METHODOLOGY

Root Anatomy based on Root Cross-Section Image Analysis with Deep Learning

Xukun Li[1], Chaoxin Wang[1], Doina Caragea[1*], Raju Bheemanahalli[2] and S.V. Krishna Jagadish[2]

[*]Correspondence: dcaragea@ksu.edu
[1]Department of Computer Science, Kansas State University, Manhattan, KS, 66502, USA
Full list of author information is available at the end of the article

Abstract

Background: The 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 the potential to further increase food production. However, the 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. Automated machine learning tools have the potential to accomplish this complex task.

Methods: We propose a fully automated approach, called Deep Learning for Root Anatomy (DL-RootAnatomy), for identifying anatomical traits in root cross-section images. Using the Faster Region-based Convolutional Neural Network (Faster R-CNN), the DL-RootAnatomy models detect objects such as root, stele and late metaxylem, and predict rectangular bounding boxes around such objects. Subsequently, the bounding boxes are used to estimate the root diameter, stele diameter, late metaxylem average diameter and number.

Conclusions: Experimental evaluation using standard object detection metrics, such as intersection-over-union and mean average precision, has shown that our models can accurately detect the root, stele and late metaxylem objects. Furthermore, the results have shown that the measurements estimated based on predicted bounding boxes have very small root mean square error when compared with the corresponding ground truth values, suggesting that DL-RootAnatomy can be used to accurately detect anatomical features. Finally, a comparison with existing approaches, which involve some degree of human interaction, has shown that the proposed approach is more accurate than existing approaches.

Keywords: Image Analysis; Deep Learning; Object Detection; Root Anatomy

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 [1, 2, 3]. 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 [4, 5]. Hence, root potential has largely been untapped in crop improvement programs [4, 5]. Over the past decade, different root phenotyping approaches have been developed for studying root architecture, e.g., basket method for root angle [6]; rhizotron method for tracking root branching, architecture and growth dynamics [7]. 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 [8, 9, 10, 11, 12]. 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 approach that could become a blanket fit [13, 14, 15]. The recent high throughput approaches, 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 approaches have been used successfully to address a wide variety of biological problems, including problems relevant to crop sciences, such as genome annotation [16], predicting gene functions [17], discovery of genetic variation and genotyping [18], identification of genomic regions of interest [10, 19], high throughput phenotyping based on aerial image analysis [20]. Furthermore, applications of advanced deep learning and image analysis 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 [21, 22].

Most relevant to this work, machine learning, and more specifically, deep learning, are expanding the ability to accurately predict a plant phenotype [23, 24, 25, 26, 27, 28, 29, 30], enabling the 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 [24].

In particular, several prior studies have focused on data-driven approaches and tools for identifying and quantifying root morphological parameters, for example, changes in root architecture, or branching and growth [31, 32, 33, 34, 35, 36]. While the study of root morphological parameters is important in relation to the health and productivity of crops, the study of root anatomical parameters, such as stele and the xylem vessels, is equally important. Root anatomical parameters represent the conduits for transport of water and nutrients to the plant's aboveground parts. Hence, they are significantly affected by different rhizosphere conditions, and in turn, affect crop productivity [5].

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 [37, 38, 39]. 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 leafs and roots) in the context of secondary growth [40, 41], 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 handle the amount of data produced by image acquisition techniques at a low cost.

To address this limitation, we have taken advantage of recent advances in deep learning and image analysis, and developed a state-of-the-art, fully-automated deep learning approach for identifying and quantifying root anatomical parameters, indicative of the physiological and genetic responses of root anatomical plasticity in field crops. Specifically, we have considered the following parameters: root diameter (RD), stele diameter (SD), late metaxylem diameter (LMXD) and late metaxylem number (LMXN), which were studied in relation to water-deficit stress by Kadam et al. [42, 43]. A graphical illustration of these parameters is shown in Figure 1.

Our proposed approach is based on the Faster R-CNN network [44], and can be used to produce models that can 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 can be used to estimate anatomical parameters such as RD, SD, LMXD, LMXN. Once trained, our models generalize well to unseen images, while eliminating the need for the end-user to hand-draw a stele border or manually choose the metaxylem cells, tasks that are time-consuming, and also prone to noise and errors.

To summarize, our contributions are as follows:

- We have proposed an approach based on Faster R-CNN to detect root, stele and late metaxylem objects, and their corresponding bounding boxes, in root cross-section images.
- We have investigated the Faster R-CNN models with respect to the number of instances needed to accurately detect root, stele and late metaxylem objects, and their corresponding bounding boxes.
- We have evaluated the ability of the predicted bounding boxes to produce accurate estimates for the RD, SD, LMXD, and LMXN anatomical properties, and performed error analysis to identify sources of errors.
- We have compared the results of the proposed fully-automated Faster R-CNN approach with the results obtained from existing automated (i.e., RootAnalyzer) and semi-automated (i.e., RootScan) approaches in terms of accuracy.

2 Background

Plant phenomics, defined as “characterizing the plasticity of the plant response when exposed to a range of environmental conditions” [23], has become possible due to advances in high-throughput imaging technologies, coupled with advances in deep learning and image analysis [24, 25, 26]. Most of the recent studies on plant phenotyping are focused on identifying and quantifying aboveground plant traits using deep learning approaches. For example, several studies have used use deep learning regression approaches to count leaves in rosette plants based on high-resolution RBF images [27, 28, 29], motivated by a recent leaf segmentation challenge [45]. Other investigations have focused on identifying diseases [46] or phenotyping for stress/nutrient deficiencies [24].

While the belowground plant phenotype is overall less studied than the aboveground phenotype, several studies have focused on approaches and tools for quantifying morphological traits such as root length [35] and root system architecture [33]. Such approaches generally rely on standard image analysis techniques as opposed to state-of-the-art machine learning approaches.

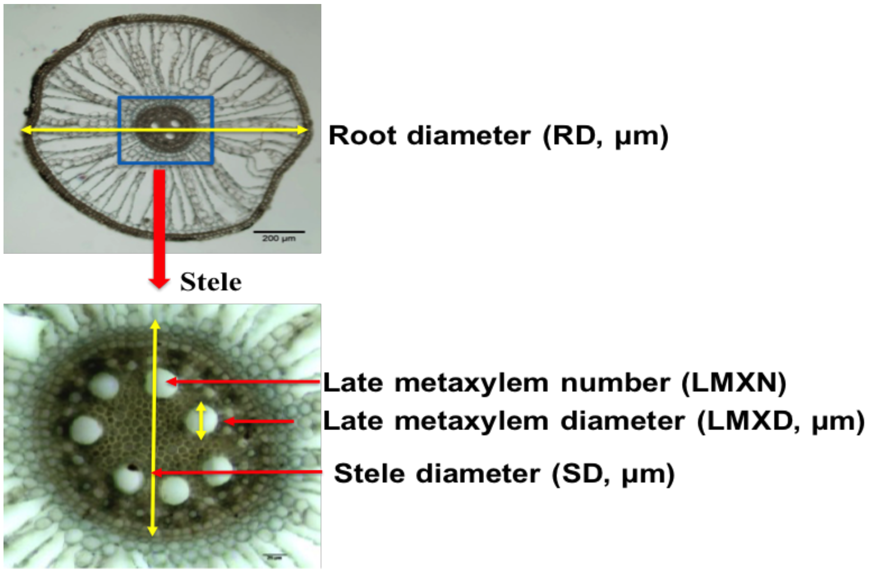
Root diameter (RD, µm)
200 µm
Stele
Late metaxylem number (LMXN)
Late metaxylem diameter (LMXD, µm)
Stele diameter (SD, µm)

Figure 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. Image taken at 100x magnification.

Of most interest to us are approaches and/or tools that can be used to identify anatomical root traits (such as RD, SD, LMXD and LMXN) in root cross-section images. 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 [47] is an image analysis tool that has been extensively used to manually identify and quantify root anatomical traits [42, 48, 43], given that it enables researchers to mark objects of interest and obtain their measurements. In particular, the ImageJ software was used to acquire ground truth annotations for our analysis.

Semi-automated tools require user feedback to tune parameters for individual images in order to get accurate results. *RootScan* [37] and *PHIV-RootCell* [39] are semi-automated image analysis tools that identify and quantify anatomical root traits. RootScan involves several steps for the analysis of each image. It starts by isolating the cross-section from the background using a global thresholding technique [49]. 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. After each step, the user has to "approve" the segmentation/classification performed automatically 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 output of RootScan consists of a table with area measurements and counts of different anatomical traits.

The *PHIV-RootCell* tool for root anatomy is built using the ImageJ software [47], 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 [39]. For example, it requires the user to validate the root selection, stele selection, central metaxylem selection, among others. Given the amount of user involvement required by this tool, PHIV-RootCell was not used in our comparative analysis.

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* 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 allows 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 also tested on maize [38].

As opposed to the tools for root anatomy described above, which use background knowledge about the root anatomy of the species of interest, current automated machine learning tools for hypocotyl cross-section analysis, such as [40, 41], use user input in the form of labeled data to train a model, and can potentially be generalized to other species [50]. Both [40] and [41] use image processing and analysis tools to extract features, which are subsequently used to train traditional support vector machines and random forest classifiers, respectively.

Similar to Hall et al. [40] and Sankar et al. [41], we have developed an automated approach, DL-RootAnatomy, that uses training data to construct models for detecting objects in root cross-section images. However, instead of extracting features and building standard machine learning classifiers, we have used deep learning, specifically Faster R-CNN networks, to concurrently extract features and detect objects of interest, as described in the next section.

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), 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), as motivated by [42, 43]. These tasks can be achieved with modern object detection techniques.

3.1 Overview of the Proposed Approach

We have used Faster R-CNN [44], a state-of-the-art 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 similar 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 model, 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 proposed Faster R-CNN model architecture used to detect the root and stele in a root cross-section image is shown in Figure 2. Faster R-CNN has two main components. The first component consists of a Region Proposal Network (RPN), which identifies Regions of Interest (which potentially contain objects of interest), and also their location. The second component consists of a Fast R-CNN [51], which classifies the proposed 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 VGG-16 [52], 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 used to detect objects and generate bounding boxes are provided below.

3.2 Convolutional Neural Networks and VGG-16

Convolutional Neural Networks (CNNs) [53] 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 the example in Figure 3 (which shows a specific CNN architecture called VGG-16 [52]). A convolutional layer uses 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

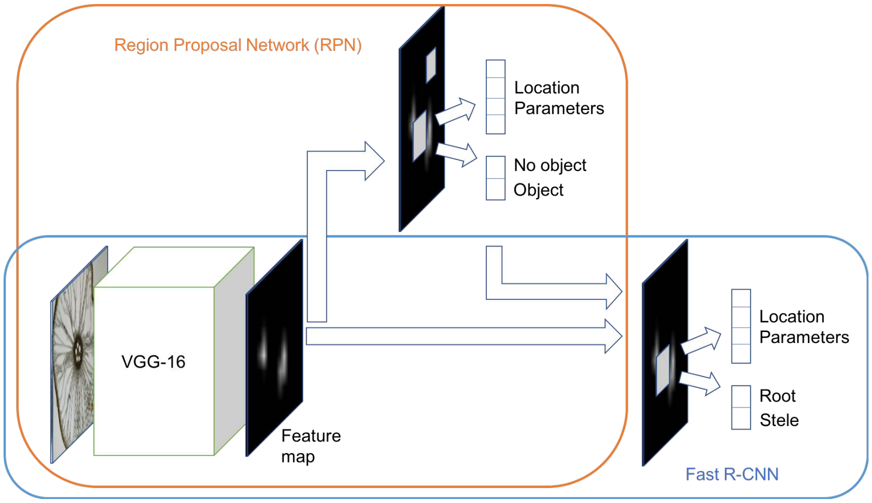

Figure 2 Proposed Faster R-CNN model architecture, which has two main components: 1) a region proposal network (RPN), which identifies regions that may contain objects of interest and their approximate location; and 2) a Fast R-CNN network, which classifies objects as root or stele, and refines their location, defined using bounding boxes. The two components share the convolutional layers of the pre-trained VGG-16 [52].

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 equivariant to scale and translation [54].

After a sequence of convolutional layers (together with non-linear activations) and pooling layers, a CNN has one or more fully connected layers. In fully connected layers all neurons in the current layer are connected to all neurons in the next 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 depenencies between features [54]. The last fully connected layer uses a softmax activation function, and has as many output neurons as the number of classes in the classification task.

There are several pre-trained CNN architectures available, including VGG-16 [52], shown in Figure 3. VGG-16 has been shown to give very good performance in

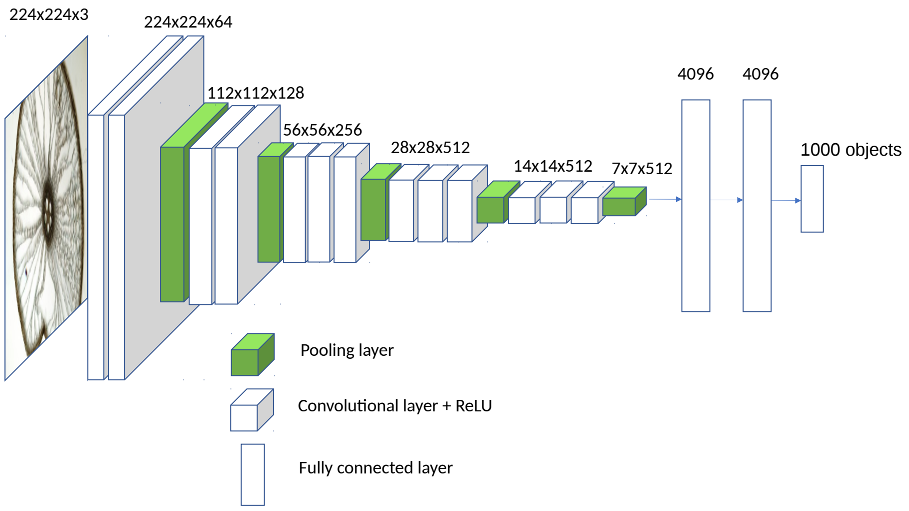

Figure 3 VGG-16. The original VGG-16 architecture consists of 13 convolution+ReLU layers, five pooling layers, and three fully connected layers. A convolution+ReLU layer produces a feature map, while a pooling layer reduces the dimensionality of the feature map. The last fully connected layer uses a softmax activation function to predict one of the 1000 categories. The dimensions corresponding to each layer are also shown.

the ImageNet competition, where the network was trained on millions on images with 1000 categories [52]. Furthermore, VGG-19 was used with good results in the original Faster R-CNN study [44], which motivated us to use it also in our model. As can be seen in Figure 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 3.

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 [52]. 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 [44], 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.

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 [51], 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 [51], in the Fast R-CNN component of the Faster R-CNN model, the external region proposal method is replaced by an internal RPN, described in the previous subsection, which is trained to identify regions of interest [44]. 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 [51].

3.5 Faster R-CNN Training

The Fast R-CNN network and the region proposal network share several convolutional layers, specifically the 13 convolutional layers of VGG-16. We have initialized the parameters of the 13 convolutional layers using the VGG-16 network pre-trained on the ImageNet dataset. As many image features are highly transferable between different datasets, this initialization based on VGG-16 allowed us to learn accurate models from a relatively small number of root cross-section labeled images.

Given that the region proposal network and the Fast R-CNN network share 13 convolutional layers, they are 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) [44]. All the model parameters are updated using stochastic gradient descent (SGD).

In our preliminary experimentation, we found that it is difficult to accurately detect the late metaxylem at the same time with the root and stele. Thus, we have trained a Faster R-CNN model to detect root and stele, and another Faster R-CNN model to detect late metaxylem.

4 Experiments and Results

Our goal is to utilize the proposed approach to identify and quantify root anatomical parameters based on cross-section images from genetically diverse accessions in different crops. We evaluated the approach using rice images as a case study.

4.1 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 obtain fully mature roots. The roots were harvested and washed thoroughly and stored in 40% alcohol. 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. Images of root sections were acquired with the Axioplan 2 compound microscope (Zeiss,

Table 1 Ground Truth Statistics. Statistics such as minimum (Min), maximum (Max), and average together with standard deviation (Avg ±std) are shown for the ground truth measurements, including RD, SD, LMXD and LMXN.

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

Germany) at 50x and 100x magnification. For each of the 25 rice accessions included in the study, three biological replicate root samples from root-shoot junction and 6 cm from the root tip were obtained. From each replicate, 2-3 images were taken at root-shoot junction, and 2-3 images at 6 cm from the tip of the root. Images may have two versions: a 50× magnification version, which captures the whole root diameter (top image in Fig. 1), and a 100× magnification version, which captures only the stele diameter (bottom image in Fig. 1). However, not all 50× images have a 100× correspondent. Specifically, 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 [47]. The manual measurements and counts constitute our ground truth to which we compared the measurements produced by our models. Statistics about the dataset, including the minimum, maximum, average and standard deviation for root diameter, stele diameter, late metaxylem average diameter and late metaxylem number, respectively, are presented in Table 1.

In addition, 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 the late metaxylem. We used the LabelImg tool [55] to perform the bounding box labeling. This tool produces annotations in the Pascal Visual Object Classes (VOC) XML format [56], a standard format used for annotating images with rectangular bounding boxes corresponding to objects. The bounding boxes 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.

4.2 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. For each fold, Table 2 also shows the number of corresponding 100× magnification images (instances) available (note that 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 were averaged 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 Fold distribution. 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 in the last column.

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

4.3 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 with a preference for higher values. Finally, we used the root mean square error (RMSE) metric to measure the ability of the proposed approach 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, smaller values are better.

4.4 Hyper-parameter Tuning

Deep learning models, in general, and the Faster R-CNN models, in particular, have many tunable hyper-parameters. We tuned several hyper-parameters shown by Zhang et al. [57] to affect the performance of the Faster R-CNN models, and used the values suggested by Ren et al. [44] 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 valued 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 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 bounded boxes with the ground truth bounded 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 problem. 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 consisted of the following values: 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 the our experiments described in the next section.

4.5 Experimental Results

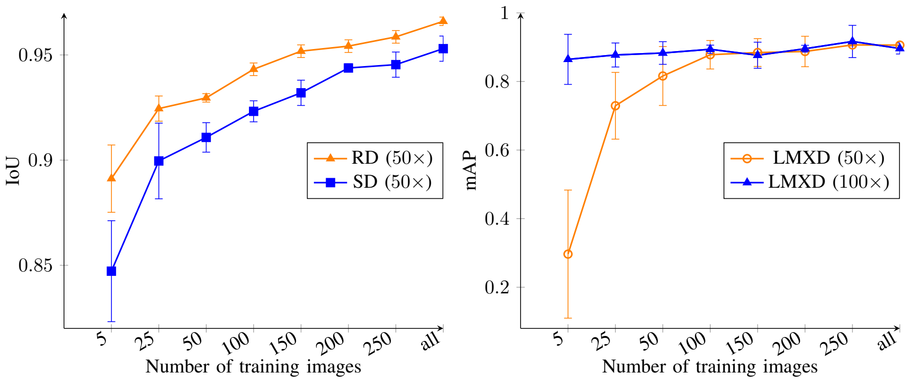

Figure 4 Variation of performance with the number of training images for root/stele detection model (Left plot), and for the LMX detection model (Right plot), respectively. We used 50× magnification images to detect root and stele objects, and both 50× and 100× magnification images to detect LMX. The 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.

4.5.1 Variation of Performance 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, our proposed deep learning 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 that are generally more complex than root, stele and late metaxylem.

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 that split, respectively, to train models for detecting the root, stele and late metaxylem in an images. The 50× magnification images were used to train the models for root/stele, while and 100× magnification images were used to train models for late metaxylem. The models learned were used to detect the root, stele, and LMX objects in the 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 bounded 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. The variation of performance with the number of training images is shown in Figure 4 for root/stele (Left plot) and LMX (Right plot). As can be seen, in the case of the 50× magnification images, the performance increases with the number of training images, but tends to stabilize generally around 250 images. This confirms our intuition that only a small number of labeled images is needed to learn accurate models for the problem at hand. 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, although the root bounding boxes are slightly better than the stele bounding boxes. Similarly, the LMX objects are detected with high accuracy, as shown on the right plot of Figure 4, where the mAP values are close to 0.9 consistently for models trained with smaller or larger 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, as shown in Figure 5.

4.5.2 Performance Evaluation Using RMSE

As described above, once the root/stele and LMX detection models were trained, they were used to detect root/stele/LMX objects in the test data. Subsequently,

Table 3 RMSE Results. The RMSE results for root diameter (RD), stele diameter (SD), late metaxylem diameter (LMXD) and late metaxylem number (LMXN) for 5 splits, together with the average and standard deviation over the 5 splits. 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 that were used to train the model that produced that measurement (i.e., 50× or 100×) is also shown. Furthermore, for each split, the test fold corresponding to that split is shown.

Split (Test Fold)	RD	SD	LMXD	LMXD	LMXN	LMXN
	(50×)	(50×)	(50×)	(100×)	(50×)	(100×)
Split 1 (Fold 5)	62.77	21.93	3.67	2.45	0.81	1.37
Split 2 (Fold 4)	32.18	17.54	3.77	3.13	0.71	0.45
Split 3 (Fold 3)	61.19	21.96	3.53	3.22	0.91	0.83
Split 4 (Fold 2)	33.12	20.01	3.58	3.56	1.90	0.63
Split 5 (Fold 1)	43.67	20.94	2.43	1.61	0.74	0.25
Average	46.59	20.39	3.40	2.79	1.02	0.71
Standard deviation	14.77	1.81	0.55	0.77	0.50	0.43

the detected objects were further used to calculate RD, SD, LMXD and LMXN. To evaluate the models in terms of their ability to produce the right root/stele/LMX diameter and LMX number, we have used the RMSE error computed by comparing the measurement/count estimates obtained from the predicted bounded 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 results of the experiments corresponding to the five splits, together with their average and standard deviation, are shown in Table 3.

As can be seen from Table 3, the average RMSE error for RD over the 5 splits is 46.59 μm. Given that root diameter for the images in our dataset varies between 354 μm and 1352 μm (see Table 1), this result is very encouraging. Similarly, the average RMSE error for SD over the five splits is 20.39 μm, which is low, given that the stele diameter varies between 115 μm and 419 μm. As opposed to root and stele, the LMXD is significantly smaller, varying between 15 μm and 65 μm. However, the average RMSE error is 3.40 μm for the model trained using the 50× magnification images, and decreases by almost 1 μm for the model trained with the 100× magnification images (the exact value is 2.79 μm). In terms of LMXN, the ground truth numbers vary between 1 and 12, with an average of 5.4 LMX objects per image. The average RMSE error for LMXN is 1.02 for the models trained on the 50× magnification images, and down to 0.71 for the models trained on the 100× magnification images. Thus, we can say that our models miss roughly one LMX per image, when trained with the 50× magnification images, and less than that, when trained with the 100× magnification images. We performed error analysis to understand if these results might be useful 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 5. This observation is not surprising, as human annotators are prone to mistakes and inconsistencies. As opposed to that, the automated models produced by our proposed approach produce more consistent results, once they are well trained. At last, Table 3 shows that the RMSE results obtained do not vary significantly with the split, as shown by the relatively small standard deviation. Together, these results suggest that the proposed approach has the potential to replace the labor-intensive manual annotations of root cross-section images.

4.5.3 Comparison with RootAnalyzer and RootScan

We aimed to compare DL-RootAnatomy with RootAnalyzer and RootScan tools on all 388 50× magnification images in our dataset.

Given the batch processing capability of RootAnalyzer by comparison with the amount of user effort involved by the RootScan, we started our comparative analysis with RootAnalyzer. As described in Section 2, RootAnalyzer has five parameters that need to be tuned: W, T, A_l, A_s and N. To understand the range of parameters that our images require for good performance, we experimented with a variety of images and parameters, and observed that RootAnalyzer freezes for some images, regardless of the parameters used. Specifically, it freezes or produces degenerate results on images for which the root has a dark, solid boundary, and the epidermal cells are not clearly identifiable, while it works as expected on images for which the root has identifiable epidermal cells. Examples of images that can or cannot be analyzed by RootAnalyzer are shown in Figure 6 (a) and (b), respectively. Out of the 388 images, we identified 323 with clear epidermal cells and used those in the

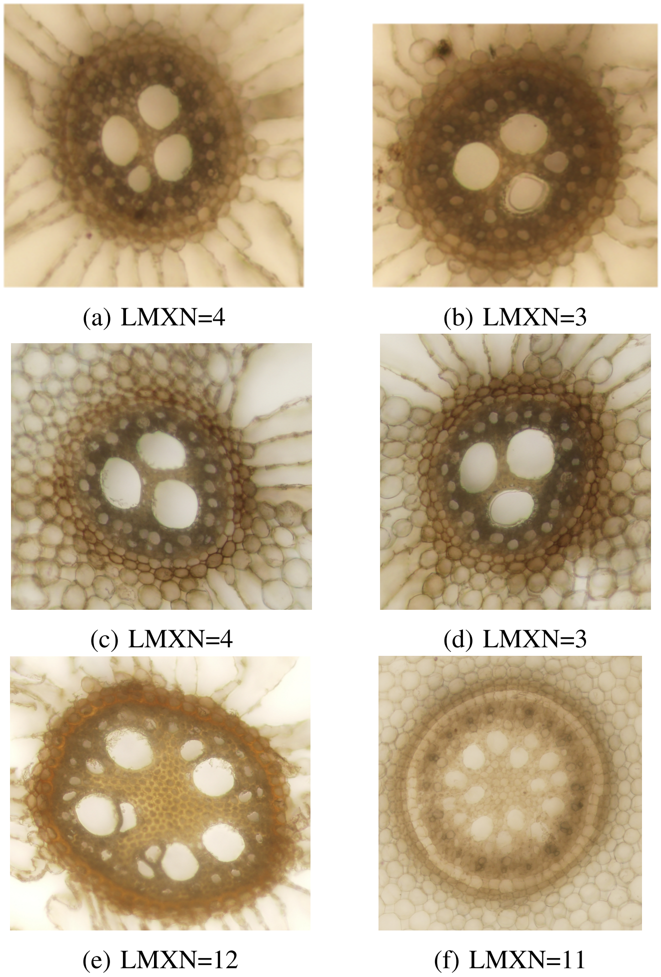
(a) LMXN=4
(b) LMXN=3
(c) LMXN=4
(d) LMXN=3
(e) LMXN=12
(f) LMXN=11

Figure 5 Examples of inconsistent annotations in our dataset. Specifically, image (a) was 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)). Similarly, image (c) was incorrectly labeled as having LMXN=4, while the similar image in (d) was properly labeled as having LMNX=3. 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 and which were not counted to get the 12 and 11 counts, respectively.

Table 4 Number of instances that can be analyzed with RootAnalyzer in each of the 5 folds, out of the total number of instances in each fold. These are the images that were used in the comparison between our approach and existing tools. The total number of instances used in the comparison is also shown in the last column.

Fold	Fold 1	Fold 2	Fold 3	Fold 4	Fold 5	Total
Instances	57	70	73	58	65	323
Total-fold	71	79	86	77	75	388

comparison between our tool and other related tools. The distribution of the 323 images analyzed with RootAnalyzer over the 5 folds is shown in Table 4.

As mentioned before, RootScan is a semi-automated tool, which requires human interaction/approval at each stage. For example, after automatically detecting the root border (and similarly the stele border and late metaxylem border), the tool presents the user with the opportunity to manually redraw or adjust the border, if the automatically detected border does not look as expected. We ran RootScan with human interaction to estimate the human error/bias in our dataset, under the assumption that without human error, the adjusted borders should lead to minimal differences between the original ground truth manual measurements and the RootScan measurements. We refer to this experiment as RootScan (adjusted) in what follows. We also run RootScan in an automated fashion, where we approved the borders identified by the tool, without any further adjustment. We refer to this experiment as RootScan (automated).

First, we ran RootAnalyzer on the 323 images in batch mode, with a set of overall good parameters manually identified in our preliminary examination of these images. With just one click, all the images were processed and the results were output in a csv file. We then fed the 323 images into the semi-automated tool, RootScan, and performed experiments in either the automated or adjusted modes. We used the measurements produced by each tool to calculate the RMSE error for each fold and averaged the error over the 5 folds. Finally, we tested our DL-RootAnatomy approach on the 323 images using the 5-fold cross-validation setup. For LMXD and LMXN, we used both models trained on the 50× magnification images and models trained on the 100× magnification images. The results of the experiments described are shown in Table 5.

As can be seen from Table 5, our proposed approach, DL-RootAnatomy, is the most accurate when compared to the existing tools, specifically RootAnalyzer and

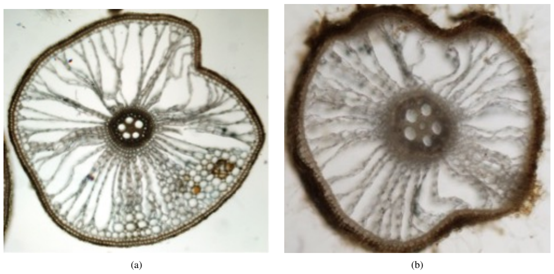

Figure 6 Examples of root boundaries: (a) boundary with clear and identifiable epidermal cell; (b) dark solid boundary with unclear or unidentifiable epidermal cell.

Table 5 Comparison between the RMSE results of our proposed approach (called DL-RootAnatomy), and the RMSE results of RootAnalyzer, RootScan (automated) and RootScan (adjusted). The number of images used in these experiments is 323. RootScan (adjusted) is seen as an estimate of the human error. DL-RootAnatomy was run on both $50\times$ and $100\times$ magnification images to detect LMX objects. The other tools were used with the $50\times$ magnification images (as they do not work properly with $50\times$ magnification images). The RMSE is calculated for: root diameter (RD), stele diameter (SD), LMX diameter (LMXD), and LMX number (LMXN). The results are averaged over five splits. Corresponding to each average, standard deviation is also show.

Method	RD	SD	LMXD	LMXN
RootAnalyzer	208.44 ± 22.40	172.21 ± 20.65	32.89 ± 10.62	4.01 ± 0.54
RootScan (automated)	132.33 ± 40.08	428.89 ± 13.29	45.20 ± 2.88	19.58 ± 1.66
DL-RootAnatomy ($50\times$)	**43.67 ± 16.80**	**20.51 ± 1.84**	**3.58 ± 0.57**	**1.13 ± 0.43**
DL-RootAnatomy ($100\times$)	N/A	N/A	**2.79 ± 0.93**	**0.64 ± 0.31**
RootScan (adjusted) ≋ Human Error	66.82 ± 20.86	42.27 ± 25.54	6.26 ± 2.39	0.72 ± 0.23

RootScan. First, the DL-RootAnatomy models have an average RMSE error of 43.67 μm for RD (slightly different from the error reported in Table 3, as a smaller number of images were used in this experiment, as described above). The RootScan (adjusted) average RMSE, which is assumed to approximate the human error in the ground truth, is 66.82 μm, slightly higher than the error obtained by the DL-RootAnatomy. Second, our approach gives the smallest error also for SD, specifically, 20.51 μm, followed again by RootScan (adjusted) with an error of 42.27. Third, our approach has the smallest error for LMXD (3.58 μm for the 50×, and 2.79 μm for the 100× images), with the RootScan (adjusted) as the second winner (with RMSE 6.26 μm). Finally, in terms of LMXN, our models trained with 50× images are slightly worse than RootScan (adjusted) (1.13 versus 0.72), while the models trained with 100× images are slightly better than RootScan (adjusted)

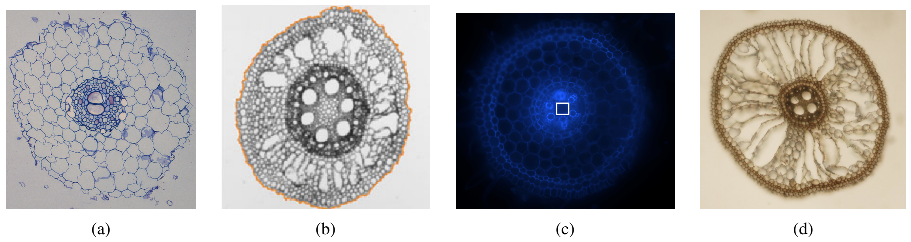

Figure 7 (Top) RootAnalyzer: With the same set of parameters, in the left image the root border (red), stele border (yellow), endodermis (green) and late-metaxylem (purple) are detected reasonably well, while in the right image, only half of the stele border is detected. Given that the tool fails to properly detect the stele border, it also fails to detect the late metaxylem. (Bottom) RootScan: In the left image, RootScan fails to detect the stele border. In the right image, it correctly identifies three out of six late metaxylem, and introduces one false positive.

(0.64 versus 0.72). The other two automated tools, RootAnalyzer and RootScan (automated), give significantly higher errors overall, with RootAnalyzer being better between the two, but not good enough to be used for the automated analysis of rice images.

Thus, we claim that the existing tools do not work well on the rice root images studied in this article. We identified several possible reasons:

(1) For a given tool, it is hard to find parameters that are universally good for all images in our dataset. For example, for a given set of parameters, the segmentation result from the RootAnalyzer in Figure 7 (Top) shows that the parameters are appropriate for the left image (a), but not appropriate for the right image (b). Similarly, the default parameters used by RootScan do not work well for all images, as can be seen in Figure 7 (Bottom), where the RootScan does a poor job at identifying the stele border in the image on the left (c), and also a poor job at identifying LMX in the image on the right (d).

(2) Input images are grown in different conditions, for example in hydroponic (water based nutrient supply) or in soil, and/or are collected using different techniques. Growing or collection differences lead to differences in image's color, contrast and brightness. Figure 8 shows input images for RootAnalyzer, RootScan, PHIV-RootCell, and DL-RootAnatomy, respectively.

(3) Each tool is designed with certain image characteristics in mind, and may not work on images that do not exhibit those characteristics. As described above,

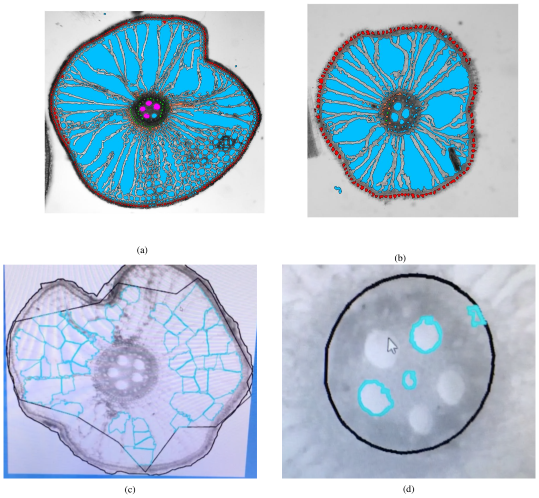

Figure 8 Sample input images used by different tools. (a) Sample input image for RootAnalyzer, which shows a clear difference between background pixels and cell border pixels. (b) Sample input image for RootScan. (c) Sample input image for PHIV-RootCell, which works with root cross-section images that contain a central metaxylem (marked with a white rectangle). (d) Sample root cross-section image from the dataset used in this study.

RootAnalyzer assumes a clear cell boundary and does not work for images that contain a solid boundary where the cells are not clearly identifiable.

(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 [38].

5 Conclusions and Future Work

In this paper, we presented a fully automated approach for processing root cross-section images to extract anatomical root features. The approach is based on Faster R-CNN networks, which were used 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. Our Faster R-CNN models 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.

We evaluated the models in terms of their ability to detect the objects of interest, and also in terms of their ability to lead to accurate measurements for RD, SD, LMXD and LMXN. The results of the evaluation showed that our models produced accurate and consistent annotations, when trained on a relatively small number of training images. Furthermore, a comparison with existing tools for analyzing root cross-section images showed that our automated tool identifies anatomical features more accurately than those obtained with tools that require manual adjustment. Overall, these results suggest that our tool, DL-RootAnatomy, can potentially be used in practice to accelerate the speed at which root cross-section images are analyzed, and save significant human efforts and costs.

As part of future work, we plan to package our approach as a user-friendly standalone applications that allows easy training and annotation of images by agricultural domain experts. The evaluation in this paper was done on rice images. We plan to study domain adaptation approaches that allow the transfer of knowledge from rice models to models for other plant species, without labeling a large number of images from the other species of interest.

Availability of data and materials

The datasets used and analyzed during the current study are available from the corresponding author on reasonable request.

Competing interests

The authors declare that they have no competing interests.

Author's contributions

XL carried out the model design and implementation, with input from DC. XL, CW and DC carried out the computational experiment design, with input from SVKJ. RB and SVKJ performed the biological experiment design and collection of the data. XL and CW carried out all the computational experiments. RB performed the labeling of the data according to RD, SD, LMXD and LMXN measurements. XL and CW performed the bounding box labeling. SVKJ is the agronomy project leader with technical background in root phenotyping. DC is the computational project leader, with background in machine learning and deep learning. XL and CW drafted the first version of the manuscript, and DC and SVKJ contributed to the preparation of the final version of the manuscript. RJ contributed biological knowledge to the manuscript and provided feedback on the preliminary version. All authors read and approved the final manuscript.

Acknowledgements

Contribution No. 19-072-J from Kansas Agriculture Experiment Station . . .

Author details

[1]Department of Computer Science, Kansas State University, Manhattan, KS, 66502, USA. [2]Department of Agronomy, Kansas State University, Manhattan, KS, 66502, USA.

References

1. Araus, J.L., Slafer, G.A., Royo, C., Serret, M.D.: Breeding for yield potential and stress adaptation in cereals. Critical Reviews in Plant Sciences **27**(6), 377–412 (2008). doi:10.1080/07352680802467736. https://doi.org/10.1080/07352680802467736
2. S., K.G.: Strategies for increasing the yield potential of cereals: case of rice as an example. Plant Breeding **132**(5), 433–436 (2013). doi:10.1111/pbr.1991. https://onlinelibrary.wiley.com/doi/pdf/10.1111/pbr.1991
3. Bishopp, A., Lynch, J.: The hidden half of crop yields **1**, 15117 (2015)
4. Jung, J., Mccouch, S.: Getting to the roots of it: Genetic and hormonal control of root architecture **4**, 186 (2013)
5. E. Schmidt, J., C.M. Gaudin, A.: Toward an integrated root ideotype for irrigated systems **22** (2017)
6. Uga, Y., Sugimoto, K., Ogawa, S., Rane, J., Ishitani, M., Hara, N., Kitomi, Y., Inukai, Y., Ono, K., Kanno, N., Inoue, H., Takehisa, H., Motoyama, R., Nagamura, Y., wu, J., Matsumoto, T., Takai, T., Okuno, K., Yano, M.: Control of root system architecture by deeper rooting 1 increases rice yield under drought conditions **45** (2013)
7. Bucksch, A., Burridge, J., York, L.M., Das, A., Nord, E., Weitz, J.S., Lynch, J.P.: Image-based high-throughput field phenotyping of crop roots. Plant Physiology (2014). doi:10.1104/pp.114.243519. http://www.plantphysiol.org/content/early/2014/09/03/pp.114.243519.full.pdf
8. Mooney, S.J., Pridmore, T.P., Helliwell, J., Bennett, M.J.: Developing x-ray computed tomography to non-invasively image 3-d root systems architecture in soil. Plant and Soil **352**(1), 1–22 (2012). doi:10.1007/s11104-011-1039-9
9. Schulz, H., Postma, J.A., van Dusschoten, D., Scharr, H., Behnke, S.: Plant root system analysis from mri images. In: Csurka, G., Kraus, M., Laramee, R.S., Richard, P., Braz, J. (eds.) Computer Vision, Imaging and Computer Graphics. Theory and Application, pp. 411–425. Springer, Berlin, Heidelberg (2013)
10. Topp, C.N., Iyer-Pascuzzi, A.S., Anderson, J.T., Lee, C.-R., Zurek, P.R., Symonova, O., Zheng, Y., Bucksch, A., Mileyko, Y., Galkovskyi, T., Moore, B.T., Harer, J., Edelsbrunner, H., Mitchell-Olds, T., Weitz, J.S., Benfey, P.N.: 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), 1695–1704 (2013). doi:10.1073/pnas.1304354110. http://www.pnas.org/content/110/18/E1695.full.pdf
11. van Dusschoten, D., Metzner, R., Kochs, J., Postma, J.A., Pflugfelder, D., Buehler, J., Schurr, U., Jahnke, S.: Quantitative 3d analysis of plant roots growing in soil using magnetic resonance imaging. Plant Physiology (2016). doi:10.1104/pp.15.01388. http://www.plantphysiol.org/content/early/2016/01/04/pp.15.01388.full.pdf
12. Pfeifer, J., Kirchgessner, N., Colombi, T., Walter, A.: Rapid phenotyping of crop root systems in undisturbed field soils using x-ray computed tomography **11:14** (2015)
13. Durham Brooks, T.L., Miller, N.D., Spalding, E.P.: Plasticity of arabidopsis root gravitropism throughout a multidimensional condition space quantified by automated image analysis. Plant Physiology **152**(1), 206–216 (2010). doi:10.1104/pp.109.145292. http://www.plantphysiol.org/content/152/1/206.full.pdf
14. Clark, R.T., MacCurdy, R.B., Jung, J.K., Shaff, J.E., McCouch, S.R., Aneshansley, D.J., Kochian, L.V.: Three-dimensional root phenotyping with a novel imaging and software platform. Plant Physiology **156**(2), 455–465 (2011). doi:10.1104/pp.110.169102. http://www.plantphysiol.org/content/156/2/455.full.pdf
15. Sozzani, R., Busch, W., Spalding, E.P., Benfey, P.N.: Advanced imaging techniques for the study of plant growth and development. Trends in Plant Science **19**(5), 304–310 (2014). doi:10.1016/j.tplants.2013.12.003
16. Yip, K.Y., Cheng, C., Gerstein, M.: Machine learning and genome annotation: a match meant to be? Genome biology **14**(5), 205 (2013)
17. Rhee, S.Y., Mutwil, M.: Towards revealing the functions of all genes in plants. Trends in plant science **19**(4), 212–221 (2014)
18. DePristo, M.A., Banks, E., Poplin, R., Garimella, K.V., Maguire, J.R., Hartl, C., Philippakis, A.A., Del Angel, G., Rivas, M.A., Hanna, M., *et al.*: A framework for variation discovery and genotyping using next-generation dna sequencing data. Nature genetics **43**(5), 491 (2011)
19. Heslot, N., Yang, H.-P., Sorrells, M.E., Jannink, J.-L.: Genomic selection in plant breeding: a comparison of models. Crop Science **52**(1), 146–160 (2012)
20. Khan, Z., Rahimi-Eichi, V., Haefele, S., Garnett, T., Miklavcic, S.J.: Estimation of vegetation indices for

high-throughput phenotyping of wheat using aerial imaging. Plant methods **14**(1), 20 (2018)

21. Kamilaris, A., Prenafeta-Boldú, F.X.: Deep learning in agriculture: A survey. Computers and Electronics in Agriculture **147**, 70–90 (2018)
22. Jones, W., Alasoo, K., Fishman, D., Parts, L.: Computational biology: deep learning. Emerging Topics in Life Sciences **1**(3), 257–274 (2017)
23. Tardieu, F., Cabrera-Bosquet, L., Pridmore, T., Bennett, M.: Plant phenomics, from sensors to knowledge. Current Biology **27**(15), 770–783 (2017)
24. Singh, A., Ganapathysubramanian, B., Singh, A.K., Sarkar, S.: Machine learning for high-throughput stress phenotyping in plants. Trends in Plant Science **21**(2), 110–124 (2016). doi:10.1016/j.tplants.2015.10.015
25. Pound, M.P., Atkinson, J.A., Townsend, A.J., Wilson, M.H., Griffiths, M., Jackson, A.S., Bulat, A., Tzimiropoulos, G., Wells, D.M., Murchie, E.H., Pridmore, T.P., French, A.P.: Deep machine learning provides state-of-the-art performance in image-based plant phenotyping. GigaScience **6**(10), 1–10 (2017)
26. Namin, S.T., Esmaeilzadeh, M., Najafi, M., Brown, T.B., Borevitz, J.O.: Deep phenotyping: Deep learning for temporal phenotype/genotype classification. bioRxiv, 134205 (2017)
27. Aich, S., Stavness, I.: Leaf counting with deep convolutional and deconvolutional networks. arXiv preprint arXiv:1708.07570 (2017)
28. Dobrescu, A., Giuffrida, M.V., Tsaftaris, S.A.: Leveraging multiple datasets for deep leaf counting. In: ICCV Workshops 2017, pp. 2072–2079 (2017)
29. Ubbens, J.R., Stavness, I.: Deep plant phenomics: a deep learning platform for complex plant phenotyping tasks. Frontiers in plant science **8**, 1190 (2017)
30. Namin, S.T., Esmaeilzadeh, M., Najafi, M., Brown, T.B., Borevitz, J.O.: Deep phenotyping: deep learning for temporal phenotype/genotype classification. Plant methods **14**(1), 66 (2018)
31. Kuijken, R.C., van Eeuwijk, F.A., Marcelis, L.F., Bouwmeester, H.J.: Root phenotyping: from component trait in the lab to breeding. Journal of experimental botany **66**(18), 5389–5401 (2015)
32. Jiangsan, Z., Peter, S., Gernot, B., Boris, R.: Root traits of european vicia faba cultivars-using machine learning to explore adaptations to agroclimatic conditions. Plant, Cell & Environment **0**(0) (2017). doi:10.1111/pce.13062. https://onlinelibrary.wiley.com/doi/pdf/10.1111/pce.13062
33. Delory, B.M., Li, M., Topp, C.N., Lobet, G.: archidart v3. 0: A new data analysis pipeline allowing the topological analysis of plant root systems. F1000Research **7** (2018)
34. Pound, M.P., Fozard, S., Torres, M.T., Forde, B.G., French, A.P.: Autoroot: open-source software employing a novel image analysis approach to support fully-automated plant phenotyping. Plant methods **13**(1), 12 (2017)
35. Betegón-Putze, I., González, A., Sevillano, X., Blasco-Escámez, D., Caño-Delgado, A.I.: Myroot: A novel method and software for the semi-automatic measurement of plant root length. bioRxiv, 309773 (2018)
36. Reeb, C., Kaandorp, J., Jansson, F., Puillandre, N., Dubuisson, J.-Y., Cornette, R., Jabbour, F., Coudert, Y., Patiño, J., Flot, J.-F., *et al.*: Quantification of complex modular architecture in plants. New Phytologist **218**(2), 859–872 (2018)
37. Burton, A.L., Williams, M., Lynch, J.P., Brown, K.M.: Rootscan: software for high-throughput analysis of root anatomical traits. Plant and Soil **357**(1-2), 189–203 (2012)
38. Chopin, J., Laga, H., Huang, C.Y., Heuer, S., Miklavcic, S.J.: Rootanalyzer: a cross-section image analysis tool for automated characterization of root cells and tissues. PloS one **10**(9), 0137655 (2015)
39. Lartaud, M., Perin, C., Courtois, B., Thomas, E., Henry, S., Bettembourg, M., Divol, F., Lanau, N., Artus, F., Bureau, C., *et al.*: Phiv-rootcell: a supervised image analysis tool for rice root anatomical parameter quantification. Frontiers in plant science **5**, 790 (2015)
40. Hall, H.C., Fakhrzadeh, A., Luengo Hendriks, C.L., Fischer, U.: Precision automation of cell type classification and sub-cellular fluorescence quantification from laser scanning confocal images. Frontiers in plant science **7**, 119 (2016)
41. Sankar, M., Nieminen, K., Ragni, L., Xenarios, I., Hardtke, C.S.: Automated quantitative histology reveals vascular morphodynamics during arabidopsis hypocotyl secondary growth. Elife **3** (2014)
42. Kadam, N., Yin, X., Bindraban, P., Struik, P., Jagadish, K.: Does morphological and anatomical plasticity during the vegetative stage make wheat more tolerant of water-deficit stress than rice? Plant physiology, 114 (2015)

43. Kadam, N., Tamilselvan, A., Lawas, L.M.F., Quinones, C., Bahuguna, R., Thomson, M.J., Dingkuhn, M., Muthurajan, R., Struik, P., Yin, X., Jagadish, K.: Genetic control of plasticity in root morphology and anatomy of rice in response to water-deficit. Plant physiology, 00500 (2017)
44. Ren, S., He, K., Girshick, R., Sun, J.: Faster R-CNN: Towards real-time object detection with region proposal networks. In: Advances in Neural Information Processing Systems, pp. 91–99 (2015)
45. in Plant Phenotyping (CVPPP), C.V.P.: Leaf Segmentation Challenge (LFC). https://www.plant-phenotyping.org/cvppp2014-challenge Accessed 2018-10-02
46. Mohanty, S.P., Hughes, D.P., Salathé, M.: Using deep learning for image-based plant disease detection. Frontiers in plant science **7**, 1419 (2016)
47. Schneider, C.A., Rasband, W.S., Eliceiri, K.W.: Nih image to imagej: 25 years of image analysis. Nature methods **9**(7), 671 (2012)
48. Yamauchi, T., Watanabe, K., Fukazawa, A., Mori, H., Abe, F., Kawaguchi, K., Oyanagi, A., Nakazono, M.: 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)
49. Otsu, N.: A threshold selection method from gray-level histograms. IEEE transactions on systems, man, and cybernetics **9**(1), 62–66 (1979)
50. Wunderling, A., Ben Targem, M., Barbier de Reuille, P., Ragni, L.: Novel tools for quantifying secondary growth. Journal of experimental botany **68**(1), 89–95 (2016)
51. Girshick, R.: Fast r-cnn. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 1440–1448 (2015)
52. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556 (2014)
53. LeCun, Y., Boser, B., Denker, J.S., Henderson, D., Howard, R.E., Hubbard, W., Jackel, L.D.: Backpropagation applied to handwritten zip code recognition. Neural computation **1**(4), 541–551 (1989)
54. LeCun, Y., Bengio, Y., Hinton, G.: Deep learning. nature **521**(7553), 436 (2015)
55. Tzutalin: LabelImg. https://github.com/tzutalin/labelImg
56. Everingham, M., Eslami, S.M.A., Van Gool, L., Williams, C.K.I., Winn, J., Zisserman, A.: The pascal visual object classes challenge: A retrospective. International Journal of Computer Vision **111**(1), 98–136 (2015)
57. Zhang, L., Lin, L., Liang, X., He, K.: Is faster r-cnn doing well for pedestrian detection? In: European Conference on Computer Vision, pp. 443–457 (2016). Springer