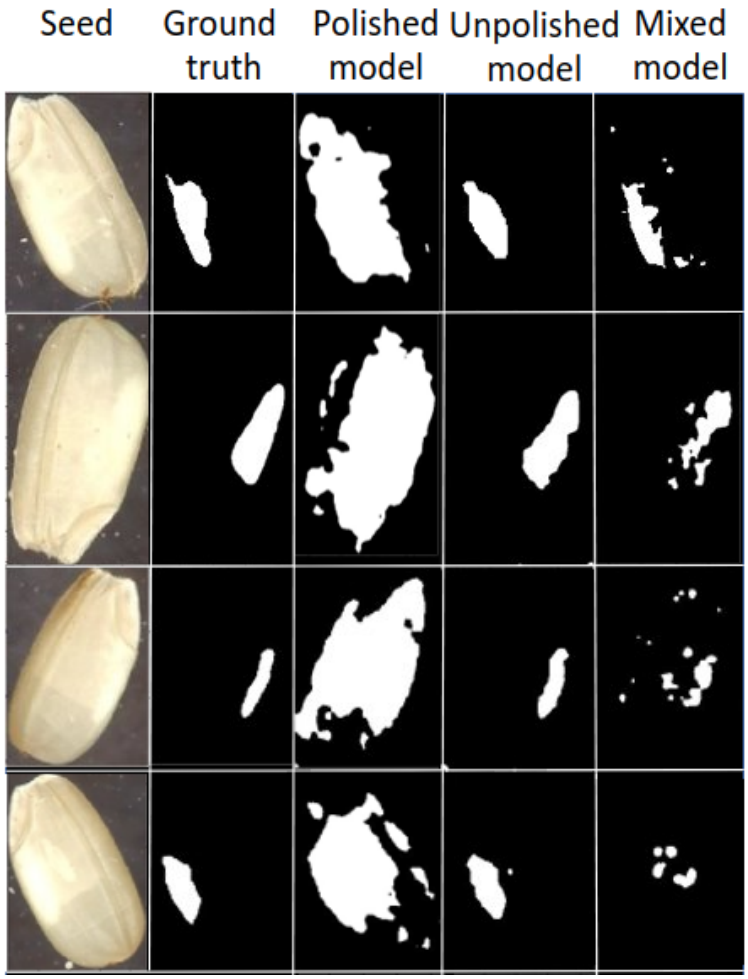


Figure 4.7: *Examples of chalkiness binary masks for four unpolished rice grains. The binary masks obtained from the Grad-CAM heatmaps (with ResNet-101 as backbone) using a threshold $T = 60\%$ are shown form the polished, unpolished and mixed models, respectively, by comparison with the ground truth binary mask.*

Table 4.7: *Classification results on unpolished rice when ResNet-101 is used as backbone in the weakly supervised Grad-CAM approach. Three models are evaluated: 1) polished model trained on polished rice images; 2) unpolished model trained on Unpolished (12); 3) mixed model, obtained by further training the polished model using the Unpolished (12) images. Performance is reported in terms of Accuracy (Acc.), Precision (Pre.), Recall (Rec.) and F1 measure (F1). Precision, Recall and F1 measure values are reported separately for the Chalky and Non-Chalky classes. All three models are evaluated on the test subset corresponding to the Unpolished (12) rice images. The best performance for each type of model for each metric is highlighted using bold font.*

ResNet-101	Acc.(%)	**Chalky**			**Non-Chalky**		
		Pre. (%)	Rec.(%)	F1(%)	Pre.(%)	Rec.(%)	F1(%)
polished	63.01	0.00	0.00	0.00	**63.01**	**100.00**	**77.31**
unpolished	83.43	**98.50**	82.19	89.61	43.65	91.67	59.14
mixed	**84.20**	98.08	**83.45**	**90.18**	44.77	89.17	59.61

model largely over-estimates the chalkiness area given the opaque nature of the unpolished seeds, as opposed to the translucent appearance of the polished seeds. The mixed model improves the masks but not as much as the unpolished model that is trained specifically on unpolished rice seeds. Together, these results suggest that not much knowledge can be transferred directly from the polished images to unpolished images, as the appearance of the chalkiness is different between polished and unpolished seeds. The results can be improved with the mixed model which fine-tunes the polished models on unpolished rice, although the fine-tuned models still lag behind the models trained directly on unpolished rice. Hence, models developed using polished or unpolished grains needs to be used based on the objective with poor transferability across these two categories.

4.3.3 Answers to the research questions and error analysis

To answer RQ1, we evaluated several CNN architectures in terms of classification accuracy, memory and time requirements, and also chalkiness detection performance in polished rice. While the architectures studied have comparable classification performance, the ResNet-101 network was found to be superior with respect to chalkiness detection in polished rice, and has relatively small memory and time requirements. Furthermore, we compared the best weakly

Table 4.8: *Chalkiness segmentation results of the weakly supervised Grad-CAM approach with ResNet-101 as backbone on unpolished rice. Only 240 chalky seed images in the Unpolished (12) test set were used for chalkiness segmentation evaluation. Performance is reported using the following metrics: Ground-Truth Localization Accuracy (GT-known Loc. Acc.), which represents the fraction of ground-truth chalky seed images with IoU* ≥ 0.5*; Localization Accuracy (Loc. Acc.), which represents the fraction of ground-truth chalky images, with IoU* ≥ 0.5*, correctly predicted by the model; Average IoU (Avg. IoU), which represents the average IoU for the set of chalky seed images. To calculate the IoU, the mask of the predicted chalkiness is obtained using a threshold* $T = 60\%$ *of the maximum pixel intensity. The last two columns show the layer that was used for generating the heatmap and the threshold used to binarize the heatmap when calculating IoU, respectively.*

Grad-CAM (ResNet-101)	GT-known Loc. Acc. (%)	Loc. Acc. (%)	Avg. IoU (%)	Layer	T (%)
polished model	7.92 = 19/240	0 = 0/240	26.79	layer2_0_conv2	60
unpolished model	63.75 = 153/240	63.75 = 153/240	51.76	layer2_0_conv2	60
mixed model	20.42 = 49/240	20.42 = 49/240	29.91	layer2_3_conv2	60

supervised Grad-CAM models with the Mask R-CNN segmentation model to answer RQ2 and found that the Grad-CAM models performed better than Mask R-CNN, which needs more expensive pixel level annotation. Overall, the chalkiness detection results obtained for polished rice are remarkably good, with an average IoU of 68.11%, GT-known accuracy of 83.34% and localization accuracy of 83.13%. Finally, to answer RQ3, we used Grad-CAM models trained on polished rice, unpolished rice, and a mix of polished and unpolished rice and evaluated them on unpolished rice. When studying the transferability of the models trained on polished rice to unpolished rice, we found that fine-tuning on unpolished rice is necessary. In fact, models trained directly on the unpolished rice performed the best in our study. More specifically, our evaluation on unpolished rice grain images showed that the best model trained directly with unpolished rice had an average IoU of 51.76%, while both the GT-known accuracy and the localization accuracy were 63.75%. It is not surprising that the models perform better on polished rice as chalkiness is easier to detect after the interfering aluerone layer is removed through milling

While the use of the Grad-CAM approach for rice chalkiness segmentation was extremely successful, one challenge that we encountered was the tuning of the layer to be used for generating the heatmaps as well as the threshold for producing the binary masks for chalkiness area. Our goal was to find a good overall layer and a good overall threshold for a model to avoid the pitfall of tuning the threshold for each type of rice seed. Our analysis showed that a lower layer generally results in better chalkiness detection. One explanation for this is that higher levels undergo more extensive down-sampling (through successive applications of pooling layers) and this causes loss of information that cannot be recovered in the chalkiness heatmaps. Regarding the threshold for binarization, our results showed that a higher threshold (e.g., $T = 60\%$) produces better overall results.

Error analysis of the polished models revealed several sources of errors that lead to disagreement between model predictions and ground truth annotations. Such sources are illustrated in Figure 4.8 and include: (a) inconsistencies in the way chalkiness is manually annotated due to the soft/fuzzy boundaries of chalkiness (as opposed to binary chalky versus non-chalky boundaries); (b) scratches or marks (referred to as noise) on the chalkiness area are interpreted as non-chalkiness and lead to mismatches with the ground truth annotations in terms of IoU metric; (c) irregular chalkiness shapes also make it hard to annotate chalkiness very precisely; (d) abrasion stains that are recognized as chalkiness (white dots on the right in the figure) despite the fact that the Grad-CAM model uses deeper feature maps that presumably miss some "details"; (e) irregular shape and fuzzy boundaries affect the ground truth annotations and consequently the predictions in unpolished rice as well. Despite such errors, we found that the best Grad-CAM model for unpolished rice, trained on the Kati and CO-39 genotypes, can generalize well to unpolished rice grains from the other genotypes included in the biological experiment. Supplementary Figure S6 shows the prediction results of the unpolished model on 12 rice grains randomly selected from the genotypes not used in the training, together with their manual annotations.

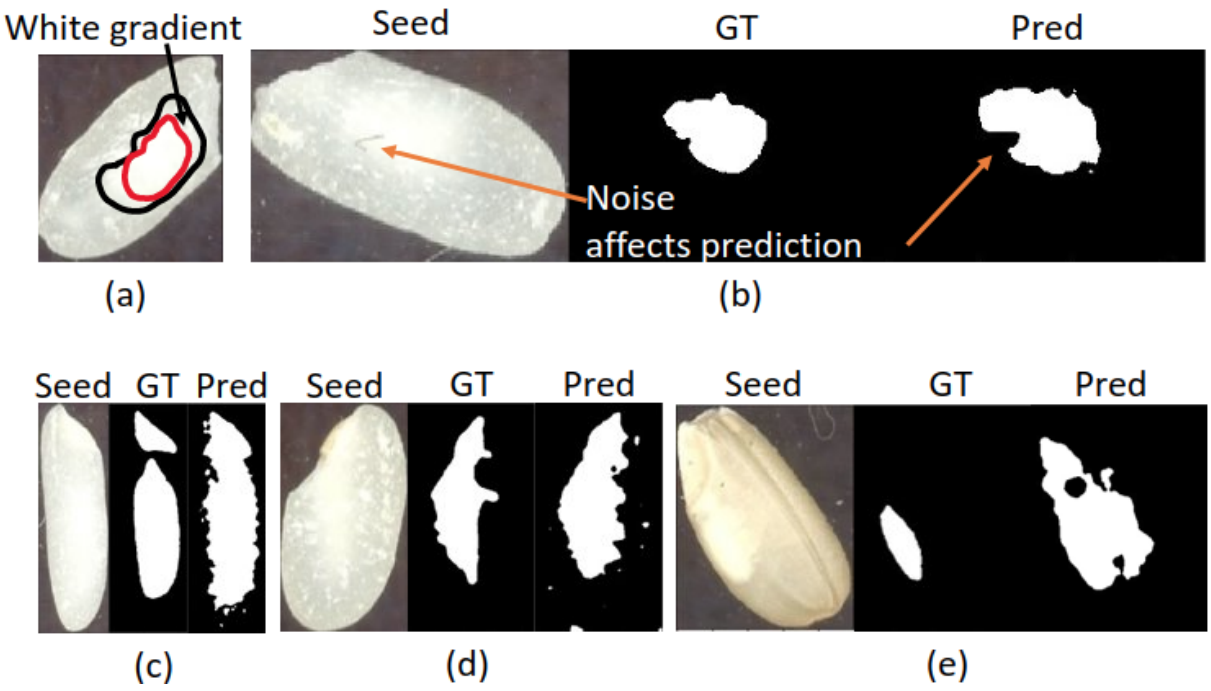


Figure 4.8: *Sources of errors for the Grad-CAM models. Images (a)-(d) correspond to polished rice, while image (e) corresponds to unpolished rice. The sources of error can be summarized as: (a) Inconsistencies in the way chalkiness is manually annotated, due to the white gradient nature of chalkiness; (b) Scratches or marks (referred as noise) on the chalkiness area can be interpreted as non-chalkiness; (c) Irregular chalkiness shape makes it hard to annotate chalkiness very precisely; (d) Abrasion stains can be recognized as chalkiness (white dots on the right in the figure); (e) Irregular shape and fuzzy boundaries affect the ground truth annotations and the predictions in unpolished rice as well.*

4.3.4 Tool availability and time requirements

In terms of time requirements, our experiments showed the average time for training a ResNet-101 model on an EC2 p3-2xlarge instance available from AWS is 1668.41 seconds, as shown in Table 4.4, and no human intervention is required during that time. Once the model is trained, the average time to predict the label of a new image and create a chalkiness heatmap is less than one second using an EC2 p2-xlarge instance. Given these time requirements and assuming that a relatively large number of images, on the order of thousands, need to be annotated for genetic mapping studies, our models could be extremely cost-effective and help save significant human efforts and time that would otherwise be invested in the manual annotation.

4.3.5 Development of rice with less chalk under future hotter climate

Quantifying rice chalkiness rapidly and accurately continues to be a limitation for capturing the degree of chalkiness across a wide range of genetic backgrounds due to the lack of a high throughput phenotyping tool. Developing such a tool is important and timely as the proportion of chalky grains are bound to increase under warming scenarios, particularly with increasing night temperatures (Impa et al., 2021; Shi et al., 2013). We used the tool developed based on Grad-CAM to determine the percent chalkiness area and the chalkiness score for each of the 13,101 unpolished rice grains extracted from the original scanned images. As opposed to the chalkiness area, which is obtained based on a binary map, the chalkiness score considers the intensity of chalkiness for each pixel, with red indicating greater proportion of chalk per pixel and blue indicating the least proportion of chalk per pixel (Figures 4.5 and 4.6). Subsequently, we aggregated the percent chalkiness and the chalkiness score per sample (i.e., for each combination genotype/tiller/treatment). Using the aggregates, we analyzed differences between genotypes, tiller and treatment in terms of chalkiness in three

scenarios. In scenario 1, where the chalkiness was determined using the coarse chalky versus non-chalky classification of the grains, analysis based on the number of grains with and without chalk resulted in a poor analytical resolution and failed to detect any differences or significant interaction effects (Table T3). In scenario 2, analysis based on the proportion of area of chalkiness determined from the Grad-CAM binarized heatmaps improved the prediction power where apart from genotype (G) main effect, the interaction effects of HNT treatment (T) *G, G* panicle type (P), and T*G*P interaction effects were significant (Table 4.9). This finding indicated that the approach was able to detect the differential proportion of chalkiness in different tillers across genotypes under HNT exposure during grain-filling. Using this approach, genotypic differences in the proportion of accumulation of chalkiness were observed with IR1561 and WAS-174 which recorded an increase of chalkiness in grains in primary and other panicles as compared to main tiller under HNT, while the same was reduced in IR-22 and Kati and was not affected in CO-39 and Oryzica (Table 4.9). Percent change in proportion of chalkiness under HNT in primary and other panicles compared to main panicle ranged from -0.89% in IR1561 to 122% in WAS-174. Grains from both primary and other panicles recorded an increase in proportion of chalkiness by 63 and 122%, respectively, compared to main panicle under HNT in WAS-174 (Table 4.9). In scenario 3, the chalkiness score was calculated using the pixel intensity in the chalkiness heatmaps produced by Grad-CAM and analysis of variance for chalkiness score revealed a significant effect of G, T*G, G*P and T*G*P further indicating an improvement in prediction potential for chalkiness among genotypes, treatments and tiller types (Table 4.9).

Similar to proportion of chalkiness area, chalkiness score showed an increase under HNT compared to control in IR1561 and WAS-174 in primary and other panicles, while the same was reduced in IR-22 and Kati (Table 4.9). Among the genotypes, WAS-174 recorded highest percent increase in chalkiness score under HNT in grains from primary (74%) and other panicles (59%) compared to main panicle (Table 4.9). In contrast, Oryzica recorded an increase in chalkiness score under HNT in grains from primary (46%) and other panicles

(99%) compared to main panicle. Genotypes like CO-39, IR1561 and IR-22 showed minimal changes in chalkiness score between tillers under HNT (Table 4.9). In summary, identifying and using such germplasm (for example, CO-39 and Oryzica) with minimal chalkiness even under HNT will help develop rice varieties that can sustain quality under future warming scenarios without having a negative impact of economic revenue of the rice farmers. In addition, the ability to obtain the level of chalkiness in less than one second per image and in batch mode allows these models to be used efficiently as a high-throughput phenotyping tool for capturing chalkiness in large breeding populations and to efficiently incorporate genetics leading to low grain chalkiness into ongoing global rice breeding programs.

4.4 Conclusions

In this study, we presented the application of a high throughput deep learning tool to detect the chalkiness area in polished and unpolished rice grains. To avoid the need for cumbersome pixel-level annotation, we used a weakly supervised segmentation approach, Grad-CAM, which addresses the problem as a binary classification task and subsequently uses the gradients of the grain chalk to produce a chalkiness heatmap. Experimental results showed that it is possible to use the Grad-CAM model with ResNet-101 as a backbone to generate accurate chalkiness heatmaps for both polished and unpolished rice grains. However, the analysis also showed that detecting rice chalkiness is easier in polished rice as compared to unpolished rice and that the polished models are not directly transferable to unpolished rice. Our study shows that weakly supervised deep learning models can be used to assist research in both phenotyping and rice quality control in several ways: (i) perform high-throughput rice seed image analysis to identify chalky seeds and generate chalkiness maps, (ii) replace the expensive error-prone human annotations with rapid and continuous annotations without compromising the accuracy, and (iii) provide quantitative measures for chalkiness area. We successfully demonstrated the application of this tool in accurately capturing the HNT

induced differential level of chalkiness in different tillers in rice. The models trained in this study are made publicly available. Being already trained, they will be easy-to-use, scalable and can be readily utilized in ongoing rice breeding programs, without requiring researchers to have computer science or machine learning expertise.

Additional Files

Additional file 1: Fig. S1. Steps for rice chalk seed image scanning.

Additional file 2: Fig. S2. Image scan of rice seeds.

Additional file 3: Table T1. Polished rice seeds statistics.

Additional file 4: Table T2. Unpolished rice seeds statistics.

Additional file 5: Fig. S3. Examples of Grad-CAM/SqueezeNet-1.0 heatmaps.

Additional file 6: Fig. S4. Examples of Grad-CAM/DenseNet-121 heatmaps.

Additional file 7: Fig. S5. Examples of Grad-CAM/VGG-19 heatmaps.

Additional file 8: Fig. S6. Examples of predictions on unpolished rice.

Additional file 9: Table T3. Number of grains with and without chalk.

Availability of data and materials

The datasets generated and analysed during the current study are available on GitHub, https://github.com/cwang16/Phenotyping-of-Chalkiness-in-Rice.

Funding

We thank the financial support by National Science Foundation, USA, Award No. 1736192, to Krishna Jagadish, Kansas State University. Contribution number 21-318-J from the Kansas Agricultural Experiment Station.

	Main panicle		Primary panicle		Other panicle	
Chalkiness (% area)	CNT	HNT	CNT	HNT	CNT	HNT
CO-39	7.54 ±0.8	8.17 ±1.0	6.95 ±0.6	7.73 ±0.3	8.00 ±1.8	7.19 ±1.1
IR1561	13.35 ±2.5	16.22 ±3.0	8.21 ±0.2	16.37 ±2.4	8.52 ±1.1	12.35 ±1.2
IR-22	8.02 ±0.8	6.33 ±0.5	9.36 ±2.2	5.27 ±0.5	5.89 ±0.9	5.30 ±0.6
Kati	7.39 ±0.7	7.44 ±0.4	13.56 ±1.2	10.34 ±2.5	14.32 ±1.9	10.70 ±2.5
Oryzica	10.61 ±1.4	10.75 ±1.8	5.32 ±0.5	5.64 ±0.6	5.05 ±0.6	4.83 ±1.4
WAS-174	7.25 ±2.0	5.76 ±1.7	5.91 ±0.4	9.39 ±2.6	4.44 ±0.8	12.81 ±1.4
Chalkiness score	CNT	HNT	CTN	HNT	CTN	HNT
CO-39	0.07518 ±0.008	0.06827 ±0.009	0.07157 ±0.004	0.07780 ±0.003	0.07449 ±0.016	0.07119 ±0.010
IR1561	0.10415 ±0.015	0.12933 ±0.026	0.07026 ±0.002	0.14653 ±0.032	0.07006 ±0.006	0.11615 ±0.012
IR-22	0.07276 ±0.003	0.06294 ±0.006	0.09692 ±0.017	0.05694 ±0.006	0.05916 ±0.012	0.05686 ±0.007
Kati	0.09238 ±0.009	0.09252 ±0.002	0.17087 ±0.017	0.13309 ±0.032	0.16940 ±0.018	0.13586 ±0.029
Oryzica	0.14890 ±0.024	0.16370 ±0.029	0.08302 ±0.011	0.08862 ±0.006	0.07928 ±0.014	0.08246 ±0.026
WAS-174	0.09386 ±0.024	0.07700 ±0.023	0.08449 ±0.005	0.13393 ±0.038	0.06039 ±0.011	0.18932 ±0.018

Table 4.9: *Percentage chalkiness area and chalkiness score were obtained for individual seeds randomly selected across treatments and genotypes. A three-way analysis of variance for these traits (Chalkiness Area (%) and Score) were performed under completely randomized design (CRD) using PROC GLM procedure in SAS. Means were separated using HSD (Tukey's Studentized Range) test at p=0.05. Table includes mean and ± SEM for three way comparison. Chalkiness area (%) was significantly affected by genotype (G) ($p < 0.001$), treatment (T) x G ($p < 0.001$) and G x panicle type (P) ($p < 0.001$) and T x G x P ($p < 0.001$) interaction effects. Chalkiness score was significantly affected by G ($p < 0.001$), T x G ($p < 0.016$), G x P ($p < 0.001$) and T x G x P ($p = 0.03$) interaction effects.*

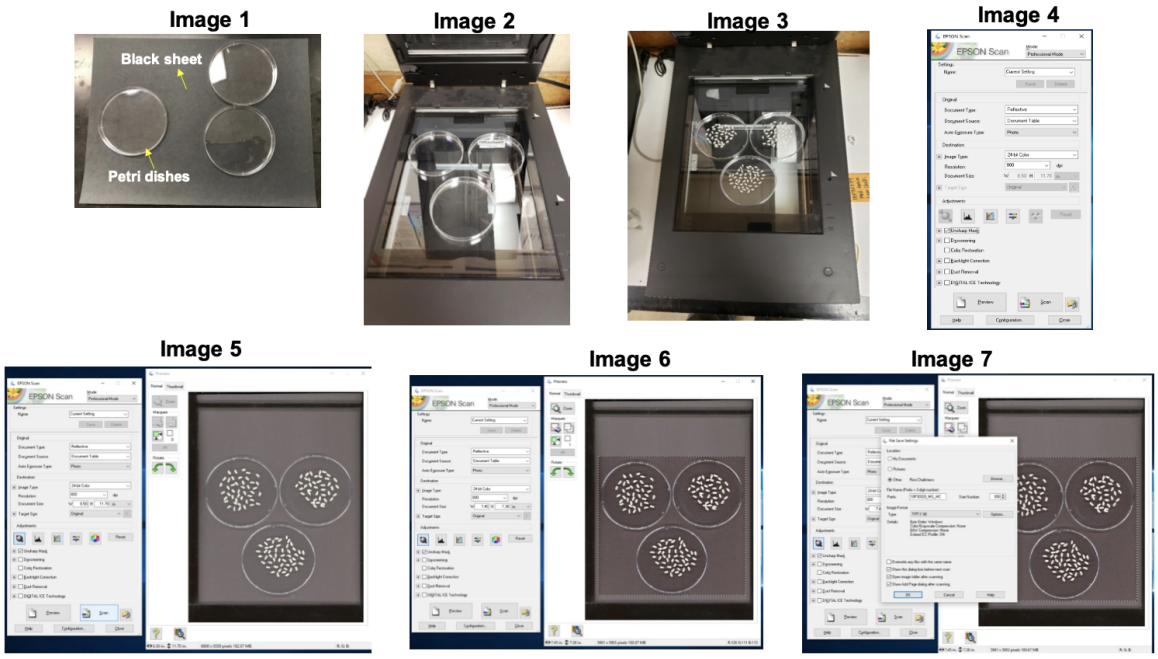


Figure S1: *Steps for rice chalk seed image scanning using Epson Perfection V800 photo scanner.*

Image 1: Showing the materials required: Rice seeds (dehusked), Epson perfection V800 photo scanner attached with the computer, Petri-dishes, non-metal forceps, a black sheet of paper (A4).

Steps for image scanning of both polished and unpolished grains

- Place the transparent Petri-dishes on the scanner away from the edges of the scanner, but not touching each other **(See image 2)**
- Spread the rice seeds in the middle of the Petri-dishes **(See image 3)**
- Place the black sheet of paper over dishes and shut the scanner lid
- Open the computer and scanner
- Click the scanner software desktop icon

Scanner setting (See image 4)

- Start Epson Scan and select Professional Mode as the Mode setting
- Select these settings under the Original section:
  - Document Type setting = "Reflective"
  - Select the Document source = "Document Table"
  - Select Auto Exposure type = "Photo."
- Select these settings in the Destination section:
  - Select the image type = "24-bit Color"
  - Set resolution dpi as - 800 and click OK (do not change the dpi, keep this as constant across scans)
- Select these settings in the Adjustments section:
  - Check Unsharp Mask
  - Uncheck all other options.
- Click 'PREVIEW' to see the scan
- If the dishes are all within the scan area and all of the rice grains are away from the edges of the dishes, then continue to the next step. If not, readjust the location of dishes and/or rice and click preview again. (**See image 5**)
- Using the mouse, left click and drag to create a box around the Petri-dishes in the scan preview. This will set the scanning area for the final scan. Should include all borders of the Petri-dishes but not excessive blank area outside of the dishes. (**See image 6**)
- Return to the Main window and Click 'SCAN'.

- In the Location, setting, click on “Other” and then Browse... to select folder location the images should be saved. (**See image 7**)
- In the File Name section type file name in Prefix and set start number to 1. Start Number will automatically increase with each scan.
- In the Image Format Section, choose file type TIFF (*.tif)
- In the bottom section:
  - Uncheck “Overwrite any files with the same name.”
  - Check “Show this dialog box before next scan.”
  - Check “Open image folder after scanning”
  - Check “Show Add Page dialog after scanning”
- Click OK.
- After scan, return rice back to packets and begin again at the top of the page until the number of replications has been reached, or begin on next sample.

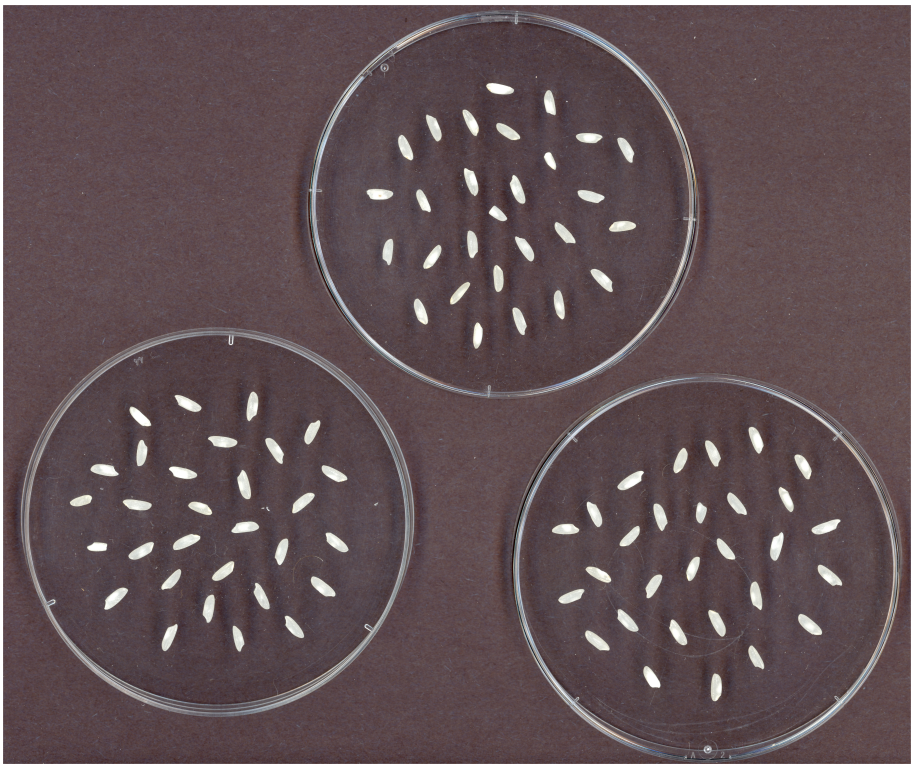


Figure S2: *Image scan of rice seeds spread on three Petri dishes covered with a black background. The seeds on the three dishes correspond to one size/chalkiness combination for polished rice, and one genotype/tiller/condition for unpolished rice, respectively. Three such images were obtained for each combination resulting in three replicates, each with different seeds.*

Table T1: *Polished rice grains statistics. For each combination of grain size (column 1), chalkiness degree (column 2) and replicate (column 3), the total number of grains in the corresponding high resolution image and the number of grains used in the analysis are shown in columns 4 and 5, respectively. Columns 6 and 7 show the number of (used) grains annotated as chalky and non-chalky, respectively.*

Grain size	Chalkiness degree	Replicate	Grains original	Grains used	Chalky	Non-chalky
long	low	1	76	76	27	49
long	low	2	96	96	27	69
long	low	3	88	88	25	63
long	medium	1	96	96	21	75
long	medium	2	102	102	31	71
long	medium	3	85	85	23	62
long	high	1	87	81	61	20
long	high	2	91	91	73	18
long	high	3	88	88	65	23
medium	low	1	90	90	21	69
medium	low	2	97	97	17	80
medium	low	3	80	80	17	63
medium	medium	1	106	106	25	81
medium	medium	2	105	105	33	72
medium	medium	3	99	99	24	75
medium	high	1	66	66	46	20
medium	high	2	100	100	61	39
medium	high	3	102	99	63	36
		Total	1654	1645	660	985

Table T2: *Unpolished rice grains statistics. For each combination of genotype (column 1), tiller (column 2), condition (column 3) and replicate (column 4), the total number of grains in the corresponding high resolution image and the number of grains used in the analysis are shown in columns 5 and 6, respectively. Columns 7 and 8 show the number of (used) grains annotated as chalky and non-chalky, respectively.*

Genotype	Tiller	Condition	Replicate	Grains original	Grains used	Chalky	Non-chalky
Kati	Main	CNT	1	139	138	122	16
		CNT	2	160	160	101	59
		CNT	3	95	95	63	32
		HNT	1	169	165	158	7
		HNT	2	161	160	139	21
		HNT	3	152	147	119	28
	Primary	CNT	1	152	152	121	31
		CNT	2	137	137	92	45
		CNT	3	150	150	117	33
		HNT	1	148	148	145	3
		HNT	2	146	146	122	24
		HNT	3	149	149	136	13
	Other	CNT	1	133	133	115	18
		CNT	2	136	136	88	48
		CNT	3	132	132	79	53
		HNT	1	154	152	145	7
		HNT	2	166	166	132	34
		HNT	3	148	148	138	10
CO-39	Main	CNT	1	166	166	104	62
		CNT	2	86	86	5	81
		CNT	3	125	123	98	25
		HNT	1	164	164	155	9
		HNT	2	89	89	13	76
		HNT	3	186	186	130	56
	Primary	CNT	1	167	167	128	39
		CNT	2	150	148	48	100
		CNT	3	145	145	106	39
		HNT	1	148	148	138	10
		HNT	2	149	148	99	49
		HNT	3	155	150	118	32
	Other	CNT	1	150	150	134	16
		CNT	2	152	151	117	34
		CNT	3	154	154	110	44
		HNT	1	152	152	130	22
		HNT	2	147	147	55	92
		HNT	3	145	145	114	31

Table T2: *Continued*

Genotype	Tiller	Condition	Replicate	Grains original	Grains used	Chalky	Non-chalky
IR-22	Main	CNT	1	97	97	0	97
		CNT	2	86	86	0	86
		HNT	1	97	96	0	96
		HNT	2	50	50	0	50
		HNT	3	43	43	0	43
	Primary	CNT	1	133	133	1	132
		CNT	2	156	155	0	155
		HNT	1	152	152	1	151
		HNT	2	90	90	0	90
		HNT	3	95	95	1	94
	Other	CNT	1	155	155	0	155
		CNT	2	150	150	0	150
		CNT	3	105	104	0	104
		HNT	1	152	152	1	151
		HNT	2	152	150	0	150
		HNT	3	148	148	0	148
IR-1561	Main	CNT	1	124	124	0	124
		CNT	2	97	97	0	97
		CNT	3	116	116	1	115
		HNT	1	92	92	1	91
		HNT	2	85	85	0	85
		HNT	3	26	26	1	25
	Primary	CNT	1	152	151	0	151
		CNT	2	148	148	0	148
		CNT	3	149	149	0	149
		HNT	1	104	103	3	100
		HNT	2	161	157	1	156
		HNT	3	21	21	0	21
	Other	CNT	1	151	151	0	151
		CNT	2	143	142	0	142
		CNT	3	148	148	0	148
		HNT	1	160	160	24	136
		HNT	2	151	151	1	150
		HNT	3	146	146	1	145

Table T2: *Continued*

Genotype	Tiller	Condition	Replicate	Grains original	Grains used	Chalky	Non-chalky
WAS-174	Main	CNT	1	116	115	6	109
		CNT	2	180	180	0	180
		CNT	3	195	194	0	194
		HNT	1	127	127	6	121
		HNT	2	120	120	3	117
		HNT	3	155	155	0	155
	Primary	CNT	1	145	145	6	139
		CNT	2	159	154	0	154
		CNT	3	152	150	0	150
		HNT	1	156	156	4	152
		HNT	2	152	149	15	134
		HNT	3	157	157	2	155
	Other	CNT	1	157	157	4	153
		CNT	2	150	150	0	150
		CNT	3	141	141	1	140
		HNT	1	155	155	9	146
		HNT	2	141	141	3	138
		HNT	3	153	153	0	153
Oryzica	Main	CNT	1	54	54	3	51
		CNT	2	40	40	0	40
		CNT	3	55	55	0	55
		HNT	1	25	25	6	19
		HNT	2	52	52	0	52
		HNT	3	64	64	0	64
	Primary	CNT	1	77	77	3	74
		CNT	2	59	59	1	58
		CNT	3	71	71	0	71
		HNT	1	32	32	12	20
		HNT	2	100	100	2	98
		HNT	3	89	89	1	88
	Other	CNT	1	141	141	8	133
		CNT	2	89	89	0	89
		CNT	3	68	68	0	68
		HNT	1	34	34	15	19
		HNT	2	41	41	2	39
		HNT	3	55	55	2	53
		Total		13149	13101	4085	9016

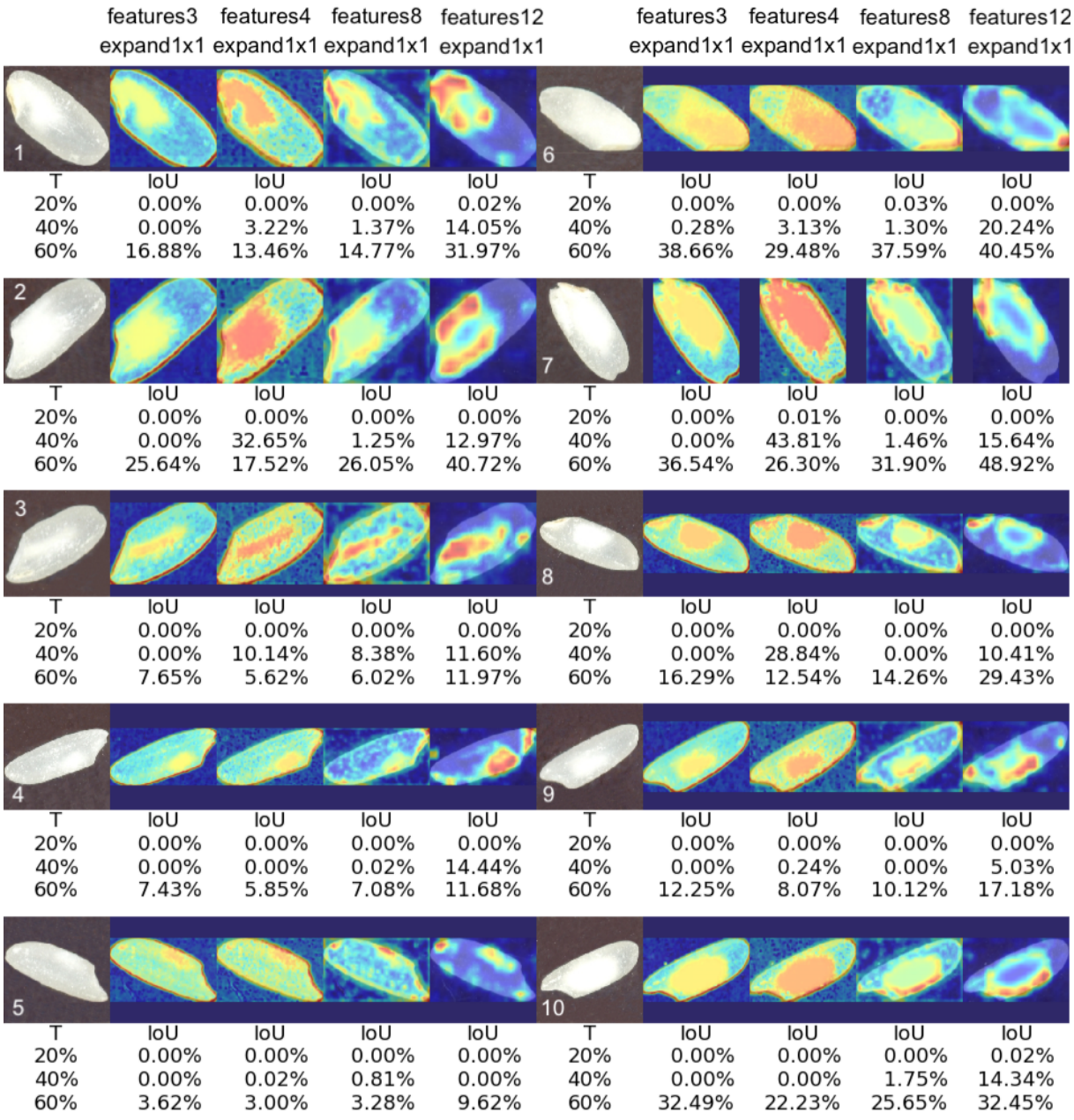


Figure S3: *Examples of Grad-CAM (SqueezeNet-1.0) heatmaps for 10 sample chalky seed images (5 on the left side and 5 on the right side). For each seed, heatmaps corresponding to the following four layers are shown: (1) squeezenet1_0 features_3_expand1x1; (2) squeezenet1_0 features_4_expand1x1; (3) squeezenet1_0 features_8_expand1x1; (4) squeezenet1_0 features_12_expand1x1. The IoU values obtained for three thresholds T (20%, 40% and 60%, respectively) are shown under each heatmap.*

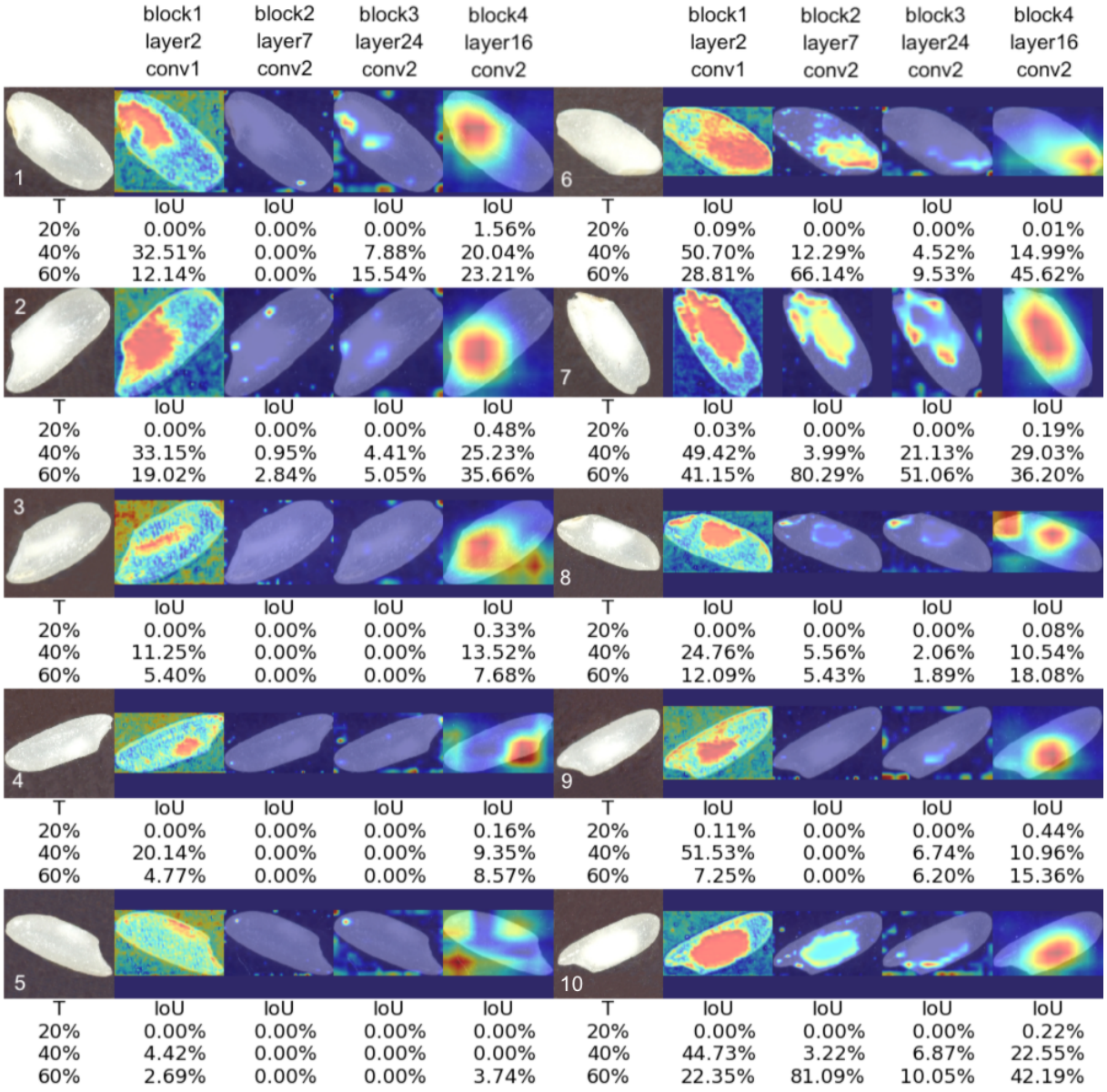


Figure S4: *Examples of Grad-CAM (DenseNet-121) heatmaps for 10 sample chalky seed images (5 on the left side and 5 on the right side). For each seed, heatmaps corresponding to the following four layers are shown: (1) densenet121 denseblock1_denselayer2_conv1; (2) densenet121 denseblock2_denselayer7_conv2; (3) denseblock3_denselayer24_conv2; (4) denseblock4_denselayer16_conv2. The IoU values obtained for three thresholds T (20%, 40% and 60%, respectively) are shown under each heatmap.*

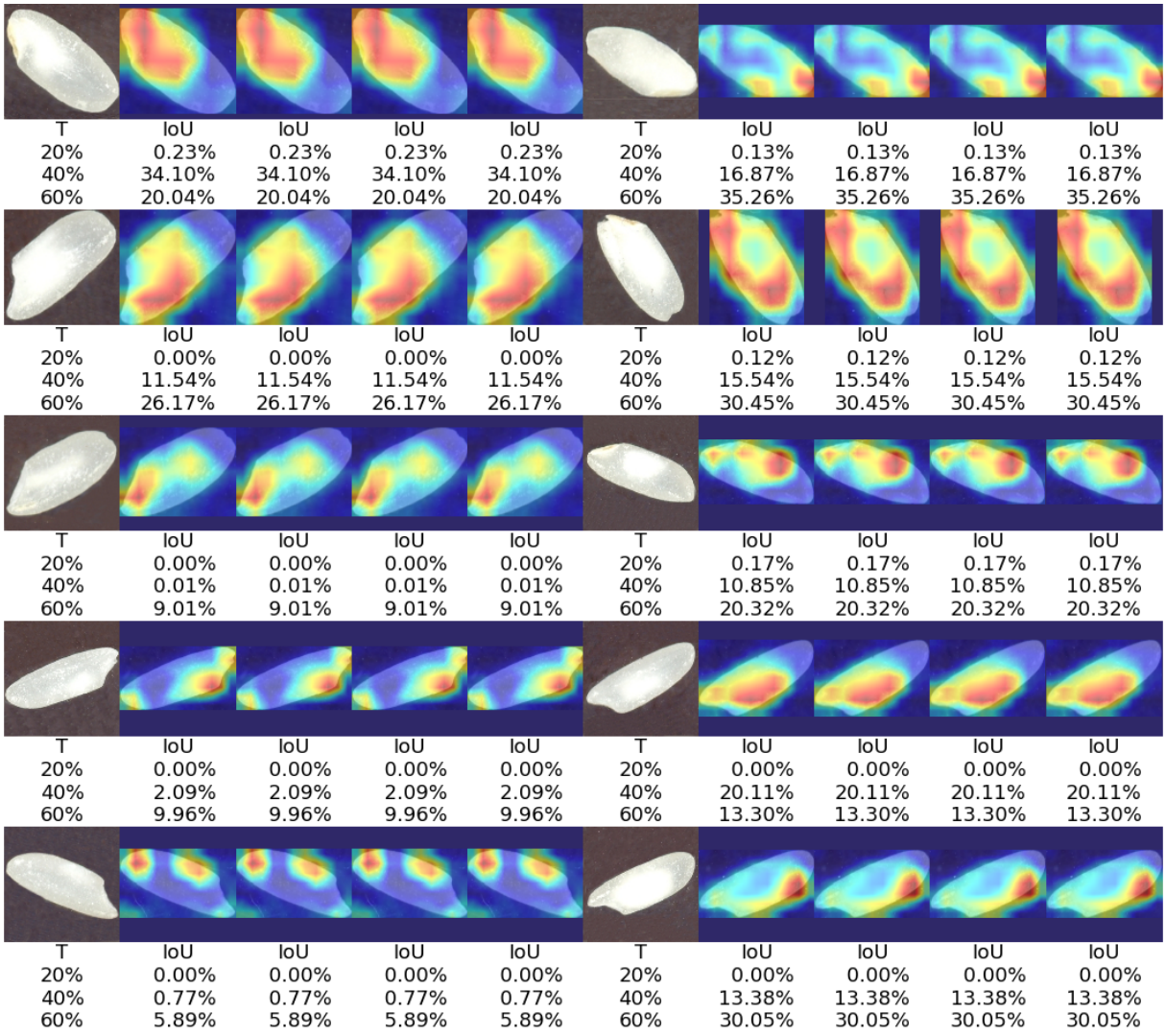


Figure S5: *Examples of Grad-CAM (VGG-19) heatmaps for 10 sample chalky seed images (5 on the left side and 5 on the right side). For each seed, heatmaps corresponding to the following four layers are shown: (1) vgg16 features_module_5; (2) vgg16 features_module_10; (3) vgg16 features_module_19; (4) vgg16 features_module_28. The IoU values obtained for three thresholds T (20%, 40% and 60%, respectively) are shown under each heatmap.*

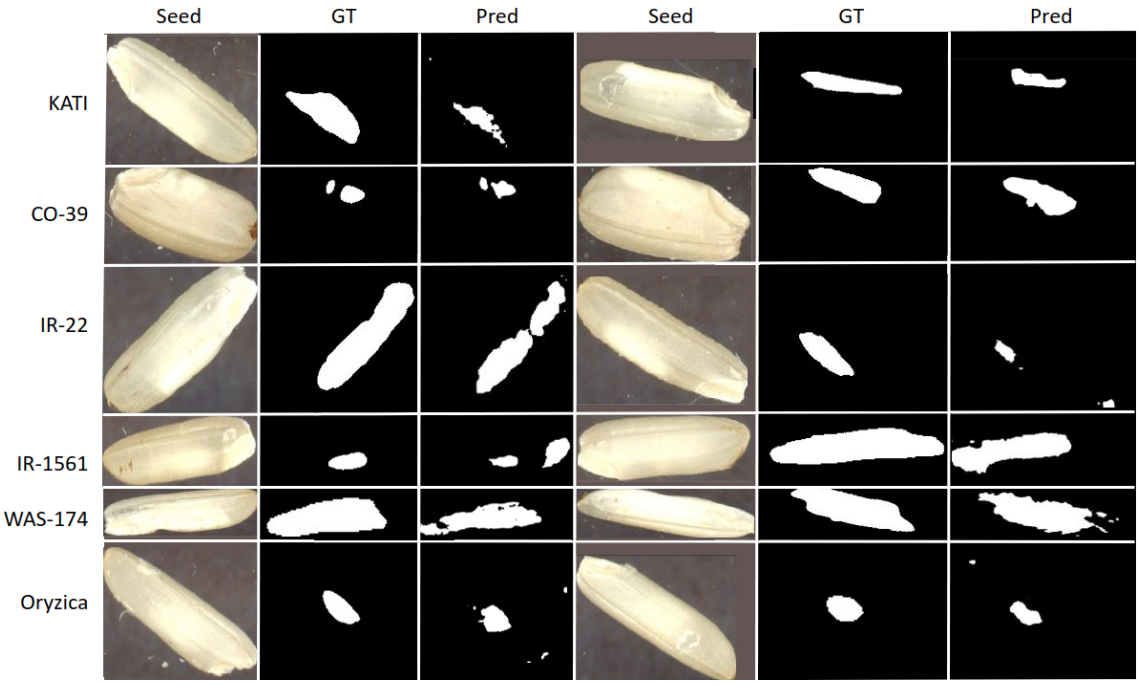


Figure S6: *Examples of binary masks predicted by Grad-CAM on 12 unpolished images, by comparison with the ground truth binary masks. The Grad-CAM model used was trained on the 12 combinations corresponding to CO-39 and Kati genotypes. The images shown are randomly selected from the other four genotypes not included in the training. These examples show that the unpolished model generalizes well from some genotypes to others.*

	% Chalkiness	
Genotype	**Control**	**HNT**
CO-39	63.2 ±9	68.8 ±9
IR-1561	0.1 ±0.1	2.76 ±1.6
IR-22	0.1 ±0.09	0.26 ±0.14
Kati	72.6 ±3.5	89.4 ±2.3
Oryzica	1.87 ±0.83	13.0 ±5.8
WAS-174	1.4 ±0.7	3.2 ±1
Treatment (T)	0.005	
Genotype (G)	<0.001	
T * G	0.2391	

Table T3: *Number of seeds with and without chalk selected across treatments and genotypes. A three-way analysis of variance for this trait was performed under completely randomized design (CRD) using PROC GLM procedure in SAS. Means were separated using HSD (Tukey's Studentized Range) test at p=0.05. Table includes mean and +/- SEM for three way comparison. No significant differences were found.*

Chapter 5

Conclusions and future work

In this dissertation, several state-of-the-art deep learning approaches for plant phenotyping tasks were studies. The tasks addressed include characterization of rice root anatomy based on microscopic root cross-section images, estimation of sorghum stomatal density and area based on microscopic images of leaf surfaces, and estimation of the chalkiness in rice exposed to high night temperature based on images of rice grains.

In Chapter 2, deep learning approaches for identifying root anatomical traits such as root, stele and late metaxylem were studies. This task was implemented using the Faster Region-based Convolutional Neural Network (Faster R-CNN) with the pre-trained VGG-16 as backbone. The model was trained on root cross-section images of roots, where the traits of interest were manually annotated as rectangular bounding boxes using the LabelImg tool. The traits were also predicted as rectangular bounding boxes, which were compared with the ground truth bounding boxes in terms of intersection over union metric to evaluate the detection accuracy. The predicted bounding boxes were subsequently used to estimate root and stele diameter, as well as late metaxylem count and average diameter. Experimental results showed that the trained models can accurately detect and quantify anatomical features, and are robust to image variations. It was also observed that using the pre-trained VGG-16 network enabled the training of accurate models with a relatively small number

of annotated images, making this approach very attractive in terms of adaptations to new tasks.

In Chapter 3, a deep learning approach for estimating sorghum stomatal density and area were studies. A deep learning approach for instance segmentation was used, specifically a Mask Region-based Convolutional Neural Network (Mask R- CNN), which produces pixel-level annotations of stomata objects. The pre-trained ResNet-101 network was used as the backbone of the model in combination with the feature pyramid network (FPN) that enables the model to identify objects at different scales. The Mask R-CNN model was trained on microscopic leaf surface images, where the stomata objects have been manually labeled at pixel level using the VGG Image Annotator tool. The predicted stomata masks were counted, and subsequently used to estimate the stomatal area. Experimental results showed a strong correlation between the predicted counts/stomatal area and the corresponding manually produced values. Furthermore, as for the root anatomy task, this study showed that very accurate results can be obtained with a relatively small number of annotated images.

In Chapter 4, a labour-less approach of using deep learning to segment rice chalkiness area was studies. For the task of estimating chalkiness based on images of rice grains exposed to high night temperatures, a weakly supervised approach was used, specifically, an approach based on Gradient-weighted Class Activation Mapping (Grad-CAM). Instead of performing pixel-level segmentation of the chalkiness in rice images, the weakly supervised approach makes use of high-level annotations of images as chalky or not-chalky. A convolutional neural network (e.g., ResNet-101) for binary classification is trained to distinguish between chalky and not-chalky images, and subsequently the gradients of the chalky class are used to determine a heatmap corresponding to the chalkiness area and also a chalkiness score for a grain. Experimental results on both polished and unpolished rice grains using standard instance classification and segmentation metrics showed that Grad-CAM can accurately identify chalky grains and detect the chalkiness area. The results also showed that the models trained on polished rice cannot be transferred between polished and unpolished rice,

suggesting that new models need to be trained and fine-tuned for other types of rice grains and possibly images taken under different conditions.

For future work, there are several challenges that remain to be addressed. Image acquisition approach should be as simple as possible to ensure usability. However, lower quality images that can be obtained relatively cheap, will require robust models to compensate for the low-cost images. To achieve robust models, the models should be trained with more types and variations of agricultural images and also with other modalities complementary to images. Specifically, multi-modal data that includes not only images but also video (taken by drones or autonomous vechicles), or even text, are also needed to perform time series analyses and predict target yield directly from the phenotype. Many other classification tasks that are important for plant phenotyping are also challenging and need more research efforts, such as 3D density estimation. In the view of state-of-the-art- approaches, we can experiment with faster light weight network and one stage or even anchor free detection and classification approaches, especially if the research will be deployed to the field.

Bibliography

S. Aich and I. Stavness. Leaf counting with deep convolutional and deconvolutional networks. *arXiv preprint arXiv:1708.07570*, 2017.

R. Alfred and C. Lun. Unsupervised learning of image data using generative adversarial network. In X.-S. Y. S. D. Joshi, editor, *Advances in Intelligent Systems and Computing*, volume 1041, pages 127–135. Springer, London, 2019.

J. L. Araus, G. A. Slafer, C. Royo, and M. D. Serret. Breeding for yield potential and stress adaptation in cereals. *Critical Reviews in Plant Sciences*, 27(6):377–412, 2008.

P. R. Armstrong, A. M. McClung, E. B. Maghirang, M. H. Chen, D. L. Brabec, K. F. Yaptenco, A. N. Famoso, and C. K. Addison. Detection of chalk in single kernels of long-grain milled rice using imaging and visible/near-infrared instruments. *Cereal Chemistry*, 96(6):1103–1111, 2019.

K. Ashida, S. Iida, and T. Yasui. Morphological, physical, and chemical properties of grain and flour from chalky rice mutants. *Cereal Chemistry*, 86(2):225–231, 2009. doi: https://doi.org/10.1094/CCHEM-86-2-0225. URL https://onlinelibrary.wiley.com/doi/abs/10.1094/CCHEM-86-2-0225.

R. N. Bahuguna, C. A. Solis, W. Shi, and K. S. V. Jagadish. Post-flowering night respiration and altered sink activity account for high night temperature-induced grain yield and quality loss in rice (oryza sativa l.). *Physiologia Plantarum*, 159(1):59–73, 2017. doi: https://doi.org/10.1111/ppl.12485. URL https://onlinelibrary.wiley.com/doi/abs/10.1111/ppl.12485.

L. T. Bertolino, R. S. Caine, and J. E. Gray. Impact of stomatal density and morphology on water-use efficiency in a changing world. *Frontiers in plant science*, 10:225, 2019.

I. Betegón-Putze, A. González, X. Sevillano, D. Blasco-Escámez, and A. I. Caño-Delgado. Myroot: A novel method and software for the semi-automatic measurement of plant root length. *bioRxiv*, 2018.

R. Bheemanahalli, R. Sathishraj, J. Tack, L. L. Nalley, R. Muthurajan, and K. S. Jagadish. Temperature thresholds for spikelet sterility and associated warming impacts for subtropical rice. *Agricultural and Forest Meteorology*, 221:122–130, 2016.

R. Bheemanahalli, C. Wang, E. Bashir, A. Chiluwal, M. Pokharel, R. Perumal, N. Moghimi, T. Ostmeyer, D. Caragea, and S. K. Jagadish. Classical phenotyping and deep learning concur on genetic control of stomatal density and area in sorghum. *Plant Physiology*, 186 (3):1562–1579, 2021.

A. Bishopp and J. Lynch. The hidden half of crop yields. *Nature Plants*, 1:15117, 08 2015.

C. R. Buckley, R. S. Caine, and J. E. Gray. Pores for thought: can genetic manipulation of stomatal density protect future rice yields? *Frontiers in plant science*, 10:1783, 2020.

A. Bucksch, J. Burridge, L. M. York, A. Das, E. Nord, J. S. Weitz, and J. P. Lynch. Image-based high-throughput field phenotyping of crop roots. *Plant Physiology*, 2014.

A. L. Burton, M. Williams, J. P. Lynch, and K. M. Brown. Rootscan: software for high-throughput analysis of root anatomical traits. *Plant and Soil*, 357(1-2):189–203, 2012.

A. M. Casa, G. Pressoir, P. J. Brown, S. E. Mitchell, W. L. Rooney, M. R. Tuinstra, C. D. Franks, and S. Kresovich. Community resources and strategies for association mapping in sorghum. *Crop science*, 48(1):30–40, 2008.

A. Casado and J. Heras. Guiding the creation of deep learning-based object detectors. *arXiv preprint arXiv:1809.03322*, 2018.

S. Chen, X. jun tao, W. Guo, R. Bu, Z. Zheng, Y. Chen, Z. Yang, and R. Lin. Colored rice quality inspection system using machine vision. *Journal of Cereal Science*, 88, 05 2019. doi: 10.1016/j.jcs.2019.05.010.

X. Chen. *TensorFlow Faster RCNN for Object Detection*, 2017. URL https://github.com/endernewton/tf-faster-rcnn. Accessed 2020-04-13.

X. Chen and A. Gupta. An implementation of faster RCNN with study for region sampling. *CoRR*, abs/1702.02138, 2017.

J. Chopin, H. Laga, C. Y. Huang, S. Heuer, and S. J. Miklavcic. Rootanalyzer: a cross-section image analysis tool for automated characterization of root cells and tissues. *PloS one*, 10(9):e0137655, 2015.

R. T. Clark, R. B. MacCurdy, J. K. Jung, J. E. Shaff, S. R. McCouch, D. J. Aneshansley, and L. V. Kochian. Three-dimensional root phenotyping with a novel imaging and software platform. *Plant Physiology*, 156(2):455–465, 2011.

T. Colombi, N. Kirchgessner, C. A. Le Marié, L. M. York, J. P. Lynch, and A. Hund. Next generation shovelomics: set up a tent and rest. *Plant and Soil*, 388(1-2):1–20, 2015.

L. Comas, S. Becker, V. M. V. Cruz, P. F. Byrne, and D. A. Dierig. Root traits contributing to plant productivity under drought. *Frontiers in plant science*, 4:442, 2013.

T. Dabi and V. Khanna. Effect of climate change on rice. *Agrotechnology*, 7(2):2–7, 2018.

B. M. Delory, M. Li, C. N. Topp, and G. Lobet. archidart v3. 0: A new data analysis pipeline allowing the topological analysis of plant root systems. *F1000Research*, 7, 2018.

H. Dittberner, A. Korte, T. Mettler-Altmann, A. P. Weber, G. Monroe, and J. de Meaux. Natural variation in stomata size contributes to the local adaptation of water-use efficiency in arabidopsis thaliana. *Molecular ecology*, 27(20):4052–4065, 2018.

A. Dobrescu, M. V. Giuffrida, and S. A. Tsaftaris. Leveraging multiple datasets for deep leaf counting. In *ICCV Workshops 2017*, pages 2072–2079, 2017.

K. T. Duarte, M. A. G. de Carvalho, and P. S. Martins. Segmenting high-quality digital images of stomata using the wavelet spot detection and the watershed transform. In *VISIGRAPP (4: VISAPP)*, pages 540–547, 2017.

J. Dunn, L. Hunt, M. Afsharinafar, M. A. Meselmani, A. Mitchell, R. Howells, E. Wallington, A. J. Fleming, and J. E. Gray. Reduced stomatal density in bread wheat leads to increased water-use efficiency. *Journal of Experimental Botany*, 70(18):4737–4748, 2019.

T. L. Durham Brooks, N. D. Miller, and E. P. Spalding. Plasticity of arabidopsis root gravitropism throughout a multidimensional condition space quantified by automated image analysis. *Plant Physiology*, 152(1):206–216, 2010.

A. Dutta and A. Zisserman. The via annotation software for images, audio and video. pages 2276–2279, 10 2019. ISBN 978-1-4503-6889-6. doi: 10.1145/3343031.3350535.

A. Dutta, A. Gupta, and A. Zisserman. *VGG Image Annotator (VIA)*, 2017. URL https://www.robots.ox.ac.uk/~vgg/software/via/via-1.0.6.html. Accessed April 13, 2020.

G. Elmasry, N. Mandour, S. Al-Rejaie, E. Belin, and D. Rousseau. Recent applications of multispectral imaging in seed phenotyping and quality monitoring—an overview. *Sensors*, 19:1090, 03 2019. doi: 10.3390/s19051090.

M. Everingham, S. M. A. Eslami, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The pascal visual object classes challenge: A retrospective. *International Journal of Computer Vision*, 111(1):98–136, Jan. 2015.

G. D. Farquhar and T. D. Sharkey. Stomatal conductance and photosynthesis. *Annual review of plant physiology*, 33(1):317–345, 1982.

U. R. Federation. Exporting u.s. rice, sep 2020. URL https://www.usarice.com/discover-us-rice/find-a-supplier/exporting-u.s.-rice.

K. C. Fetter, S. Eberhardt, R. S. Barclay, S. Wing, and S. R. Keller. Stomatacounter: a neural network for automatic stomata identification and counting. *New Phytologist*, 223 (3):1671–1681, 2019.

M. A. Fitzgerald, S. R. McCouch, and R. D. Hall. Not just a grain of rice: the quest for quality. *Trends in Plant Science*, 14(3):133–139, 2009. ISSN 1360-1385. doi: https://doi.org/10.1016/j.tplants.2008.12.004. URL https://www.sciencedirect.com/science/article/pii/S1360138509000430.

R. T. Furbank and M. Tester. Phenomics–technologies to relieve the phenotyping bottleneck. *Trends in plant science*, 16(12):635–644, 2011.

R. Girshick. Fast r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 1440–1448, 2015.

D. C. Gitz and J. T. Baker. Methods for creating stomatal impressions directly onto archivable slides. *Agronomy Journal*, 101(1):232–236, 2009.

I. Goodfellow, Y. Bengio, and A. Courville. *Deep Learning*. MIT Press, 2016. http://www.deeplearningbook.org.

S. M. Gourdji, A. M. Sibley, and D. B. Lobell. Global crop exposure to critical high temperatures in the reproductive period: historical trends and future projections. *Environmental Research Letters*, 8(2):024041, jun 2013. doi: 10.1088/1748-9326/8/2/024041. URL https://doi.org/10.1088/1748-9326/8/2/024041.

V. R. Gowda, A. Henry, A. Yamauchi, H. Shashidhar, and R. Serraj. Root biology and genetic improvement for drought avoidance in rice. *Field Crops Research*, 122(1):1–13, 2011.

H. C. Hall, A. Fakhrzadeh, C. L. Luengo Hendriks, and U. Fischer. Precision automation of cell type classification and sub-cellular fluorescence quantification from laser scanning confocal images. *Frontiers in plant science*, 7:119, 2016.

J. R. Harlan and J. M. Wet. A simplified classification of cultivated sorghum 1. *Crop science*, 12(2):172–176, 1972.

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

K. He, G. Gkioxari, P. Dollár, and R. Girshick. Mask r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 2961–2969, 2017.

K. He, G. Gkioxari, P. Dollar, and R. Girshick. Mask r-cnn. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PP:1–1, 06 2018. doi: 10.1109/TPAMI.2018.2844175.

A. Henry, A. J. Cal, T. C. Batoto, R. O. Torres, and R. Serraj. Root attributes affecting water uptake of rice (oryza sativa) under drought. *Journal of experimental botany*, 63(13): 4751–4763, 2012.

T. Higaki, N. Kutsuna, and S. Hasezawa. Carta-based semi-automatic detection of stomatal regions on an arabidopsis cotyledon surface. *Plant Morphology*, 26(1):9–12, 2014.

A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, and H. Adam. Mobilenets: Efficient convolutional neural networks for mobile vision applications. *arXiv preprint arXiv:1704.04861*, 2017.

G. Huang, Z. Liu, L. van der Maaten, and K. Weinberger. Densely connected convolutional networks. 07 2017. doi: 10.1109/CVPR.2017.243.

M. E. Hudson. Sequencing breakthroughs for genomic ecology and evolutionary biology. *Molecular ecology resources*, 8(1):3–17, 2008.

J. Hughes, C. Hepworth, C. Dutton, J. A. Dunn, L. Hunt, J. Stephens, R. Waugh, D. D. Cameron, and J. E. Gray. Reducing stomatal density in barley improves drought tolerance without impacting on yield. *Plant physiology*, 174(2):776–787, 2017.

F. Iandola, S. Han, M. Moskewicz, K. Ashraf, W. Dally, and K. Keutzer. Squeezenet: Alexnet-level accuracy with 50x fewer parameters and ¡0.5mb model size. 02 2016.

S. M. Impa, B. Raju, N. T. Hein, J. Sandhu, P. V. Prasad, H. Walia, and S. K. Jagadish. High night temperature effects on wheat and rice: Current status and way forward. *Plant, Cell & Environment*, 2021.

S. Jagadish, P. Craufurd, and T. Wheeler. High temperature stress and spikelet fertility in rice (Oryza sativa L.). *Journal of Experimental Botany*, 58(7):1627–1635, 04 2007. ISSN 0022-0957. doi: 10.1093/jxb/erm003. URL https://doi.org/10.1093/jxb/erm003.

S. Jagadish, P. Craufurd, and T. Wheeler. Phenotyping parents of mapping populations of rice for heat tolerance during anthesis. *Crop Science*, 48(3):1140–1146, 2008.

S. Jagadish, J. Cairns, R. Lafitte, T. R. Wheeler, A. Price, and P. Q. Craufurd. Genetic analysis of heat tolerance at anthesis in rice. *Crop Science*, 50(5):1633–1641, 2010.

S. V. K. Jagadish, M. V. R. MURTY, and W. P. QUICK. Rice responses to rising temperatures – challenges, perspectives and future directions. *Plant, Cell & Environment*, 38(9):1686–1698, 2015. doi: https://doi.org/10.1111/pce.12430. URL https://onlinelibrary.wiley.com/doi/abs/10.1111/pce.12430.

H. Jayakody, S. Liu, M. Whitty, and P. Petrie. Microscope image based fully automated stomata detection and pore measurement method for grapevines. *Plant methods*, 13(1): 1–12, 2017.

Y. Jia. *Caffe*. URL http://caffe.berkeleyvision.org/. Accessed 2020-04-13.

W. Jones, K. Alasoo, D. Fishman, and L. Parts. Computational biology: deep learning. *Emerging Topics in Life Sciences*, 1:257–274, 11 2017. doi: 10.1042/ETLS20160025.

J. Jung and S. Mccouch. Getting to the roots of it: Genetic and hormonal control of root architecture. *Frontiers in plant science*, 4:186, 06 2013.

N. Kadam, X. Yin, P. Bindraban, P. Struik, and K. Jagadish. Does morphological and anatomical plasticity during the vegetative stage make wheat more tolerant of water-deficit stress than rice? *Plant physiology*, pages pp–114, 2015.

N. Kadam, A. Tamilselvan, L. M. F. Lawas, C. Quinones, R. Bahuguna, M. J. Thomson, M. Dingkuhn, R. Muthurajan, P. Struik, X. Yin, and K. Jagadish. Genetic control of plasticity in root morphology and anatomy of rice in response to water-deficit. *Plant physiology*, pages pp–00500, 2017.

A. Kamilaris and F. X. Prenafeta-Boldú. Deep learning in agriculture: A survey. *Computers and Electronics in Agriculture*, 147:70–90, 2018.

A. Kamilaris and F. X. Prenafeta-Boldú. Deep learning in agriculture: A survey. *Computers and Electronics in Agriculture*, 147:70–90, 2018. ISSN 0168-1699. doi: https://doi.org/10.1016/j.compag.2018.02.016. URL https://www.sciencedirect.com/science/article/pii/S0168169917308803.

Z. Khan, V. Rahimi-Eichi, S. Haefele, T. Garnett, and S. J. Miklavcic. Estimation of vegetation indices for high-throughput phenotyping of wheat using aerial imaging. *Plant methods*, 14(1):20, 2018.

G. S. Khush. Strategies for increasing the yield potential of cereals: case of rice as an example. *Plant Breeding*, 132(5):433–436, 08 2013.

E. Komyshev, G. M.A., and D. Afonnikov. Evaluation of the seedcounter, a mobile applica-

tion for grain phenotyping. *Frontiers in Plant Science*, 7, 01 2017. doi: 10.3389/fpls.2016.01990.

A. Krizhevsky, I. Sutskever, and G. Hinton. Imagenet classification with deep convolutional neural networks. *Neural Information Processing Systems*, 25, 01 2012. doi: 10.1145/3065386.

H. Laga, F. Shahinnia, and D. Fleury. Image-based plant stornata phenotyping. In *2014 13th International Conference on Control Automation Robotics & Vision (ICARCV)*, pages 217–222. IEEE, 2014.

S. B. Lanning, T. J. Siebenmorgen, P. A. Counce, A. A. Ambardekar, and A. Mauromoustakos. Extreme nighttime air temperatures in 2010 impact rice chalkiness and milling quality. *Field Crops Research*, 124(1):132–136, 2011.

M. Lartaud, C. Perin, B. Courtois, E. Thomas, S. Henry, M. Bettembourg, F. Divol, N. Lanau, F. Artus, C. Bureau, et al. Phiv-rootcell: a supervised image analysis tool for rice root anatomical parameter quantification. *Frontiers in plant science*, 5:790, 2015.

Y. LeCun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard, and L. D. Jackel. Backpropagation applied to handwritten zip code recognition. *Neural computation*, 1(4):541–551, 1989.

Y. LeCun, Y. Bengio, and G. Hinton. Deep learning. *Nature*, 521(7553):436, 2015.

B. Leff, N. Ramankutty, and J. A. Foley. Geographic distribution of major crops across the world. *Global biogeochemical cycles*, 18(1), 2004.

X. Li, D. Caragea, H. Zhang, and M. Imran. Localizing and quantifying infrastructure damage using class activation mapping approaches. *Social Network Analysis and Mining*, 9(1):44, 2019.

G. H. Liang, A. Dayton, C. Chu, and A. Casady. Heritability of stomatal density and distribution on leaves of grain sorghum 1. *Crop Science*, 15(4):567–570, 1975.

T.-Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie. Feature pyramid networks for object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2117–2125, 2017.

A. J. Lisle, M. Martin, and M. A. Fitzgerald. Chalky and translucent rice grains differ in starch composition and structure and cooking properties. *Cereal Chemistry*, 77 (5):627–632, 2000. doi: https://doi.org/10.1094/CCHEM.2000.77.5.627. URL https://onlinelibrary.wiley.com/doi/abs/10.1094/CCHEM.2000.77.5.627.

N. B. Lyman, K. S. V. Jagadish, L. L. Nalley, B. L. Dixon, and T. Siebenmorgen. Neglecting rice milling yield and quality underestimates economic losses from high-temperature stress. *PLOS ONE*, 8(8):1–9, 08 2013. doi: 10.1371/journal.pone.0072157. URL https://doi.org/10.1371/journal.pone.0072157.

J. P. Lynch, J. G. Chimungu, and K. M. Brown. Root anatomical phenes associated with water acquisition from drying soil: targets for crop improvement. *Journal of Experimental Botany*, 65(21):6155–6166, 2014.

Matterport-Inc. *Mask R-CNN for Object Detection and Segmentation*, 2017. URL https://github.com/matterport/Mask_RCNN. Accessed April 13, 2020.

S. P. Mohanty, D. P. Hughes, and M. Salathé. Using deep learning for image-based plant disease detection. *Frontiers in plant science*, 7:1419, 2016.

S. J. Mooney, T. P. Pridmore, J. Helliwell, and M. J. Bennett. Developing x-ray computed tomography to non-invasively image 3-d root systems architecture in soil. *Plant and Soil*, 352(1):1–22, Mar 2012.

G. P. Morris, P. Ramu, S. P. Deshpande, C. T. Hash, T. Shah, H. D. Upadhyaya, O. Riera-Lizarazu, P. J. Brown, C. B. Acharya, S. E. Mitchell, et al. Population genomic and genome wide association studies of agroclimatic traits in sorghum. *Proceedings of the National Academy of Sciences*, 110(2):453–458, 2013.

R. Muchow and T. Sinclair. Epidermal conductance, stomatal density and stomatal size among genotypes of sorghum bicolor (l.) moench. *Plant, Cell & Environment*, 12(4): 425–431, 1989.

N.A. Food supply - crops primary equivalent database, a. URL http://www.fao.org/faostat/en/#data/QC.

N.A. The future of food and agriculture. trends and challenges., b. URL http://www.fao.org/3/a-i6583e.pdf.

N.A. World population prospects 2019: Data booklet, c. URL https://population.un.org/wpp/Publications/Files/WPP2019_DataBooklet.pdf.

N.A. k-means advantages and disadvantages — clustering in machine learning, d. URL https://developers.google.com/machine-learning/clustering/algorithm/advantages-disadvantages.

S. T. Namin, M. Esmaeilzadeh, M. Najafi, T. B. Brown, and J. O. Borevitz. Deep phenotyping: Deep learning for temporal phenotype/genotype classification. *bioRxiv*, page 134205, 2017.

U. N. D. of Public Information. Food production must double by 2050 to meet demand from world's growing population, innovative strategies needed to combat hunger, experts tell second committee, oct 2009. URL https://www.un.org/press/en/2009/gaef3242.doc.htm.

A. Ohsumi, T. Kanemura, K. Homma, T. Horie, and T. Shiraiwa. Genotypic variation of stomatal conductance in relation to stomatal density and length in rice (oryza sativa l.). *Plant Production Science*, 10(3):322–328, 2007.

C. P. Osborne and R. P. Freckleton. Ecological selection pressures for c4 photosynthesis in the grasses. *Proceedings of the Royal Society B: Biological Sciences*, 276(1663):1753–1760, 2009.

N. Otsu. A threshold selection method from gray-level histograms. *IEEE transactions on systems, man, and cybernetics*, 9(1):62–66, 1979.

S. Payman, A. Bakhshipour, and H. Zareiforoush. Development of an expert vision-based system for inspecting rice quality indices. *Quality Assurance and Safety of Crops & Foods*, 10(1):103–114, 2018.

J. Pfeifer, N. Kirchgessner, T. Colombi, and A. Walter. Rapid phenotyping of crop root systems in undisturbed field soils using x-ray computed tomography. *Plant Methods*, 11:14, 08 2015.

M. P. Pound, J. A. Atkinson, A. J. Townsend, M. H. Wilson, M. Griffiths, A. S. Jackson, A. Bulat, G. Tzimiropoulos, D. M. Wells, E. H. Murchie, T. P. Pridmore, and A. P. French. Deep machine learning provides state-of-the-art performance in image-based plant phenotyping. *GigaScience*, 6(10):1–10, 2017a.

M. P. Pound, S. Fozard, M. T. Torres, B. G. Forde, and A. P. French. Autoroot: open-source software employing a novel image analysis approach to support fully-automated plant phenotyping. *Plant methods*, 13(1):12, 2017b.

D. K. Ray, N. D. Mueller, P. C. West, and J. A. Foley. Yield trends are insufficient to double global crop production by 2050. *PLOS ONE*, 8(6):1–8, 06 2013. doi: 10.1371/journal.pone.0066428. URL https://doi.org/10.1371/journal.pone.0066428.

C. Reeb, J. Kaandorp, F. Jansson, N. Puillandre, J.-Y. Dubuisson, R. Cornette, F. Jabbour, Y. Coudert, J. Patiño, J.-F. Flot, et al. Quantification of complex modular architecture in plants. *New Phytologist*, 218(2):859–872, 2018.

S. Ren. *faster_rcnn*, 2015. URL https://github.com/ShaoqingRen/faster_rcnn. Accessed 2020-04-13.

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

R. Richards and J. Passioura. A breeding program to reduce the diameter of the major xylem vessel in the seminal roots of wheat and its effect on grain yield in rain-fed environments. *Australian Journal of Agricultural Research*, 40(5):943–950, 1989.

M. Rieger and P. Litvin. Root system hydraulic conductivity in species with contrasting root anatomy. *Journal of experimental botany*, 50(331):201–209, 1999.

A. J. Rowland-Bamford, C. Nordenbrock, J. T. Baker, G. Bowes, and L. H. Allen Jr. Changes in stomatal density in rice grown under various co2 regimes with natural solar irradiance. *Environmental and Experimental Botany*, 30(2):175–180, 1990.

R. Rs, M. Cogswell, R. Vedantam, D. Parikh, and D. Batra. Grad-cam: Visual explanations from deep networks via gradient-based localization. *International Journal of Computer Vision*, 10 2019. doi: 10.1007/s11263-019-01228-7.

O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, et al. Imagenet large scale visual recognition challenge. *International journal of computer vision*, 115(3):211–252, 2015.

W. Sadok and S. K. Jagadish. The hidden costs of nighttime warming on yields. *Trends in Plant Science*, 25(7):644–651, 2020.

K. Sakoda, T. Watanabe, S. Sukemura, S. Kobayashi, Y. Nagasaki, Y. Tanaka, and T. Shiraiwa. Genetic diversity in stomatal density among soybeans elucidated using high-throughput technique based on an algorithm for object detection. *Scientific reports*, 9 (1):1–9, 2019.

M. Sankar, K. Nieminen, L. Ragni, I. Xenarios, and C. S. Hardtke. Automated quantitative histology reveals vascular morphodynamics during arabidopsis hypocotyl secondary growth. *Elife*, 3, 2014.

J. E. Schmidt and A. C. Gaudin. Toward an integrated root ideotype for irrigated systems. *Trends in Plant Science*, 22:433–443, 03 2017.

C. A. Schneider, W. S. Rasband, and K. W. Eliceiri. Nih image to imagej: 25 years of image analysis. *Nature methods*, 9(7):671, 2012.

H. Schulz, J. A. Postma, D. van Dusschoten, H. Scharr, and S. Behnke. Plant root system analysis from mri images. In G. Csurka, M. Kraus, R. S. Laramee, P. Richard, and J. Braz, editors, *Computer Vision, Imaging and Computer Graphics. Theory and Application*, pages 411–425, Berlin, Heidelberg, 2013. Springer Berlin Heidelberg. ISBN 978-3-642-38241-3.

M. Schumacher, A. Genz, and M. Heinrich. Weakly supervised pancreas segmentation based on class activation maps. In *Medical Imaging 2020: Image Processing*, volume 11313, page 1131314. International Society for Optics and Photonics, 2020.

D. Šebela, R. Bheemanahalli, A. Tamilselvan, N. N. Kadam, and S. K. Jagadish. Genetic dissection of photochemical efficiency under water-deficit stress in rice. *Plant Physiology Reports*, 24(3):328–339, 2019.

R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra. Grad-cam: Visual explanations from deep networks via gradient-based localization. *International Journal of Computer Vision*, 128, 10 2019. doi: 10.1007/s11263-019-01228-7.

P. Sethy, N. Barpanda, and A. Rath. Quantification of rice chalkiness using image processing. pages 2278–4853, 08 2018.

E. Shelhamer, J. Long, and T. Darrell. Fully convolutional networks for semantic segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39:1–1, 05 2016. doi: 10.1109/TPAMI.2016.2572683.

W. Shi, R. Muthurajan, H. Rahman, J. Selvam, S. Peng, Y. Zou, and K. S. Jagadish. Source–sink dynamics and proteomic reprogramming under elevated night temperature and their impact on rice yield and grain quality. *New Phytologist*, 197(3):825–837, 2013.

W. Shi, X. Yin, P. C. Struik, C. Solis, F. Xie, R. C. Schmidt, M. Huang, Y. Zou, C. Ye, and S. V. K. Jagadish. High day- and night-time temperatures affect grain growth dynamics in contrasting rice genotypes. *Journal of Experimental Botany*, 68(18):5233–5245, 10 2017. ISSN 0022-0957. doi: 10.1093/jxb/erx344. URL https://doi.org/10.1093/jxb/erx344.

K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014a.

K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv 1409.1556*, 09 2014b.

A. Singh, B. Ganapathysubramanian, A. K. Singh, and S. Sarkar. Machine learning for high-throughput stress phenotyping in plants. *Trends in plant science*, 21(2):110–124, 2016.

M. A. Souibgui and Y. Kessentini. De-gan: A conditional generative adversarial network for document enhancement. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2020.

R. Sozzani, W. Busch, E. P. Spalding, and P. N. Benfey. Advanced imaging techniques for

the study of plant growth and development. *Trends in Plant Science*, 19(5):304 – 310, 2014.

M. F. Stuecker, M. Tigchelaar, and M. B. Kantar. Climate variability impacts on rice production in the philippines. *PLoS One*, 13(8):e0201426, 2018.

Y. Su and L. Xiao. 3d visualization and volume based quantification of rice chalkiness in vivo by using high resolution micro-ct. 01 2020. doi: 10.21203/rs.2.21396/v1.

C. Sun, T. Liu, C. Ji, M. Jiang, T. Tian, D. Guo, L. Wang, Y. Chen, and X. Liang. Evaluation and analysis the chalkiness of connected rice kernels based on image processing technology and support vector machine. *Journal of Cereal Science*, 60(2):426–432, 2014.

J. Tack, J. Lingenfelser, and S. K. Jagadish. Disaggregating sorghum yield reductions under warming scenarios exposes narrow genetic diversity in us breeding programs. *Proceedings of the National Academy of Sciences*, 114(35):9296–9301, 2017.

M. Tan and Q. Le. Efficientnet: Rethinking model scaling for convolutional neural networks. 05 2019.

F. Tardieu, L. Cabrera-Bosquet, T. Pridmore, and M. Bennett. Plant phenomics, from sensors to knowledge. *Current Biology*, 27:R770–R783, 08 2017. doi: 10.1016/j.cub.2017.05.055.

T. Tashiro and I. Wardlaw. The effect of high temperature on kernel dimensions and the type and occurrence of kernel damage in rice. *Australian Journal of Agricultural Research*, 42(3):485–496, 1991.

TensorFlow. *An end-to-end open source machine learning platform*. URL https://www.tensorflow.org/. Accessed 2020-04-13.

C. N. Topp, A. S. Iyer-Pascuzzi, J. T. Anderson, C.-R. Lee, P. R. Zurek, O. Symonova, Y. Zheng, A. Bucksch, Y. Mileyko, T. Galkovskyi, B. T. Moore, J. Harer, H. Edelsbrunner,

T. Mitchell-Olds, J. S. Weitz, and P. N. Benfey. 3d phenotyping and quantitative trait locus mapping identify core regions of the rice genome controlling root architecture. *Proceedings of the National Academy of Sciences*, 110(18):E1695–E1704, 2013.

Tzutalin. Labelimg, 2015. URL https://github.com/tzutalin/labelImg.

J. R. Ubbens and I. Stavness. Deep plant phenomics: a deep learning platform for complex plant phenotyping tasks. *Frontiers in plant science*, 8:1190, 2017.

Y. Uga, K. Sugimoto, S. Ogawa, J. Rane, M. Ishitani, N. Hara, Y. Kitomi, Y. Inukai, K. Ono, N. Kanno, H. Inoue, H. Takehisa, R. Motoyama, Y. Nagamura, J. wu, T. Matsumoto, T. Takai, K. Okuno, and M. Yano. Control of root system architecture by deeper rooting 1 increases rice yield under drought conditions. *Nature genetics*, 45, 08 2013.

D. van Dusschoten, R. Metzner, J. Kochs, J. A. Postma, D. Pflugfelder, J. Buehler, U. Schurr, and S. Jahnke. Quantitative 3d analysis of plant roots growing in soil using magnetic resonance imaging. *Plant Physiology*, 2016.

K. Vinogradova, A. Dibrov, and G. Myers. Towards interpretable semantic segmentation via gradient-weighted class activation mapping. *arXiv preprint arXiv:2002.11434*, 2020.

A. Walter, F. Liebisch, and A. Hund. Plant phenotyping: From bean weighing to image analysis. *Plant Methods*, 11:14, 03 2015. doi: 10.1186/s13007-015-0056-8.

C. Wang. *Root Anatomy github*, 2019a. URL https://github.com/cwang16/Root-Anatomy-Using-Faster-RCNN. Accessed 2020-04-13.

C. Wang. *Root Anatomy Demo*, 2019b. URL https://rootanatomy.cs.ksu.edu/. Accessed 2020-04-13.

C. Wang, X. Li, D. Caragea, R. Bheemanahalli, and S. K. Jagadish. Root anatomy based on root cross-section image analysis with deep learning. *Cold Spring Harbor Laboratory,*, 10.1101/442244, 2019.

C. Wang, D. Caragea, N. Kodadinne Narayana, N. T. Hein, R. Bheemanahalli, I. M. Somayanda, and S. Jagadish. Deep learning based high-throughput phenotyping of chalkiness in rice exposed to high night temperature. *Plant methods*, 18(1):1–23, 2022.

K. Wang, Y. Li, Y. Wang, and X. Yang. On the asymmetry of the urban daily air temperature cycle. *Journal of Geophysical Research: Atmospheres*, 122(11):5625–5635, 2017a.

S. Wang, W. Chen, S. M. Xie, G. Azzari, and D. B. Lobell. Weakly supervised deep learning for segmentation of remote sensing imagery. *Remote Sensing*, 12(2):207, 2020.

Y. Wang, F. Zhu, C. J. Boushey, and E. J. Delp. Weakly supervised food image segmentation using class activation maps. In *2017 IEEE International Conference on Image Processing (ICIP)*, pages 1277–1281. IEEE, 2017b.

A. P. Wasson, R. Richards, R. Chatrath, S. Misra, S. S. Prasad, G. Rebetzke, J. Kirkegaard, J. Christopher, and M. Watt. Traits and selection strategies to improve root systems and water uptake in water-limited wheat crops. *Journal of experimental botany*, 63(9): 3485–3498, 2012.

J.-s. Xiao, H.-h. Xu, and X.-j. Ma. Weakly supervised semantic segmentation based on superpixel sampling clustering networks. In *Proceedings of the 2nd International Conference on Computer Science and Software Engineering*, page 178–183. Association for Computing Machinery, 2019.

T. Yamauchi, K. Watanabe, A. Fukazawa, H. Mori, F. Abe, K. Kawaguchi, A. Oyanagi, and M. Nakazono. Ethylene and reactive oxygen species are involved in root aerenchyma formation and adaptation of wheat seedlings to oxygen-deficient conditions. *Journal of experimental botany*, 65(1):261–273, 2013.

S. Yang, Y. Kim, Y. Kim, and C. Kim. Combinational class activation maps for weakly supervised object localization. In *The IEEE Winter Conference on Applications of Computer Vision*, pages 2941–2949, 2020.

W. Yang, L. Duan, G. Chen, L. Xiong, and Q. Liu. Plant phenomics and high-throughput phenotyping: Accelerating rice functional genomics using multidisciplinary technologies. *Current opinion in plant biology*, 16, 04 2013. doi: 10.1016/j.pbi.2013.03.005.

Q. Yao, J. Chen, Z. Guan, C. Sun, and Z. Zhu. Inspection of rice appearance quality using machine vision. *2010 Second WRI Global Congress on Intelligent Systems*, 4:274–279, 01 2009. doi: 10.1109/GCIS.2009.91.

L. Zhang, L. Lin, X. Liang, and K. He. Is faster r-cnn doing well for pedestrian detection? In *European Conference on Computer Vision*, pages 443–457. Springer, 2016.

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

B. Zoph, V. Vasudevan, J. Shlens, and Q. V. Le. Learning transferable architectures for scalable image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 8697–8710, 2018.