Root Gap Correction with a Deep Inpainting Model

Hao Chen[1]
s1786991@ed.ac.uk
Mario Valerio Giuffrida[1]
v.giuffrida@ed.ac.uk
Peter Doerner[2]
peter.doerner@ed.ac.uk
Sotirios A. Tsaftaris[1]
S.Tsaftaris@ed.ac.uk

[1] School of Engineering,
The University of Edinburgh,
Edinburgh, UK

[2] School of Biological Sciences,
The University of Edinburgh,
Edinburgh, UK

Abstract

Imaging roots of growing plants in a non-invasive and affordable fashion has been a long-standing problem in image-assisted plant breeding and phenotyping. One of the most affordable and diffuse approaches is the use of mesocosms, where plants are grown in soil against a glass surface that permits the roots visualization and imaging. However, due to soil and the fact that the plant root is a 2D projection of a 3D object, parts of the root are occluded. As a result, even under perfect root segmentation, the resulting images contain several gaps that may hinder the extraction of finely grained root system architecture traits.

We propose an effective deep neural network to recover gaps from disconnected root segments. We train a fully supervised encoder-decoder deep CNN that, given an image containing gaps as input, generates an inpainted version, recovering the missing parts. Since in real data ground-truth is lacking, we use synthetic root images [10] that we artificially perturb by introducing gaps to train and evaluate our approach. We show that our network can work both in dicot and monocot cases in reducing root gaps. We also show promising exemplary results in real data from chickpea root architectures.

1 Introduction

Plant roots is an important component of plant architecture and function, which largely determines the growth and resilience of the plant towards several environmental stresses. Our ability to observe them in non-destructive fashion is challenging. While shoot observation is easy (it is above the ground), the root is hidden inside the soil. Several approaches exist aiming to either image the roots in-situ and non-destructively (in 2D or 3D with MRI) or destructively, by digging out the roots and imaging them in isolation. Focusing on non-destructive methods, most of these occur in specialized apparatuses, which aim to resemble growth conditions as close as possible to the field (ranging from hydroponic, on paper, on the gel and other approaches [12]). Of particular interest are methods that grow roots in the soil in *rhizotrons* or *mesocosms*, where root growth is observed through a (plexi) glass.

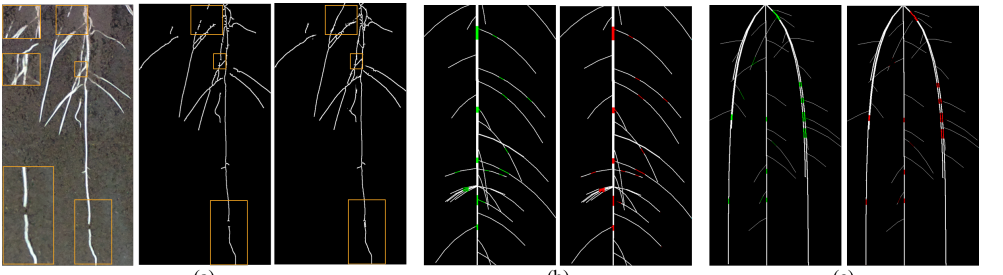

Figure 1: Examples of gap completion in a real image from a dicot (Chickpea) and in synthetic images. (a) An image of a chickpea root as obtained via digital cameras (left), and 3 zoomed-in areas (inlet) focusing on the suspected root gaps. The segmentation of the roots (middle) and the same inpainted with our algorithm (right). (b) A synthetic dicot root (left) containing numerous gaps in green. The inpainted gap results in red as obtained by our approach (right). (c) A synthetic monocot root image (right) containing numerous gaps in green, and inpainted results shown in red color (right).

The ability to image roots non-destructively allows for serial observation and via image analysis, several traits related to root system architecture (RSA) can be extracted. A fundamental step is the root segmentation to identify root pixels from the background (soil). Although mesocosms are very thin (e.g., 6mm), still roots have enough space to grow in it. Therefore, the root architecture visible through the glass results as a 2D projection of a 3D object. For this reason, a root segmentation may contain gaps. The presence of such gaps affects the extraction of fine-grained traits that rely on a precise semantic understanding of the root as a graph and phenotyping traits, such as number of laterals, number of root tips, etc. Thus, there is significant interest in developing reliable solutions that aim to resolve such gaps in images of roots obtained from rhizotrons/mesocosms (c.f. Figure 1(a)).

We address the root gap filling problem as an inpainting process [3]. In the literature, several approaches to reconstruct missing parts of an image have been proposed. As an example, similar problems are encountered in neuron, retinal, and satellite image processing [19, 21]. Also, in [17], they used a fully convolutional network to inpaint gaps in line drawings. (We further discuss related work in Section 2). However, to the best of our knowledge, the problem of root gap correction has not been addressed in the past and particularly not as an inpainting problem.

This paper presents a deep learning convolutional approach to the inpainting of roots. Our approach follows a classic encoder-decoder paradigm with fully convolutional layers. We built our architecture upon [17], where skip connections were added to learn better features on deeper layers [8]. Since absolute ground-truth from real data is very difficult to obtain, we used a publicly available dataset of synthetic root images [10] to validate our method. We artificially corrupted these images by introducing random gaps and train the model to infer a complete (inpainted) image, as Figure 1(b) shows. We train with the complete original images as output and the corrupted images as input. Our dataset contains both dicot and monocot synthetic roots. We used $7,000$ images for training, $1,500$ images for validation, and $1,500$ images for testing. Beyond image fidelity scores (e.g., mean absolute error), we used phenotyping traits, such as total root length, tip counts, and the number of connected components in graph theory for an application-driven evaluation.

The contributions of our work can be summarized as follows.

1. to the best of our knowledge, we are the first to apply an inpainting model to a root gap correction task, using a deep neural network;
2. we develop a U-Net-like network [16] with traditional convolutions that achieves state-of-the-art results on inpainting of thin-like structures, taking inspiration from [17];
3. on synthetic data we show that inpainted roots are close to complete roots and have improved phenotyping traits; and
4. we demonstrate gap completion on real data despite only training on synthetic.

The rest of the paper is organized as follows: Section 2 discusses related work, while Section 3 details the proposed method. Section 4 show experimental results and offer discussion. Finally, Section 5 concludes the paper.

2 Related Work

2.1 Segmentation of thin-like structures

Dedicated methods to the delineation and graph extraction of neurons are plentiful (and their analysis is beyond the scope of this paper). Noteworthy, in [21] the authors used the *Discrete Morse Theory* to inherently model tree-like structures. As a more general framework, in [19], they used a patch-based approach that jointly segments and corrects for gaps in images of the retina, neurons, or roads (from satellites). Others inspired by recent innovations in deep learning and convolutional methods, either use convolutional descriptors to establish a manifold where refinement occurs or propose directly refinement approaches that aim to make object boundaries more refined and crisp [22]. More recently, approaches that merge boundary segmentation with a graph network approach to vessel extraction (in retina imaging) have been proposed [18].

2.2 Inpainting

Another complementary line of work arises in the area of inpainting [3], where one is interested in fixing gaps introduced in the scene, e.g., gaps introduced by removing the lettering of words superimposed on pictures. We take particular inspiration in this area and specifically by recent papers that use convolutional encoder-decoder architectures. While the traditional patch-based methods can synthesize high-quality patch completion by exploiting the image statistics, they are highly expensive, especially when handling the root images from mesocosms, which are of large sizes and whose missing regions contain numerous pixels. Barnes et al. [1] outperformed the traditional patch-based methods, by approximating the nearest-neighbor field, resulting in a faster algorithm. Darabi et al. [9] improved the PatchMatch [1] method by incorporating image gradients. However, these approaches need the missing regions as input. It is extremely difficult to know all the positions and shape of the gaps in a corrupted root image. Therefore, in our case, we cannot specify missing regions as input. Furthermore, these methods are not effective when the missing regions have complex structures, e.g., root branches, because the usage of low-level features constrains these methods to understand the semantics of images. Inspired by feature learning, Context Encoder (CE)

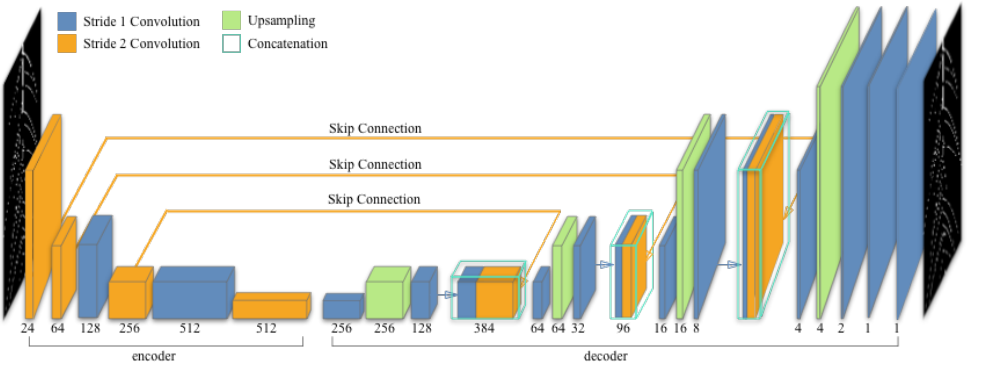

Figure 2: Overview of our architecture, consisting of a fully convolutional encoder-decoder network. As in U-Net [16], we added skip connections between the encoder and decoder.

[15] consists of an encoder that embeds the context of a central corrupted image into a latent feature space and a decoder that uses the feature map to inpaint missing image content. Iizuka et al. [7] developed the CE using a global context discriminator and a local context discriminator. The limitation of CE is that it can only handle fixed-size images, due to the presence of a fully connected layer. Yet the images of roots obtained from mesocosms can be of various sizes and the scale of the root images may result in thinner root branches which may hurt traits measurement. Also, both methods require an indication of missing regions as input. Sasaki et al. [17] overcame some of these limitations with a fully convolutional completion network that automatically detects and inpaints corrupted line drawings. This work mainly inspires our approach.

3 Proposed Method

We take inspiration from similar domain problems in image completion and namely that of inpainting of line drawings [17]. Our approach is simple, as shown in Figure 2, and trained in a fully-supervised fashion. Given images with gaps in the roots, we train the network to recover missing root segments. To train it, we use synthetic images that are initially complete, and we introduce gaps at random locations. Our model follows a classical encoder-decoder architecture with a bottleneck [20]. As depth increases, the encoder is tasked to 'compress' information (due to the bottleneck) and as such to find more global information. Then, the task of the decoder is to reconstruct the image. Such a 'compression' is what allows the network to learn an abstract representation of roots that is immune to the presence of gaps. However, since at times due to downsampling (in the encoder), inputs can be aliased, we introduce skip connections. The information from the encoder's feature maps is shared with the decoder, and then the decoder uses this information to reconstruct roots with missing parts.

In the following sections, we detail our model and its design, the losses we use, and how we train it along with the data we used.

3.1 Model Architecture

Sasaki et al. [17] used a fully convolutional network, which consists of only convolution and upsampling layers, to automatically identify the gaps between line drawings and inpaint realistic line drawings to reconnect the gaps. When seen locally, small patches of roots can resemble line drawings. Therefore, our network inherits the fully convolutional layers of the model in [17]. These layers also equip the proposed network with the ability to cope with different plant root dimensions, without the parameter constraints of fully-connected layers. Though plant roots and line drawings have similar characteristics, the root images obtained with digital cameras may have inherently complicated and irregular gap patterns. To make the proposed model robust to these random patterns, we extend their network adding skip connections between encoder and decoder. These skip connections permit the decoder to localize gaps better and yield more precise output [5, 16]. We use the network to restore the disconnected primary root and root branches, sometimes even reconnecting the detached root branches to the primary root.

The kernel size of the first convolution layer is 5, and for the other convolution layer, we use a kernel size of 3. Zero-padding is applied for all convolutions to avoid image shrinking. To 'compress' information from larger fields of view, we use 4 convolution layers of a stride of 2 to transform the image to lower dimension space, in our case, $(\frac{1}{16})$ of the original input size. This downsampling also allows the model to extract complex features and structure from root images. Sasaki et al. [17] found that upsampling layer outperforms deconvolution layer, which is commonly used in increasing spatial size. Thus, we also use 4 upsampling layers in the decoder to restore the output to the original size as the input.

Combining the idea of U-Net [16], we connect the downsampling layers and the upsampling layers. Since zero-padding at each convolution step preserves the boundary pixels, we concatenate the same size feature maps rather than cropping from the bigger ones as in U-Net. Hence, the decoder is able to utilize the information from the encoder feature maps. This cooperation between encoder and decoder should help the model recognize the complicated low-level structure of roots. Furthermore, we observed that these connections also enabled a faster and more stable training process.

In addition, we use batch-normalization [8] after each convolution layer except for the last one to further speed up the training process. We adopt the Rectified Linear Unit (ReLU) activation for all convolution layers except for the last one, where we use a sigmoid activation instead to produce the binary output.

3.2 Dataset and Training

Dataset. It is challenging to obtain ground-truth labels of real imaging data without compromise in data fidelity or data number. One could ask experts to inpaint root images that have gaps. However, even experts find difficulty in telling apart roots that should (or not) be inpainted. Since the soil occludes roots, which inherently are 3D objects, expert errors are expected to be high leading to poor data fidelity. Ideally, one after taking an image, could remove the glass from the mesocosms, spray the visible roots with colored paint, and then wash the soil and aim to interactively identify in a digital image, prior to removing the glass, the correct gaps to be linked. Naturally, this process is slow and destructive and the number of training data obtained would be limited. Instead here inspired by a seminal study that involved several experts [11] we use artificially generated data. Specifically, in this study, an artificial dataset of images was generated to evaluate the accuracy of several image process-

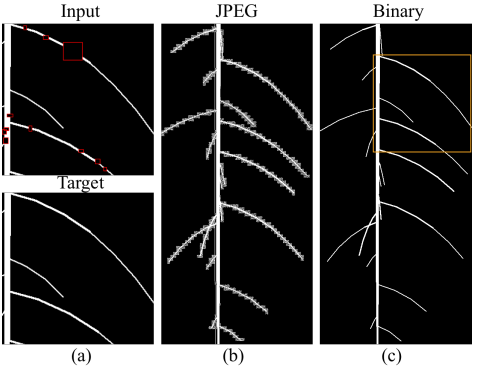

Figure 3: Example pairs and pre-processing: (a) Example of an input (top) - output (bottom) pair. (b) JPEG compression artifacts around roots. (c) The binarized image with an orange box indicating the patch extracted.

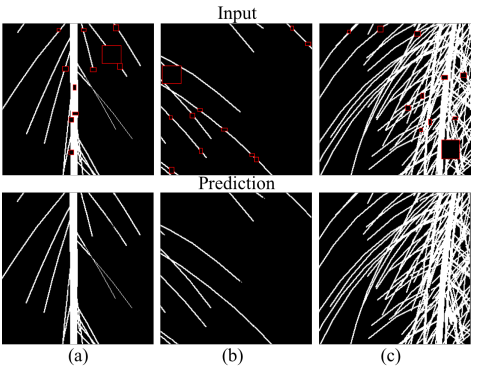

Figure 4: Examples of gap completion on synthetic (test) data. Gaps in the inputs are marked by red rectangles (top). The proposed model can detect these regions automatically and inpaint them (bottom).

ing pipelines on extracting root system architecture (RSA) traits in the presence of noise. Herein, we peruse the same data (available in [10]) for training and evaluating our approach.

Preprocessing and Data Preparation. There are approximately 5 million trainable parameters in our network. However, the synthetic roots dataset we used contains only 10,000 synthetic root images: 5,000 for dicot and 5,000 for monocot. To train the network, we extract patches from images and generate input and output pairs on-the-fly, because the full root images in the dataset have various sizes. Whilst our network can deal with different input sizes, we find that training with batches, whose images are of different sizes, may lead to memory issues and cause to adaptively have to change the batch image size to fit in the GPU memory. Instead, we use large patch training: we take a patch from an image in a batch, and then crop or pad the patches to the same size (256×256). To generate labeled data from the synthetic datasets, we randomly remove the inside of 5 to 10 squares for a given patch, each of size from 5×5 to 15×15. To imitate the often appearing large discontinuity present in real root examples, we additionally introduce a larger gap for each patch, with size randomly between 25×25 to 35×35. These artificially corrupted patches form our input and their complete version form the output as shown in Figure 3(a). Since the original images are in JPEG format, additional preprocessing was necessary due to compression artifacts present around roots, as shown in Figure 3(b). We observed that a small introduced gap may still leave such artifacts intact and affect performance [12]. We remove these artifacts by applying a threshold operation on the root images, retaining pixel values larger than 127, as shown in Figure 3(c). Finally, we normalize all inputs in the interval $[0, 1]$.

Training details and parameters. Our model is implemented in Keras [2] with TensorFlow backend. For training, we mixed 3,500 dicot and 3,500 monocot images from the synthetic root dataset. We find that a training dataset containing both dicot and monocot helps the model to achieve better performance. The comparison of results between training on mixed data and training separately on only dicot and only monocot data are shown in the supplementary material. (The supplemental is available on http://tsaftaris.com). We use a batch size of 4, and for the loss of the model, we use the mean absolute error (MAE) instead of the classic mean squared error (MSE) reconstruction loss. Although MSE penalizes the dif-

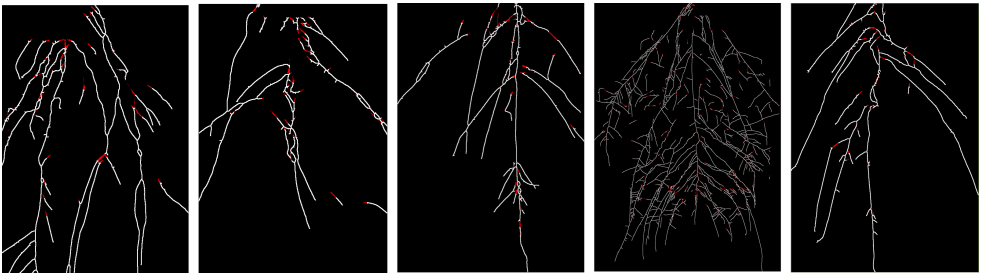

Figure 5: Examples from real images: gap correction in chickpea roots from mesocosms. Red color indicates inpainted pixels.

ference with squared errors, which is useful when large errors are undesirable, it maximizes the Gaussian distribution even the underlying truth distribution is a multimodal distribution. This means that when there are several possible outputs, the MSE loss may force the model to average together these potential results: a blurry root image in our case. Pathak et al. [15] combined MSE reconstruction loss with the adversarial loss [6] to pick a particular mode from the distribution. We simply use MAE loss to alleviate the behavior of averaging of multiple modes when using MSE loss for prediction. We use the 'AdaDelta' [23] optimizer to learn the network weights. The model is trained for nearly 45 epochs on a single NVIDIA TITAN Xp GPU, and takes roughly 10 hours. However, at test time, inference (forward pass) on a (256×256) patch takes only 120 milliseconds.

4 Results and Discussion

4.1 Evaluation process and metrics

We evaluate our proposed model qualitatively and quantitatively on exemplary synthetic full-sized root images and (256×256) patches with fixed gaps extracted from synthetic root images in our whole testing set. We also test our model on exemplary real root data that has never appeared in our training set. We compute the MAE loss on whole patches and within square gaps only for both synthetic data and real data. The results show that our model outperforms the state-of-the-art approach [17] for the inpainting of thin-like structure tasks. To evaluate how our model performs for connecting corrupted roots, we use application-driven criteria such as the ability to extract traits related to morphological and geometrical descriptors, which characterize root shape and topology [11]. We also use image-driven proxies such as connected component number and size (pixel number) to illustrate the proposed model is able to reconnect the primary root (the largest connected component). We additionally compute the number of pixels in the root as a proxy of length. On synthetic data, we perform the above on complete roots, artificially corrupted roots and inpainted roots. We show the results for dicot and monocot separately, due to the vigorous effect of 'type grouping' [11].

4.2 Qualitative Results on Synthetic Patches and Real Data

We show some exemplary patch completion results in Figure 4 on synthetic data. Our model can find, and connect, discontinuous segments of roots without the need of a mask spec-

		MAE within the gaps		MAE on full patch	
Category	Method	10 small gaps	2 large gaps	10 small gaps	2 large gaps
Syn. Dicot	[17]	.0980 (.1568)	.0397 (.0514)	.0027 (.0047)	.0030 (.0048)
	Ours	.0457 (.0833)*	.0322 (.0377)*	.0007 (.0040)*	.0012 (.0041)*
Syn. Monocot	[17]	.1229 (.1299)	.1009 (.0919)	.0184 (.0243)	.0196 (.0250)
	Ours	.1079 (.1101)*	.0886 (.0826)*	.0012 (.0009)*	.0027 (.0024)*
Real Chickpea	[17]	.1606 (.1451)	.0949 (.0639)	.0045 (.0034)	.0053 (.0038)
	Ours	.1568 (.1459)	.0928 (.0680)	.0032 (.0027)*	.0041 (.0031)*

*p-value < 0.001 of a paired t-test comparing proposed against [17].

Table 1: Comparison of results for the *mean (std)* of MAE loss on patches fro synthetic data and real data. The 10 small gaps are of random size from 5×5 to 10×10; the 2 large gaps are of random size from 25×25 to 35×35.

ifying these regions. Even when facing relatively intricate structure and patterns, such as intersections of segments of variable width, the model is still able to recover the gaps.

Although the proposed model is trained on synthetic root images, we wanted to evaluate the application in real images of chickpea roots (a dicot) from mesocosms. The model has never seen these images, and the synthetic data are not emulating chickpea roots. We show these results in Figure 5. Our model appears to reconnect the disconnected primary root and sometimes the lateral branches. While preliminary and lacking evaluation these results are particularly encouraging. Additional inpainting results on synthetic data and real data can be found in our supplementary material.

4.3 Quantitative Results

4.3.1 MAE Loss

We computed the MAE loss on 1,500 patches extracted from our testing set of the same size (256×256) with different degree of corruption: only corrupted by small square gaps and only corrupted by large square gaps. In addition, we did the same computation on 100 patches extracted from the real chickpea data. The MAE loss results are shown in Table 1. We compare against the model of Sasaki et al. [17] using the same patches with the same fixed gaps. For comparison, we ran a single-sided paired t-test, whose significance threshold is set at 0.05, between the MAE losses obtained from the methods. We find that our model is able to produce locally and globally better results: our approach decreases the mean of the errors and the errors range, especially a significant difference is indicated by t-test from synthetic data. We significantly reduce the MAE loss results on the full patch, because the skip connections in our model preserve the relatively low-level spatial features of an image. This mitigates the blurriness problem caused by the downsampling layers in the model of Sasaki et al. [17].

It is interesting to see whether more pixels missing reduces performance and its relationship to dicot/monocot. On synthetic data, we plotted the scatter plots comparing MAE loss within gaps vs. the total number of pixels lost due to gaps in a patch. We did this separately for dicot/monocot and for several small gaps or 2 large gaps cases. Results are shown in Fig.

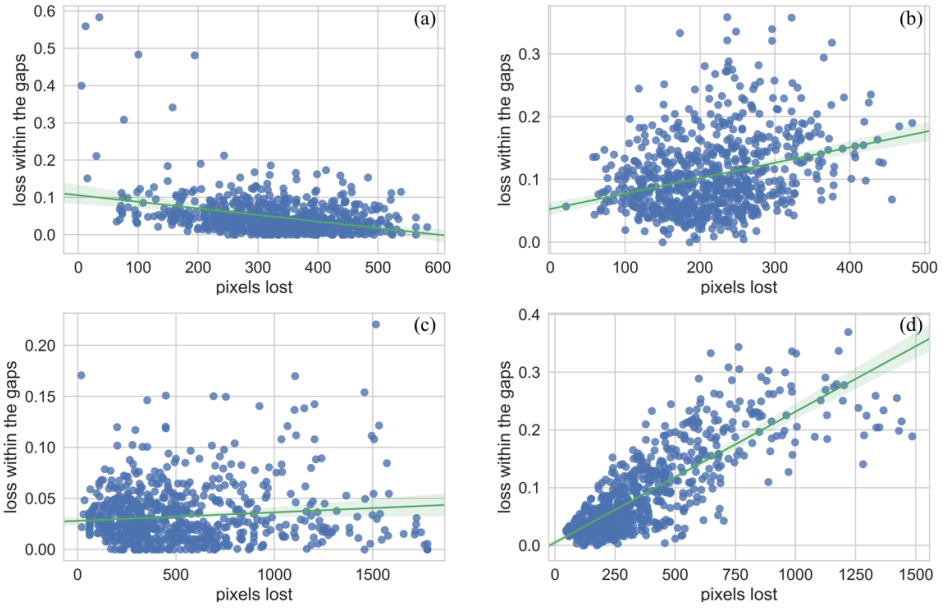

Figure 6: Plots of MAE loss on synthetic data, estimated within the square gaps, versus total number of pixels lost (within the whole patch) on testing set: (a) dicot with 10 small gaps; (b) monocot with 10 small gaps; (c) dicot with 2 large gaps; and (d) monocot with 2 large gaps. In addition, linear regression lines are plotted to show trends.

6. For dicot, one can see lack of relationship: relatively constant behaviour as the first order regression line shows, both in small (Fig. 6(a)) and big gaps (Fig. 6(c)). Moreover, it is possible to achieve perfect recovery (we see values near zero). For monocots though, a linear relationship with the spread increasing in both small and big gaps (see Fig. 6(b) and (d)) as the number of pixels lost increases perfect recovery is not possible. One possible reason for this could be losses due to aliasing (due to downsampling) in densely packed thin-like textures.

4.3.2 Phenotyping Traits and Proxy Metrics

The presence of gaps may lead to the isolation of root branches from the primary root or may break the primary root in segments. These discontinuities cast difficulty on extracting fine-grained RSA traits. To evaluate how the proposed model improves these, we compute phenotyping traits using Root Image Analysis-J (RIA-J) [9] for the corrupted full roots, complete full roots, and the inpainted full roots from our model to illustrate that the proposed model has the ability to repair some of these traits. We use the morphological descriptors such as the total root length and the number of visible root tips [11]. Additionally, we use the connected components in graph theory and the size of the largest fully connected component to estimate how closely we have recovered a single largest object (as in our ground truth) and also the total root (foreground) pixels for reference.

We performed this analysis on a representative set of synthetic root images with fixed gaps from our testing set, which contains 30 dicot and 30 monocot full root images. We

	dicot		monocot	
Metrics	Corrupted Diff.	Inpainted Diff.	Corrupted Diff.	Inpainted Diff.
Length (*mm*)	5.71 (3.39)	1.61 (2.59)*	9.82 (3.28)	4.13 (4.17)*
Tip Counts	33.13 (10.27)	6.57 (9.01)*	59.73 (19.49)	31.57 (43.91)*
Connected Comp. Num.	17.37 (7.36)	9.30 (11.07)*	13.17 (7.06)	12.93 (8.81)
Size of Primary Comp.	16,399(12,507)	627 (609)*	7,669 (4,863)	1,102 (1,317)*
Total Pixel Number	2,592 (1,416)	116 (101)*	2,016 (731)	368 (253)*

*p-value < 0.001 of a paired t-test comparing the corrupted difference and inpainted difference.

Table 2: Evaluation of phenotyping traits and proxy metrics on synthetic dicot and monocot.

computed the difference of these metrics between corrupted full roots and complete full roots, inpainted full roots and complete full roots. Additionally, we computed a single-sided paired t-test between the differences, whose significance threshold is 0.05. Results are shown in Table 2. Note the significant improvement on the accuracy of phenotyping traits of the inpainted root, compared to the corrupted root. Also, the model can connect root segments and better reconstruct the primary root. Due to the lack of ground truth of real data, it is difficult to do the same experiments on real data.

5 Conclusions and Future Work

Our approach provides an effective way of reconstructing the missing regions resulted from poor segmentation of root structures. Our model is easy to train and is robust to complicated missing patterns. Furthermore, we provide a baseline for the root gap inpainting tasks.

The ability to correct for root gaps is an interesting problem particularly for methods that adopt affordable means in developing the mesocosms and in imaging (as previously done for shoot, e.g., https://phenotiki.com as presented in [13])

Although our model produces remarkable results, this line of research is in its infancy. Due to massive downsampling, our model sometimes ignores the small gaps that appear in the thin root structures. In addition, semantic ambiguity is present as when bigger gaps occlude an area with complex structure, the model is unable to predict the content for that region and leaves it as a gap. These limitations may be relieved by a more complex model architecture or by using dilated convolution layers which can capture information from larger regions [7]. It is possible that additional losses that indirectly (e.g., adversarial loss [8]) or directly emulate phenotyping traits in the cost function will improve results. Creating a synthetic dataset that better matches the characteristics of the real data should also improve results. Finally, it is possible to optimize root segmentation and inpainting jointly.

Acknowledgement

This work was supported by the BBSRC grant BB/P023487/1. We also thank NVIDIA Inc. for providing us with a Titan Xp GPU used for our experiments.

References

[1] Connelly Barnes, Eli Shechtman, Adam Finkelstein, and Dan B Goldman. Patchmatch: A randomized correspondence algorithm for structural image editing. *ACM Transactions on Graphics-TOG*, 28(3):24, 2009.

[2] François Chollet et al. Keras. https://keras.io, 2015.

[3] Antonio Criminisi, Patrick Pérez, and Kentaro Toyama. Region filling and object removal by exemplar-based image inpainting. *IEEE Transactions on image processing*, 13(9):1200–1212, 2004.

[4] Soheil Darabi, Eli Shechtman, Connelly Barnes, Dan B Goldman, and Pradeep Sen. Image melding: Combining inconsistent images using patch-based synthesis. *ACM Trans. Graph.*, 31(4):82–1, 2012.

[5] Michal Drozdzal, Eugene Vorontsov, Gabriel Chartrand, Samuel Kadoury, and Chris Pal. The importance of skip connections in biomedical image segmentation. In Gustavo Carneiro, Diana Mateus, Loïc Peter, Andrew Bradley, João Manuel R. S. Tavares, Vasileios Belagiannis, João Paulo Papa, Jacinto C. Nascimento, Marco Loog, Zhi Lu, Jaime S. Cardoso, and Julien Cornebise, editors, *Deep Learning and Data Labeling for Medical Applications*, pages 179–187, Cham, 2016. Springer International Publishing. ISBN 978-3-319-46976-8.

[6] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Advances in neural information processing systems*, pages 2672–2680, 2014.

[7] Satoshi Iizuka, Edgar Simo-Serra, and Hiroshi Ishikawa. Globally and locally consistent image completion. *ACM Transactions on Graphics (TOG)*, 36(4):107, 2017.

[8] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *arXiv preprint arXiv:1502.03167*, 2015.

[9] Guillaume Lobet. guillaumelobet/Root-Image-Analysis-Pipeline- Evaluation: Release 2.0, December 2016. URL https://doi.org/10.5281/zenodo.208499.

[10] Guillaume Lobet, Koevoets Iko T, Pierre Tocquin, Loïc Pagès, and Claire Périlleux. Library of simulated root images, September 2016. URL https://doi.org/10.5281/zenodo.61739.

[11] Guillaume Lobet, Iko T Koevoets, Manuel Noll, Patrick E Meyer, Pierre Tocquin, Claire Périlleux, et al. Using a structural root system model to evaluate and improve the accuracy of root image analysis pipelines. *Frontiers in plant science*, 8:447, 2017.

[12] Massimo Minervini, Hanno Scharr, and Sotirios Tsaftaris. The significance of image compression in plant phenotyping applications. *Functional Plant Biology*, 42:971–988, 2015.

[13] Massimo Minervini, Mario V Giuffrida, Pierdomenico Perata, and Sotirios A Tsaftaris. Phenotiki: an open software and hardware platform for affordable and easy image-based phenotyping of rosette-shaped plants. *The Plant Journal*, 90(1):204–216, 2017.

[14] Kerstin A. Nagel, Alexander Putz, Frank Gilmer, Kathrin Heinz, Andreas Fischbach, Johannes Pfeifer, Marc Faget, Stephan Blossfeld, Michaela Ernst, Chryssa Dimaki, Bernd Kastenholz, Ann-Katrin Kleinert, Anna Galinski, Hanno Scharr, Fabio Fiorani, and Ulrich Schurr. Growscreen-rhizo is a novel phenotyping robot enabling simultaneous measurements of root and shoot growth for plants grown in soil-filled rhizotrons. *Functional Plant Biology*, 39(11):891–904, 2012.

[15] Deepak Pathak, Philipp Krahenbuhl, Jeff Donahue, Trevor Darrell, and Alexei A Efros. Context encoders: Feature learning by inpainting. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2536–2544, 2016.

[16] 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.

[17] Kazuma Sasaki, Satoshi Iizuka, Edgar Simo-Serra, and Hiroshi Ishikawa. Joint gap detection and inpainting of line drawings. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5768–5776, 2017.

[18] Seung Yeon Shin, Soochahn Lee, Il Dong Yun, and Kyoung Mu Lee. Deep vessel segmentation by learning graphical connectivity. *arXiv preprint arXiv:1806.02279*, 2018.

[19] Amos Sironi, Vincent Lepetit, and Pascal Fua. Projection onto the manifold of elongated structures for accurate extraction. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 316–324, 2015.

[20] Naftali Tishby and Noga Zaslavsky. Deep learning and the information bottleneck principle. In *Information Theory Workshop (ITW), 2015 IEEE*, pages 1–5. IEEE, 2015.

[21] Suyi Wang, Xu Li, Partha Mitra, and Yusu Wang. Topological skeletonization and tree-summarization of neurons using discrete morse theory. *arXiv preprint arXiv:1805.04997*, 2018.

[22] Yupei Wang, Xin Zhao, and Kaiqi Huang. Deep crisp boundaries. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3892–3900, 2017.

[23] Matthew D Zeiler. Adadelta: an adaptive learning rate method. *arXiv preprint arXiv:1212.5701*, 2012.