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

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

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

Abstract

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

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

1. Introduction

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

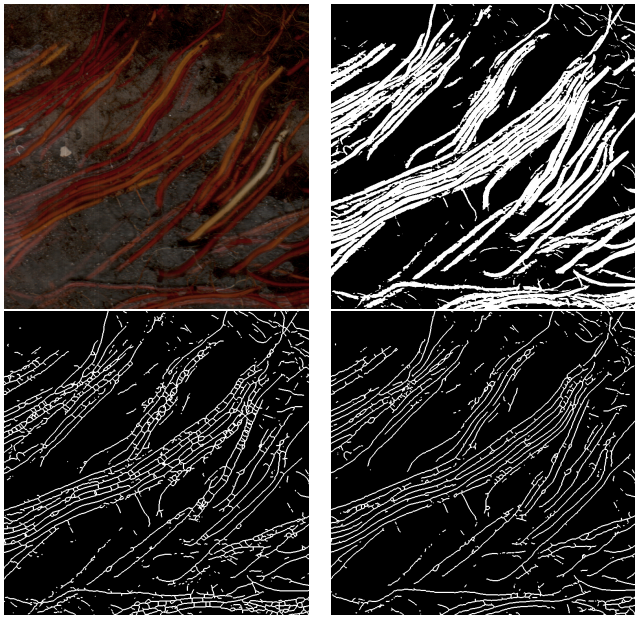

Figure 1: Top left: input image. Top right: output of a U-Net segmentation network that was trained to detect roots. Bottom left: the same output after applying the skeletonization postprocessing step. Many artifacts and loops are present due to the imperfection of the segmentation map. Bottom right: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

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

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

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

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

2. Related Work

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

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

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

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

3. Methods

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

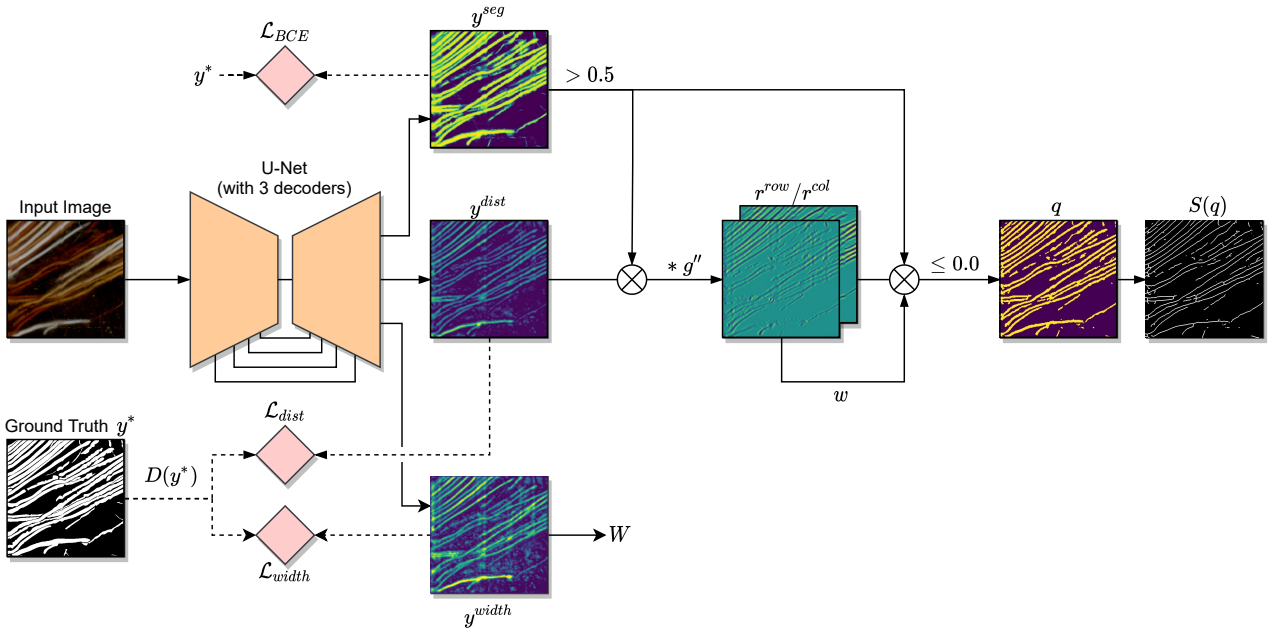

Figure 2: Schematic overview of our method. Solid lines represent data flow, dashed lines represent losses.

3.1. Network Architecture and Training

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

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

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

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

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

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

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

The overall loss function is given by:

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

where λ_0 and λ_1 are balancing hyperparameters which we both set to 0.01. We have found larger values to have a negative effect on the main segmentation head.

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

3.2. Ridge Detection

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

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

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

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

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

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

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

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

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

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

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

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

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

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

3.3. Width Estimation

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

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

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

4. Experimental Setup

4.1. Dataset

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

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

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

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

4.2. Metrics

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

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

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

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

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

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

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

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

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

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

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

4.3. Compared Methods

For skeleton metrics, we compare the following approaches:

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

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

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

5. Results

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

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

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

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

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

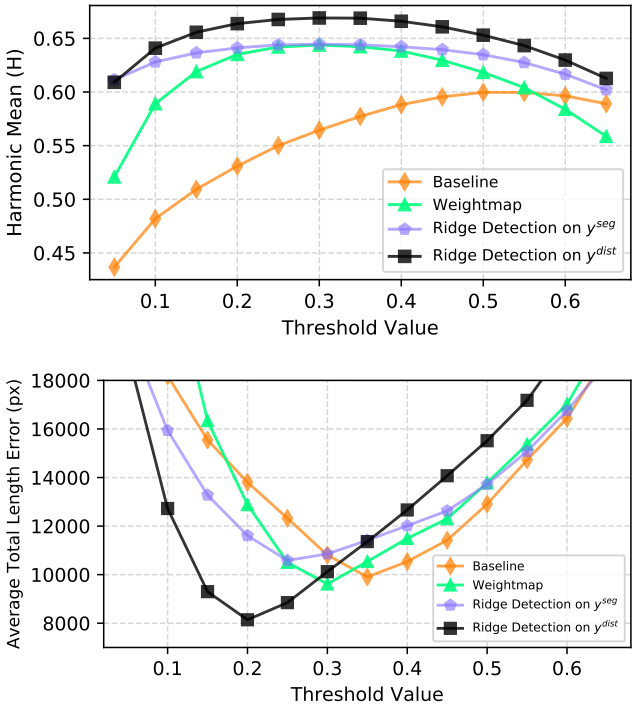

Figure 3: Influence of the segmentation threshold on the skeleton metrics and total root length. Advanced methods like ridge detection benefit more from a lower threshold than the baseline. (Evaluated on dense root systems.)

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

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

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

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

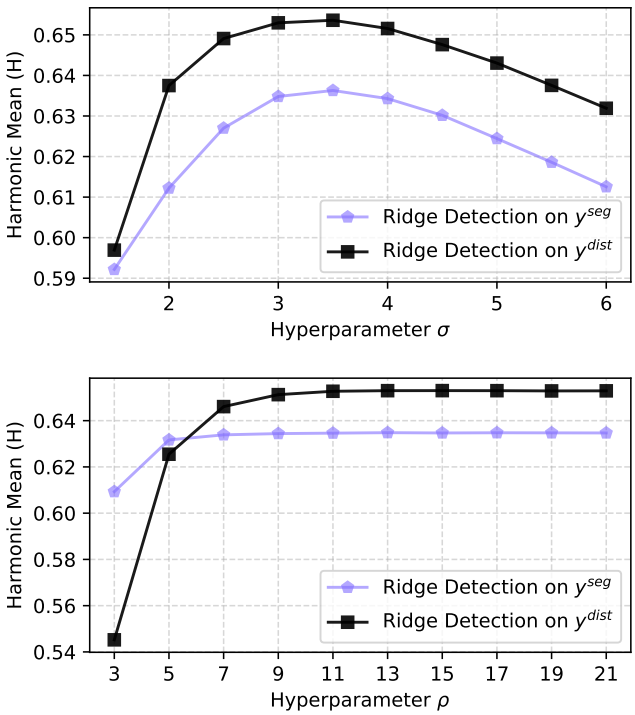

Figure 4: Influence of hyperparameters on the performance. Top: The width of the gaussian kernel σ has an optimium in the range 3 to 4. Bottom: The dilation parameter ρ for the width map computation should have minimum value of 9. Larger values do not lead to much improvement but only increase computation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

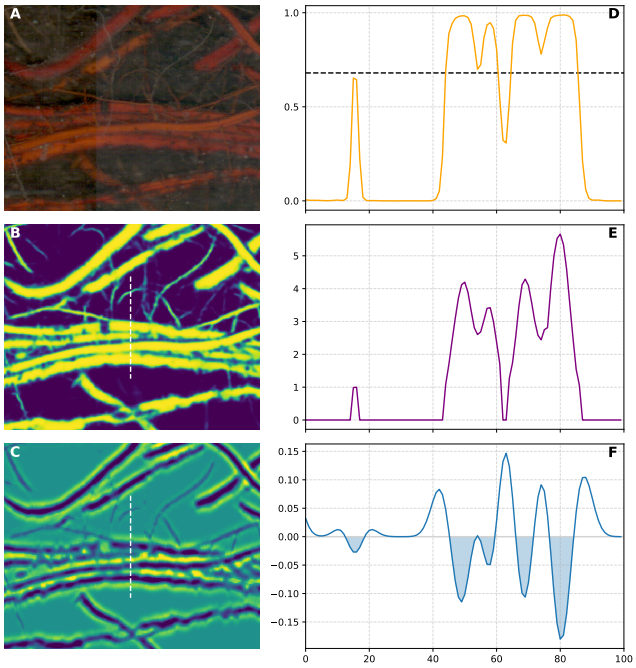

Figure 5: A particularly challenging example. (A) patch of an input image; (B) Output of the segmentation head y^{seg}; (C) $\hat{r}$ as computed in equation 7; (D) plot of y^{seg} extracted at the white dashed line. Not all roots can be detected or separated via simple thresholding, no matter where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

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

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

6. Conclusion

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

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

ACKNOWLEDGEMENTS

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

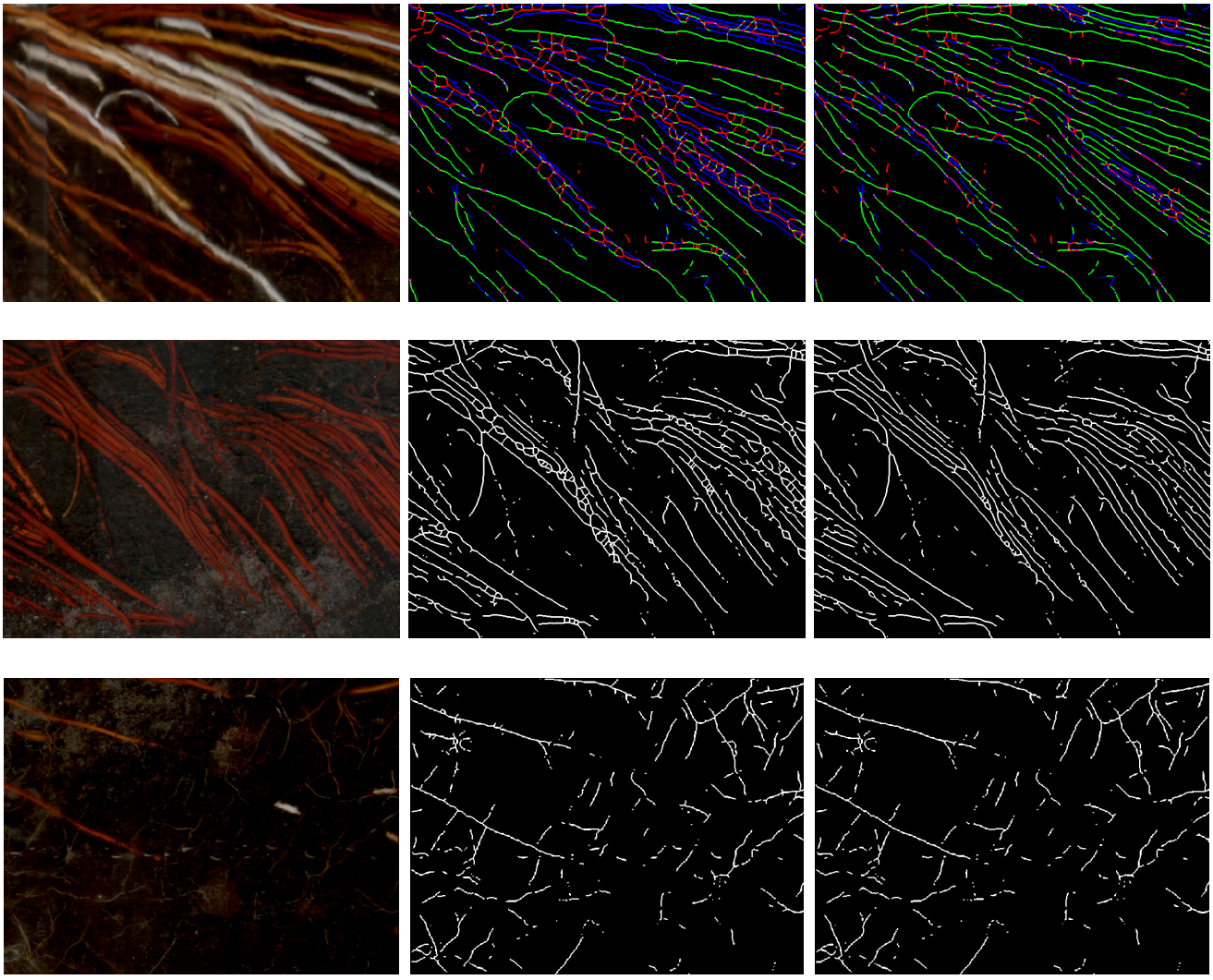

Figure 6: Some qualitative example results. From left to right: input image, baseline skeletonization, our method. In the top row green pixels represent TP_ρ, red pixels FP_ρ, blue pixels FN_ρ. Note the blurriness due to high humidity in this image. The bottom row shows an example with a sparse root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

References

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

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

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

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

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

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

[7] Abraham George Smith, Jens Petersen, Raghavendra

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

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

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

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

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

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

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

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

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

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