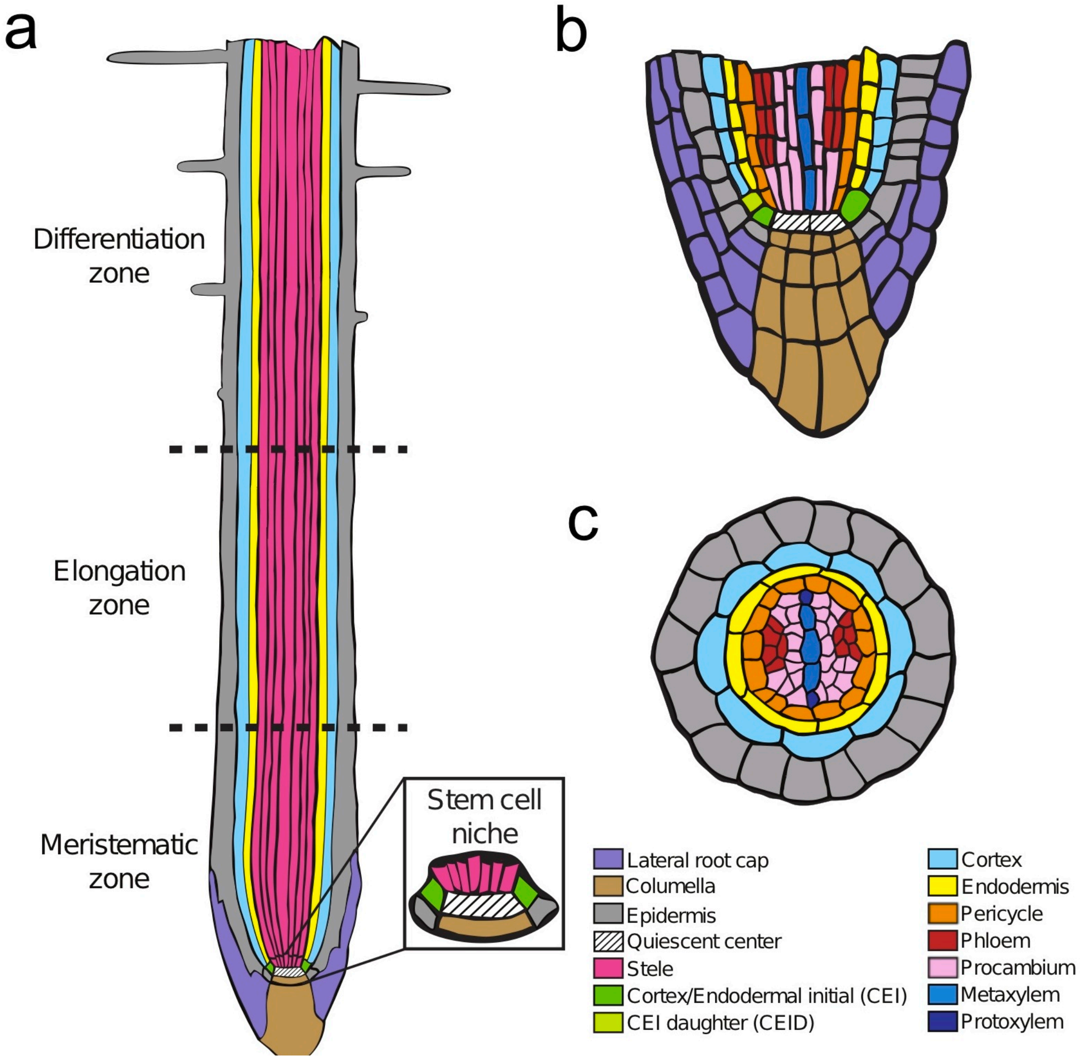
A microfluidic device and computational platform for high throughput live imaging of gene expression

Wolfgang Busch, Bradley Moore, Bradley Martsberger, Daniel L. Mace, Richard W. Twigg, Jee Jung, Iulian Pruteanu-Malinici, Scott J. Kennedy, Gregory K. Fricke, Robert L. Clark, Uwe Ohler, Philip N. Benfey

Supplementary Figure 1: **Root Anatomy based extraction of expression information**

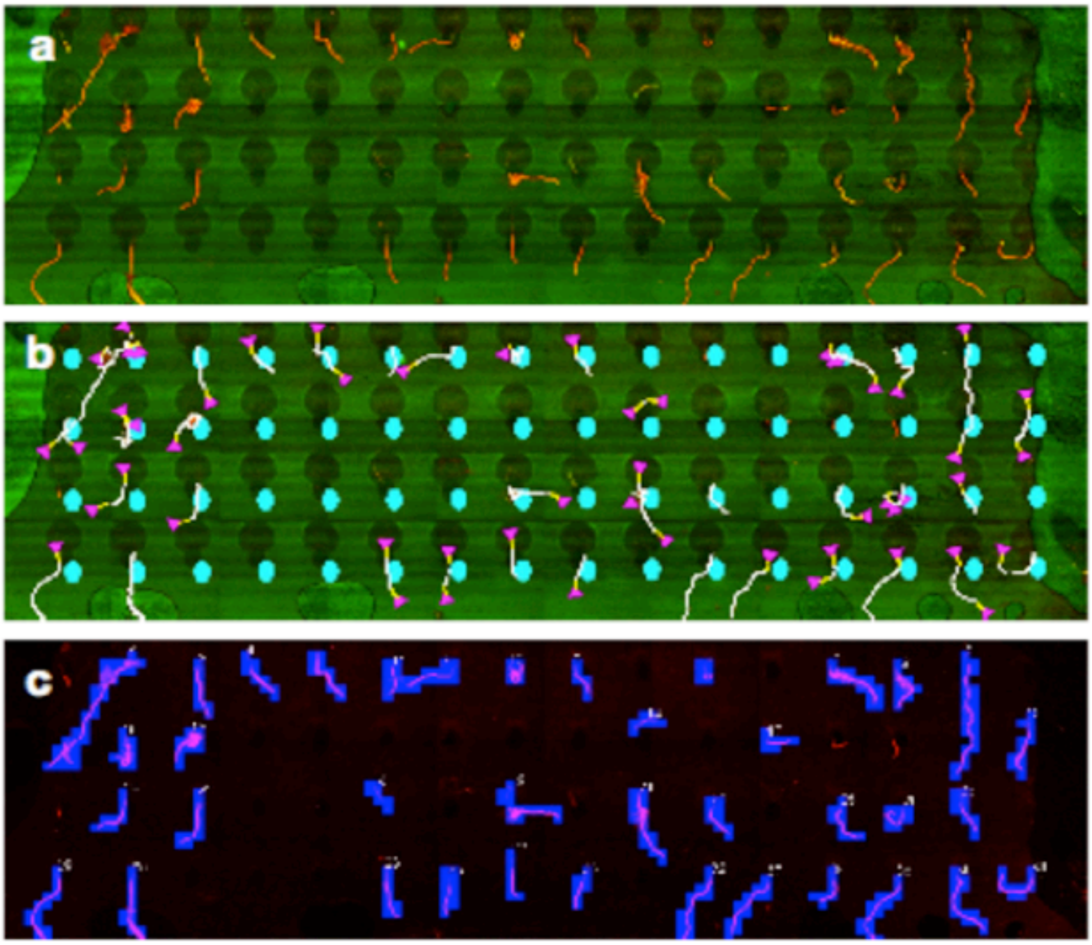


Supplementary Figure 1: Root Anatomy based extraction of expression information. (a) Side view of an Arabidopsis root. The different developmental zones are indicated. (b) A detailed view of the tip of the root. (c) A cross-section of the root. (Figures a-c are adapted from [1])

References

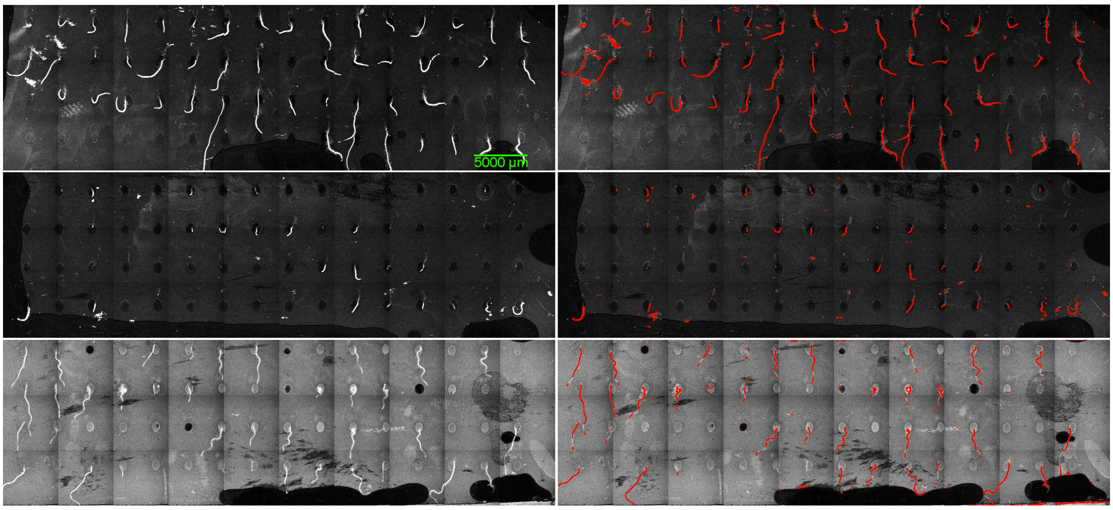
1. Van Norman, J.M., Breakfield, N.W. & Benfey, P.N. Intercellular communication during plant development. *Plant Cell* **23**, 855-864 (2011).

Supplementary Figure 2: **Low resolution imaging**



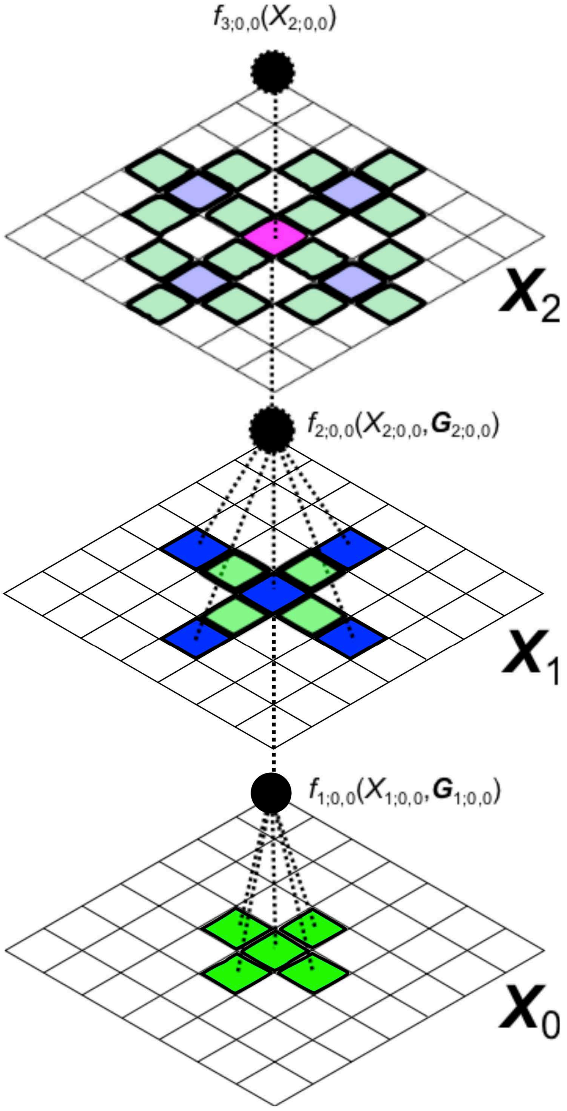
Supplementary Figure 2: Low resolution imaging. (a) A low resolution image of a complete RootArray. The red channel depicts the staining with propidium iodide, and the green channel reflects the autofluorescence of the RootArray. (b) Detected wells are marked in cyan, roots are marked in white, and the meristem regions are marked in magenta. (c) Blue squares depict the locations of high resolution images scheduled to be acquired.

Supplementary Figure 3: **Image Variability**



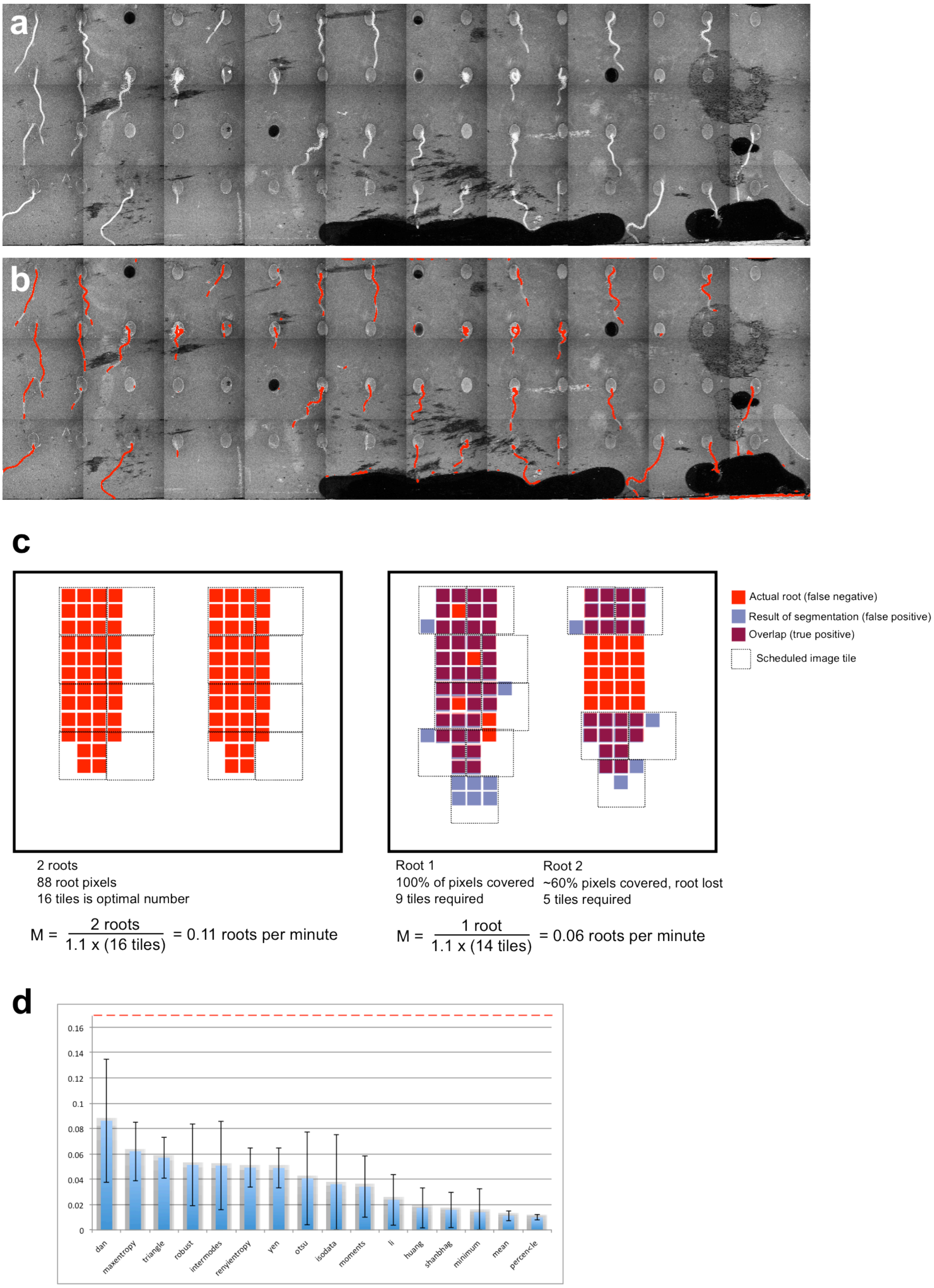
Supplementary Figure 3: Image Variability. Three samples of low resolution images of three different RootArrays; these images represent the variability in image quality and contrast. The right column shows the PI channel of these images; the left column shows the same images with the results of our factor graph segmentation highlighted in red.

Supplementary Figure 4: **Factor graph for image segmentation**



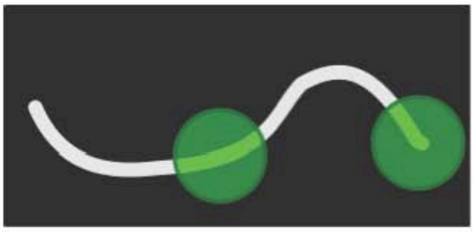
Supplementary Figure 4: Factor graph for image segmentation. The Supplementary Figure hows a model with 3 levels. Each grid represents random variables at different levels, the black circles represent functions on random variables. Dashed lines mark the relationship between functions and the variables they are dependent on. At higher levels, the Supplementary Figure hows (in a lighter color) the previous variables that are indirectly associated with the center variable. Higher levels incorporate information from broader spatial areas.

Supplementary Figure 5: **Comparative evaluation of the segmentation algorithm**



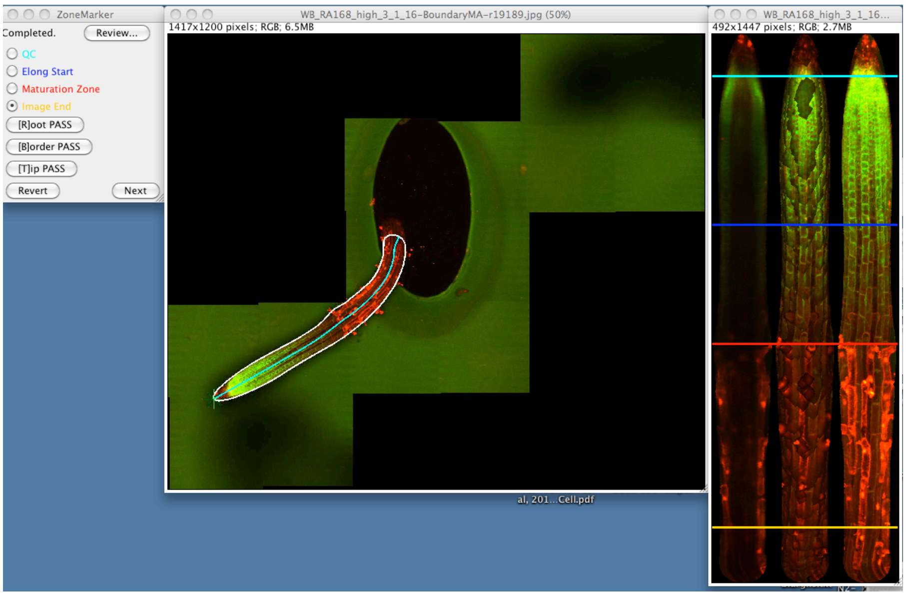
Supplementary Figure 5: Comparative evaluation of the segmentation algorithm. (a) The original low resolution PI channel image. (b) The result of the hierarchical model-based segmentation; the root labeled pixels are in red and are superimposed on the original image. (c) The roots per minute performance metric. The left panel shows two perfectly labeled roots and the subsequent tiling. The right panel shows the results of an imperfect segmentation. More than 90% of the rightmost root is missed, and therefore only one root is in the numerator. Noise at the tip of the leftmost root results in an extraneous tile being acquired. (d) Standard-deviation error bars showing the performance increase of the hierarchical model in roots-per-minute compared to 15 different segmentation algorithms previously implemented in Fiji. The dashed red line shows the mean result of hand-labeling each image (i.e. an optimal roots per minute). The optimal roots per minute reflects the limits of the microscope in terms of acquisition time and the typical size of the roots.

Supplementary Figure 6: **Topological tip detection algorithm**



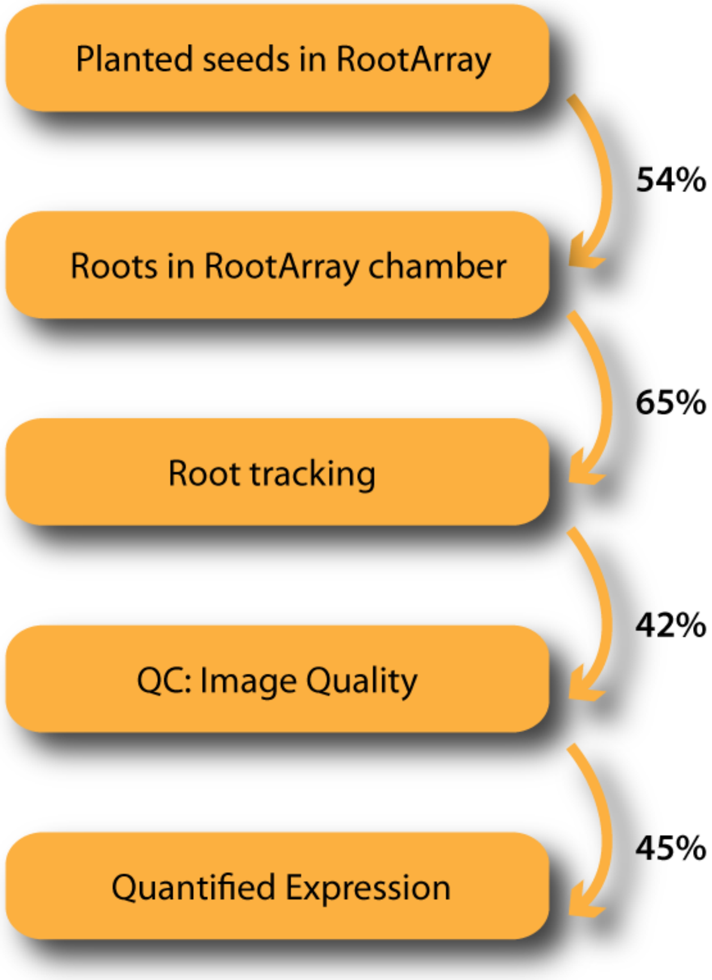
Supplementary Figure 6: Topological tip detection algorithm. Illustrative example of the topological tip detection algorithm. At each pixel of the identified root (in white), a circle (in green) is centered. The entire root image is subtracted from the circle, and the pixel is labeled with the number of components the circle was sectioned into. The left circle will be cleaved into 2 components; the right circle remains as 1 component, and therefore the pixel is identified as a tip.

Supplementary Figure 7: **Graphical User Interface for RootArray data analysis**



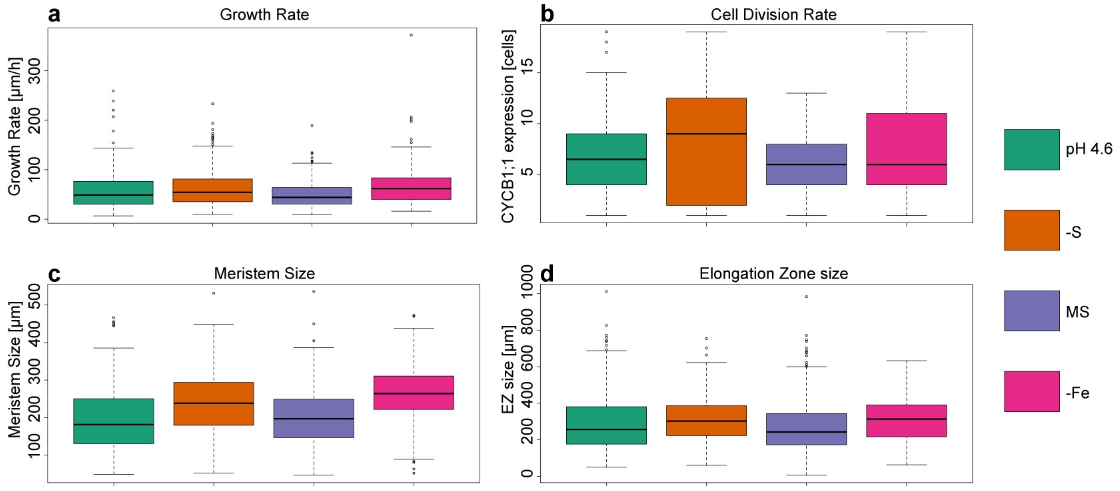
Supplementary Figure 7: Graphical User Interface for RootArray data analysis. Zonemarker screenshot. The manual zone marking and quality control software is implemented as a Fiji plugin. The leftmost picture shows an unstraightened maximum projection of a root with the boundary outlined in white and the medial axis in cyan. The rightmost image shows the computationally straightened root in three views (from left to right): medial section, surface, and max projection. The four horizontal lines are zones manually marked by the user: the quiescent center (cyan), the start of elongation (blue), the start of maturation (red), and a vertical crop location (yellow).

Supplementary Figure 8: **Step-wise performance evaluation of the RootArray pipeline**



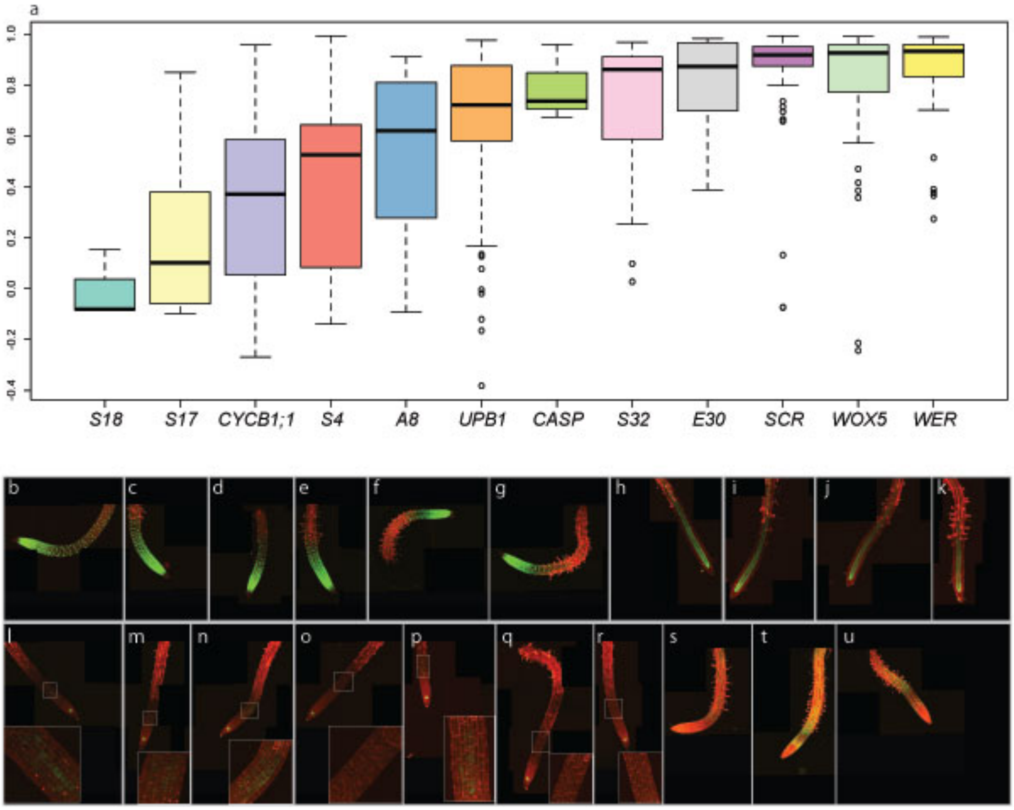
Supplementary Figure 8: Step-wise performance evaluation of the RootArray pipeline. The numbers denote the average fraction of successfully recovered roots compared to total number of roots in the previous step.

Supplementary Figure 9: **Effects of growth conditions on growth rate**



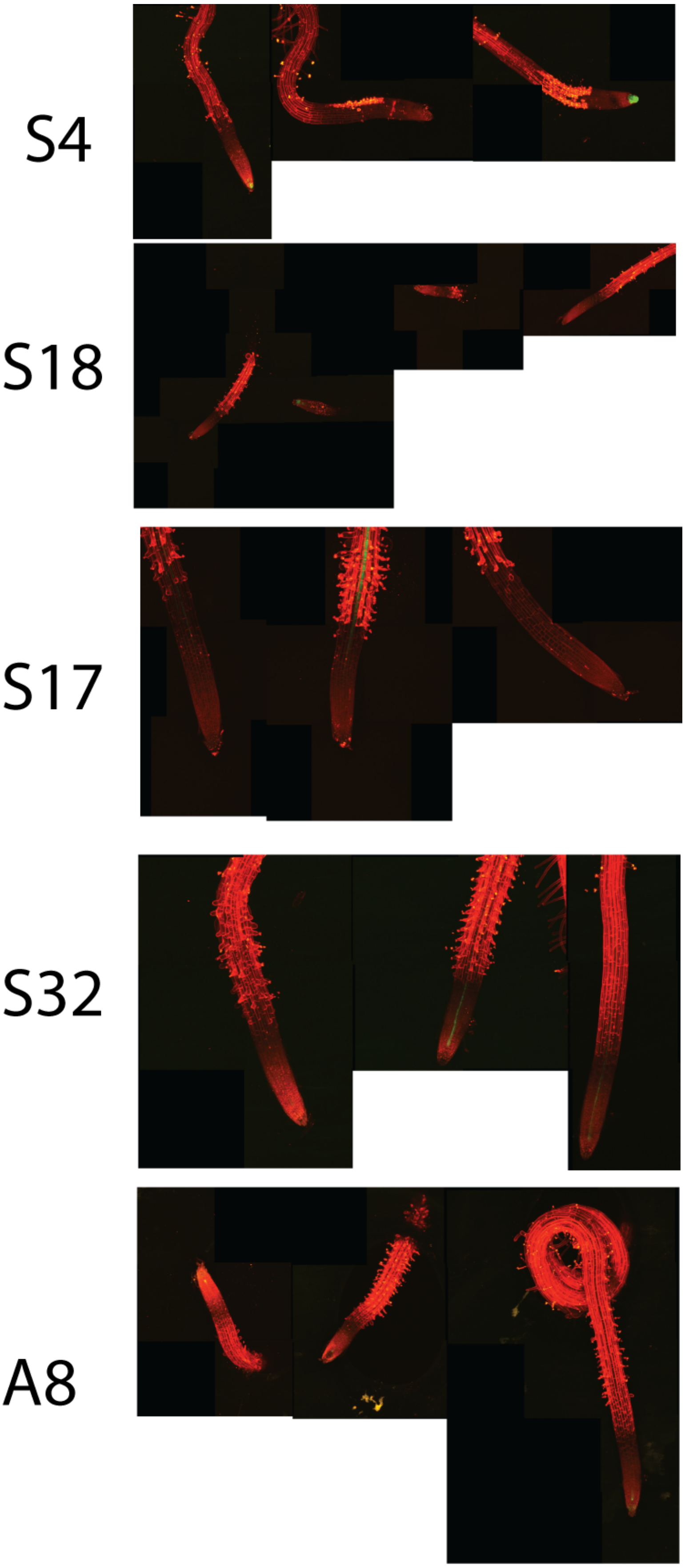
Supplementary Figure 9: Effects of growth conditions on growth rate (a), cell cycle progression (b), meristem size (c) and elongation zone size (d). Data is from the first 24h of treatments.

Supplementary Figure 10: **Heterogeneity Quantification**



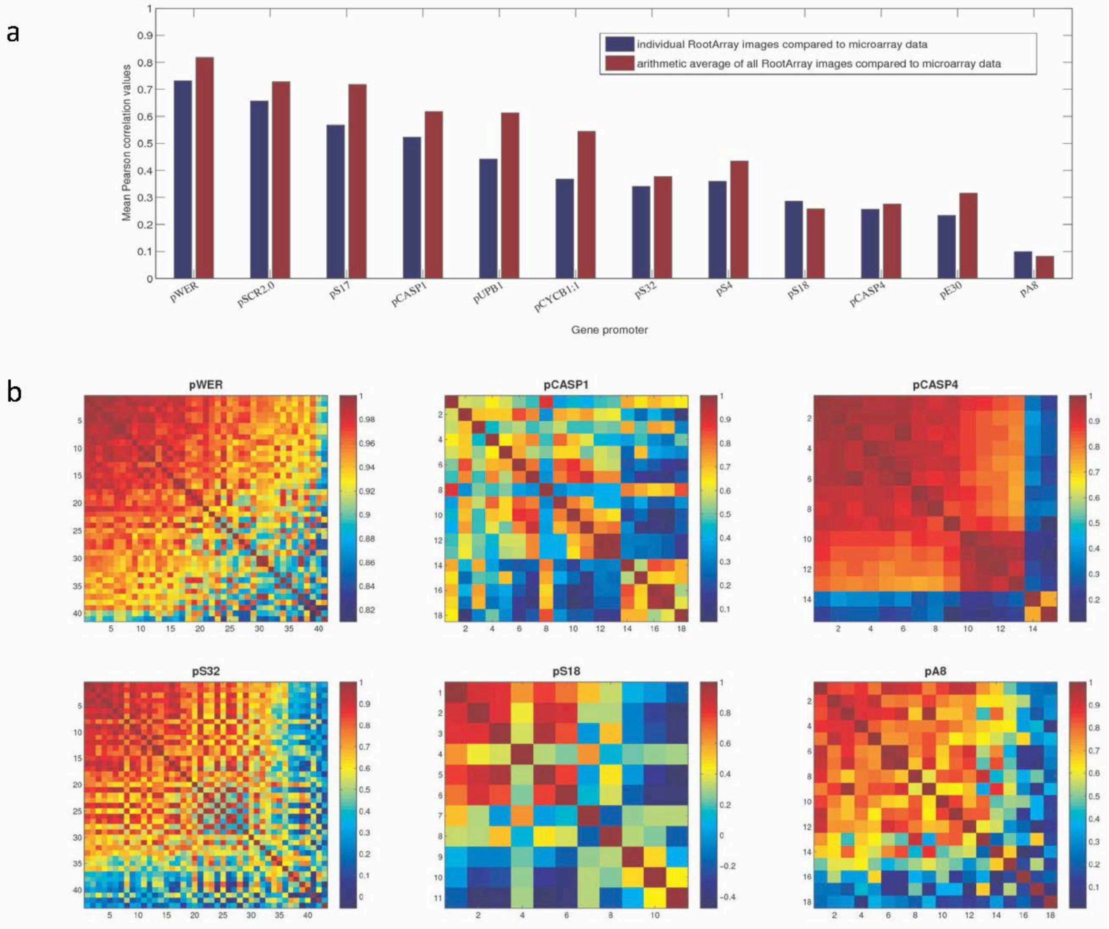
Supplementary Figure 10: Heterogeneity Quantification. Heterogeneity of gene expression between isogenic plants imaged on the same RootArray at the same time point. (a) Boxplot of Pearson correlation of expression in all tissue types between roots in the same RootArray at the same timepoint. High correlation (y-axis) indicates a homogeneous expression pattern, low correlation high heterogeneity. (b-k) Examples of low heterogeneity expression patterns of roots in the same array at the same time point such as *WER* (b-g) and *SCR* (h-k), and higher heterogeneity such as *WOX5 (l-r)* and *UPB1* (s-u). Insets are magnified portions of the root indicated with boxes in the main figure.

Supplementary Figure 11: **Examples for high heterogeneity of expression**



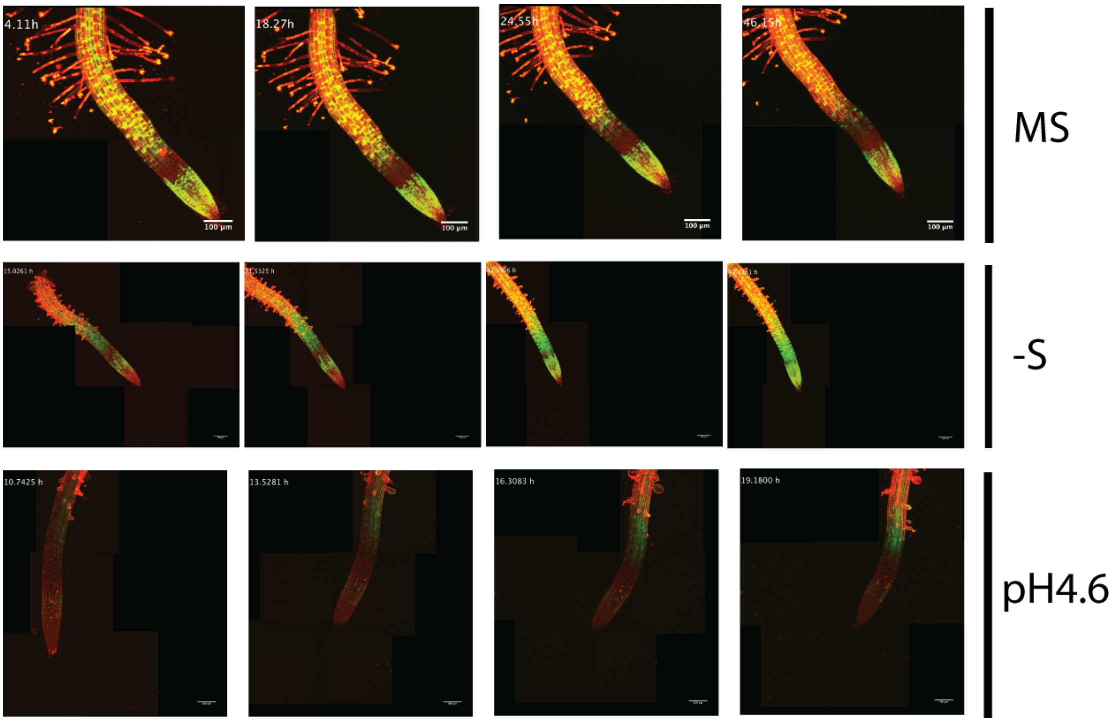
Supplementary Figure 11: Examples for high heterogeneity of expression. Examples of high heterogeneity of expression in isogenic plants imaged on the same array at the same time point.

Supplementary Figure 12: **Correlation between RootArray and RootMap**



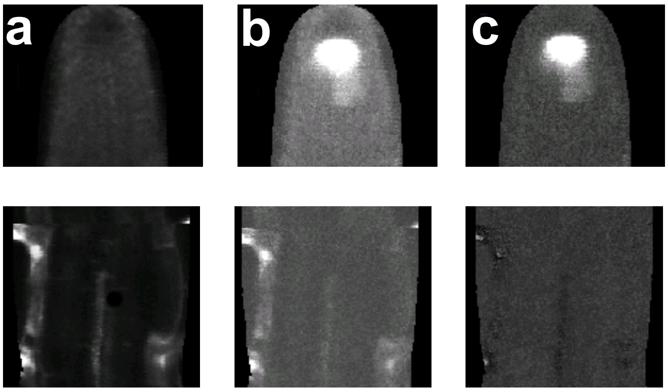
Supplementary Figure 12: Correlation between RootArray and RootMap. Pearson correlation values between RootArray and RootMap, as well as heterogeneity of spatial gene expression profiles. (a) An 18-dimensional expression vector for different tissue types and developmental zones was extracted from each RootArray image, or from the arithmetic average of all RootArray images for a given genotype, and compared to the corresponding FAC-sorted microarray gene expression data from the RootMap for the respective gene promoter. (b) Pair-wise Spearman rank correlation values for root replicate images obtained from the same RootArray under standard growth conditions were computed. This identifies outliers within a set of images belonging to the same gene promoter, as well as heterogeneous/multimodal patterns.

Supplementary Figure 13: **Progression of UPB1 expression change**



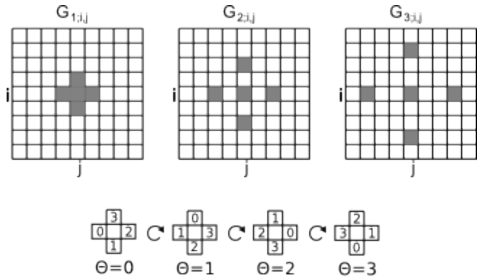
Supplementary Figure 13: Progression of UPB1 expression change. Examples of progression of expression change in *UPB1* reporter line in different growth conditions.

Supplementary Figure 14: **Crosstalk correction**



Supplementary Figure 14: Crosstalk correction. (a) The PI channel for pWOX5 reporter. The top image shows the meristem, the bottom the mature part of the root. (b) The original GFP channel. Notice that there is unexpected signal in the epidermis. (c) The GFP channel with cross-talk removed. We see the expected signal in the quiescent center but not in the mature epidermis

Supplementary Figure 15: **Neighborhood relations in the segmentation model**



Supplementary Figure 15: Neighborhood relations in the segmentation model. (Top) Neighborhood function at three different levels. (Bottom) Schema of the rotation function for each value of Θ.

Supplementary Tables

Supplementary Table 1: **Performance of the registration algorithm**

Tissue Type	**Zone**	**Number of Points Correct**	**Number of Points Incorrect**	**Accuracy**
Columella	meristem	90	16	0.85
Cortex	meristem	60	66	0.48
Endodermis	meristem	29	99	0.23
Epidermis	meristem	62	54	0.53
LRC	meristem	61	30	0.67
Pericycle	meristem	14	58	0.19
Stem Cell Niche	meristem	97	14	0.87
Vasculature	meristem	72	78	0.48
Cortex	elongation	122	27	0.82
Endodermis	elongation	64	51	0.56
Epidermis	elongation	130	3	0.98
Pericycle	elongation	21	38	0.36
Vasculature	elongation	113	23	0.83
Cortex	mature	218	87	0.71
Endodermis	mature	91	81	0.53
Epidermis	mature	200	27	0.88
Pericycle	mature	28	38	0.42
Vasculature	mature	171	63	0.73
Mean		**1643**	**853**	**0.66**

Supplementary Table 1: Performance of the registration algorithm. The computed tissue types of 50 sample points from 50 random images were manually verified in order to evaluate the performance of the registration algorithm. Results are reported as the total number of correct points, the total number of incorrect points, and the accuracy (total correct divided by total points). Results are divided by tissue type and longitudinal zone.

Supplementary Table 2: **Reporter lines**

AGI	Gene	Construct	Reference
AT3G48100	A8	pA8::GFP	Lee et al. 2006
AT2G36100	CASP1	pCASP1::GFP	Roppolo et al., 2011
AT3G11550	CASP2	pCASP2::GFP	Roppolo et al., 2011
AT5G06200	CASP4	pCASP4::GFP	Roppolo et al., 2011
AT4G37490	CYCB1;1	pCYCB1;1::CYCB1;1-GFP	Colon-Carmona et al., 1999
AT4G21340	E30	pE30::GFP	Lee et al. 2006
AT2G22850	S17	pS17::GFP	Lee et al. 2006
AT5G12870	S18	pS18::GFP	Lee et al. 2006
AT2G18380	S32	pS32::GFP	Lee et al. 2006
AT3G25710	S4	pS4::GFP	Lee et al. 2006
AT3G54220	SCR	pSCR::GFP	Levesque et al. 2006
AT2G47270	UPB1	pUPB1::GFP	Tsukagoshi et al. 2010
AT5G14750	WER	pWER::GFP	Lee et al., 1999
AT3G11260	WOX5	pWOX5::GFP	Xu et al., 2006

Supplementary Table 2: Reporter lines. Reporter lines used in this study.

Supplementary Table 3: **Performance metrics**

Measure	**Process Type**	**Average time [h]**	**Average recovery rate**
Roots in RootArray Chamber	Physical	NA	54%
Low Resolution Scanning	Microscopy	0.12*	NA
Root Detection	Computational	0.23*	65%
High Resolution Imaging per Timepoint	Microscopy	6.75*	NA
Well mapping and Quality Control Timepoint	Expert labeling	0.27**	42%
Mapping of Tissue Pixel Intensities	Computational	0.081***	45%

Supplementary Table 3: Performance metrics. Average time needed for each step of the pipeline and root recovery rate (i.e. root data of sufficient quality recovered from the previous step). * averages for presented data set; ** maximal effort if 64 well mapping and QC decisions would be performed; *** average running time per image on random sample of 10 images

Supplementary Table 4: **Dataset overview**

Genotype	**MS**	**pH4.6**	**-S**	**-Fe**	**Total**
pA8::GFP	18	12	12	2	44
pCASP	33[*]	40[#]	49	18	140
pCYCB1;1	92	47	40	35	214
pE30::GFP	44	24	8	3	79
pS17::GFP	14	0	24	15	53
pS18::GFP	11	4	7	6	28
pS32::GFP	48	45	17	5	115
pS4::GFP	41	34	29	8	112
pSCR::GFP	40	37	26	43	146
pUPB1::GFP	78	43	32	10	163
pWER::GFP	41	31	36	30	138
pWOX5::GFP	42	27	86	6	161
Total	502	344	366	181	**1393**
# RootArrays	**21**	**20**	**17**	**15**	**73**

Supplementary Table 4: Dataset overview. Number of whole root images and RootArrays that were contained in the final dataset for analysis. ([*] 18 *pCASP1::GFP* and 15 *pCASP2::GFP*; [#] 33 *pCASP1::GFP*, 7 *pCASP2::GFP*)

Supplementary Table 5: **Test for trait change**

Trait	**-S**	**-Fe**	**pH 4.6**
Growth Rate	2.60E-07	1.34E-07	0.0723
Cell Cycle Progression	0.4631	0.693	0.2577
Meristem Size	6.64E-13	2.20E-16	0.01622
Elongation Zone Size	9.19E-12	2.14E-08	0.1374

Supplementary Table 5: Test for trait change. Statistical test results (Wilcoxon rank sum test) for differences in growth rates, cell division rates and zone sizes.

Supplementary Table 7: **Imaging dates and treatments**

EXP_ID	**Treatment**	**Imaging start (date, time)**
WB_RA168	standard MS	2009-11-06 18:18:44 EST-0500
WB_RA178	standard MS	2009-11-13 16:53:19 EST-0500
WB_RA204	standard pH4.6	2009-12-13 21:16:01 EST-0500
WB_RA230	standard MS	2010-02-19 12:53:37 EST-0500
WB_RA232	standard pH4.6	2010-02-22 09:59:42 EST-0500
WB_RA243	standard MS	2010-03-13 19:25:05 EST-0500
WB_RA246	standard pH4.6	2010-03-20 11:40:29 EDT-0400
WB_RA248	standard pH4.6	2010-03-22 09:37:01 EDT-0400
WB_RA250	standard MS	2010-04-02 11:32:54 EDT-0400
WB_RA255	standard pH4.6	2010-04-09 19:33:35 EDT-0400
WB_RA266	standard MS	2010-04-20 09:53:16 EDT-0400
WB_RA268	standard MS	2010-04-24 18:21:55 EDT-0400
WB_RA274	standard pH4.6	2010-04-30 15:16:54 EDT-0400
WB_RA278	standard pH4.6	2010-05-04 09:48:51 EDT-0400
WB_RA285	standard MS	2010-05-17 09:55:38 EDT-0400
WB_RA286	standard pH4.6	2010-05-18 17:31:09 EDT-0400
WB_RA289	standard MS	2010-05-21 17:16:11 EDT-0400
WB_RA294	standard pH4.6	2010-05-28 15:07:29 EDT-0400
WB_RA299	standard pH4.6	2010-06-01 18:14:24 EDT-0400
WB_RA310	standard MS	2010-06-14 11:12:19 EDT-0400
WB_RA345	standard -S	2010-07-27 10:35:57 EDT-0400
WB_RA350	standard -Fe	2010-07-30 16:26:40 EDT-0400
WB_RA351	standard -Fe	2010-08-02 10:29:05 EDT-0400
WB_RA355	standard -Fe	2010-08-04 10:35:20 EDT-0400
WB_RA356	standard -Fe	2010-08-06 11:10:42 EDT-0400
WB_RA359	standard -Fe	2010-08-08 18:45:03 EDT-0400
WB_RA360	standard -Fe	2010-08-10 18:54:44 EDT-0400
WB_RA364	standard -S	2010-08-14 14:38:57 EDT-0400
WB_RA366	standard -S	2010-08-16 12:04:24 EDT-0400
WB_RA367	standard -S	2010-08-18 15:15:34 EDT-0400
WB_RA370	standard -S	2010-08-20 10:02:44 EDT-0400
WB_RA372	standard -S	2010-08-24 16:14:48 EDT-0400
WB_RA373	standard -S	2010-08-22 18:33:35 EDT-0400
WB_RA379	standard -S	2010-08-29 14:32:11 EDT-0400
WB_RA387	standard MS	2010-09-07 18:57:53 EDT-0400
WB_RA389	standard MS	2010-09-09 18:14:22 EDT-0400
WB_RA391	standard MS	2010-09-11 13:42:26 EDT-0400
WB_RA395	standard MS	2010-09-15 14:36:54 EDT-0400
WB_RA396	standard MS	2010-09-17 11:51:26 EDT-0400
WB_RA397	standard MS	2010-09-18 20:32:43 EDT-0400
WB_RA398	standard MS	2010-09-20 15:49:25 EDT-0400
WB_RA406	standard MS	2010-09-27 11:11:29 EDT-0400
WB_RA407	standard MS	2010-09-28 16:23:37 EDT-0400

WB_RA410	standard pH4.6	2010-10-01 13:52:31 EDT-0400
WB_RA412	standard pH4.6	2010-10-04 13:21:12 EDT-0400
WB_RA414	standard pH4.6	2010-10-05 14:25:35 EDT-0400
WB_RA415	standard pH4.6	2010-10-07 12:25:40 EDT-0400
WB_RA418	standard pH4.6	2010-10-11 09:26:00 EDT-0400
WB_RA419	standard pH4.6	2010-10-12 09:00:24 EDT-0400
WB_RA420	standard pH4.6	2010-10-13 10:44:16 EDT-0400
WB_RA421	standard pH4.6	2010-10-14 18:23:13 EDT-0400
WB_RA425	standard pH4.6	2010-10-18 10:28:22 EDT-0400
WB_RA426	standard -S	2010-10-19 10:29:08 EDT-0400
WB_RA428	standard -S	2010-10-21 09:58:38 EDT-0400
WB_RA429	standard -S	2010-10-22 13:20:53 EDT-0400
WB_RA430	standard -S	2010-10-23 15:59:49 EDT-0400
WB_RA434	standard -S	2010-10-28 09:40:00 EDT-0400
WB_RA436	standard -S	2010-10-29 18:17:53 EDT-0400
WB_RA441	standard -S	2010-11-03 16:28:17 EDT-0400
WB_RA446	standard -Fe	2010-11-07 20:05:06 EST-0500
WB_RA454	standard -Fe	2010-11-16 10:46:40 EST-0500
WB_RA457	standard -S	2010-11-19 13:47:41 EST-0500
WB_RA461	standard -Fe	2010-11-23 10:40:03 EST-0500

Supplementary Table 7: Imaging dates and treatments.

Supplementary Note 1. **Hierarchical Model for Root Detection.**

We defined a hierarchical multi-level model (see Supplementary Fig. 4) over the $W \times H$ image $I(i,j), 0 \le i \le W, 0 \le j \le H$ as a set of random variables $\{X_{k;i,j} : 0 \le k \le n, 0 \le i \le W, 0 \le j \le H\}$, where n specifies the number of levels in the model, k is the current *level* of the random variable, and (i,j) is the corresponding pixel location. The observed variables $\mathbf{X}_0 = \{X_{0;i,j} = I(i,j) : 0 \le i < W, 0 \le j < H\}$ make up the lowest level of the model and correspond to the pixel intensities. The remaining variables $\mathbf{X}_{k;i,j} = \{X_{k;i,j} = x_{k;i,j}\}, k > 0, 0 \le x_{k;i,j} < L_k$ are the latent variables where $\mathbf{X}_n$ represents the final label (e.g. root or background) for each pixel. The range of values (defined by L_k) for a random variable is dependent on its level. Our goal was to compute the maximum a posteriori (MAP) estimate of the latent variables,

$$\arg\max_{X_k \neq 0} \frac{P(\mathbf{X}_0 \mid \mathbf{X}_{k \neq 0}) P(\mathbf{X}_{k \neq 0})}{P(\mathbf{X}_0)}$$

Intuitively, we wanted to capture both the pixel-intensity-based detail of the image at the lowest level , while specifying labeling dependencies at increasingly larger areas in higher levels. For a random variable $X_{k;i,j}$, we therefore defined its neighborhood based on its corresponding random variables at the previous level as

$$\mathbf{G}_{k;i,j} = \left\langle X_{k-1;i-2^k,j}, X_{k-1;i,j+2^k}, X_{k-1;i+2^k,j}, X_{k-1;i,j-2^k}, X_{k-1;i,j} \right\rangle$$

Note, at $k = 1$, the neighborhood function is similar to the Ising Model [1] and the variable $X_{1;i,j}$ is related to the observed pixels as shown in Supplementary Fig. 4. As k increases, the variables are connected to the lower level variables at farther distances, and are influenced by the larger regions of the image. In each level, the neighborhood consists of five elements.

The lowest latent level, $k = 1$, aims to capture the spatially significant correlations (or “texture”) of the image. We modeled this relationship as a multivariate normal over the neighborhood $\mathbf{G}_{1;i,j}$. The probability that a random variable at this level has a value $l, 0 \le l < L_1$ is

$$P(X_{1;i,j} = l \mid \mathbf{G}_{1;,i,j}) = f_{1;i,j}(l, \mathbf{G}_{1;i,j}) = N(\mathbf{G}_{1;i,j}; \mu_l, \Sigma_l)$$

However, this relationship is dependent on the rotation of the object. Observer that $\mathbf{G}_{1;i,j}$ is comprised of five elements: the observed value at (i, j), the pixel above, the pixel below, and the two pixels on either side. For a given $\mathbf{G}_{1;i,j}$ we defined the ordered tuple,

$$\mathbf{g} = \langle g_0, g_1, g_2, g_3 \rangle = \langle X_{0;i-1,j}, X_{0;i,j+1}, X_{0;i+1,j}, X_{0;i,j-1} \rangle$$

We modeled the potential rotations of this neighborhood by permuting the outer elements around the center (see Supplementary Fig. 15). We defined the rotation function $r(\theta, \mathbf{g}), 0 \le \theta \le 3$ as

$$r(\theta, \mathbf{g}) = \langle g_{0+\theta \bmod 4}, \ldots, g_{3+\theta \bmod 4} \rangle$$

This let us define a relationship over all possible permutation. Given a parameterized multivariate normal, we defined the highest scoring rotation, $\hat{\theta}$, as

$$\hat{\theta} = \arg\max_{\theta} N(X_{1;i,j}, r(\theta, \mathbf{g}); \mu, \Sigma)$$

The probability that a random variable at level $k = 1$ has a value l given its neighborhood is

$$f_{1;i,j}(l, \mathbf{G}_{1;i,j}) = N(X_{1;i,j}, r(\hat{\theta}, \mathbf{g}); \mu_l, \Sigma_l) / K \qquad \text{Eq 1}$$

where $K = \sum_{x,\mathbf{g}} N(x, r(\hat{\theta}, \mathbf{g}); \mu_l, \Sigma_l)$ is a normalizing constant.

What remained was to define the coarser segmentation label relationships at the higher level ($k > 1$), which we modeled with a multinomial distribution. We defined the counts as

$$x_m = \left|\{g : g \in \mathbf{G}_{k;i,j} \wedge g - m\}\right|, 0 \le m < L_{k-1}$$

Our relationship uses these counts, and was defined as

$$P(X_{k;i,j} = l \mid \mathbf{G}_{k;i,j}) = f_{k;i,j}(l, \mathbf{G}_{k;i,j}) = Mult(x_0,\ldots,x_{L_{k-1}};5;p_0,\ldots,p_{L_{K-1}}) \qquad \text{Eq 2}$$

where

$$Mult(x_0,\ldots,x_{L_{K-1}};5;p_0,\ldots,p_{L_{K-1}}) = \frac{5!}{x_0!\ldots x_{L_{K-1}}!} p_0^{x_0}\ldots p_{L_{K-1}}^{x_{L_{K-1}}}$$

is the multinomial distribution. Note that the probabilities $p_0 \ldots p_{L_{K-1}}$ were defined for each value of $l \in L_K$.

Finally, we defined the joint probability of the model as

$$h(\mathbf{x}_0, \mathbf{x}_{k \neq 0}) = \prod_{k=1}^{n+1} \prod_{i,j} f_{k;i,j}(x_{k;i,j}, \mathbf{G}_{k;i,j}) \qquad \text{Eq 3}$$

where $f_{n+1;i,j}$ was the prior probability for a given final label, i.e. the object classes such as root and background. In summary, this model is a generalization of typical Markov Random Field models for segmentation, in which the observable pixel intensities are dependent on a single layer of hidden labels [20], to multiple intermediate layers that represent spatial dependencies at different resolutions. Furthermore, we included rotation invariance in the model to account for the varying directions of root growth.

Computing the Root Model Marginal Probability. To determine what the final label should be for a specific $x_{n;i,j}$ we need to marginalize the joint distribution over all *values* of the other random variables. With $\sim\{X_{n;i,j}\}$ denoting all possible value assignments over variables other than $x_{n;i,j}$ in our summation, we defined the marginal probability as

$$h_{x_{n;i,j}}(\mathbf{x}_0, \mathbf{x}_{k \neq 0}) = \sum_{\sim\{X_{n;i,j}\}} \prod_{k=1}^{n+1} \prod_{i,j} f_{k;i,j}(x_{k;i,j}, \mathbf{G}_{k;i,j})$$

The summation notation does not immediately indicate the exponential number of terms in the summation, namely $\prod_{k=1}^{n} L_k^{WH}$. We reduced the complexity of this calculation by factoring out independent terms, and sped up the computation by stochastic approximation methods.

The factor graph is a graphical model for decomposable problems. It subsumes both directed and undirected graphical models on joint distributions, such as Bayesian networks and Markov random fields, within on formalism [2]. Given a function

$$h(x_1,\ldots,x_n) = \prod_{j \in J} f_j(X_j)$$

where J is a discrete index set, X_j is a subset of $\{x_1,\ldots,x_n\}$, and $f_j(X_j)$ is a function only depending on X_j. A factor graph is a bipartite graph with variable nodes, $\{x_1,\ldots,x_n\}$, function nodes $f_j(X_j)$, and edges between functions and the variables they depend on. There are several algorithms defined over factor graphs for computing the following marginal function

$$h_i(x_i) = \sum_{\sim\{X_i\}} h(x_1,\ldots,x_n)$$

where $\sim\{x_i\}$ denotes the summation over all the values of all variables with a fixed value for x_i.

We defined the marginal probability of our hierarchical model as a factor graph problem. For a factor graph with cycles, loopy belief propagation (a message-passing algorithm) can be used to approximate the marginal function. In the straightforward application of belief propagation, the messages from the multinomial distribution functions (Eq 2) to the higher level latent variables require the calculation of a costly sum. We replaced this sum with a particle method approximation, in which an importance sampler over the incoming edges is used [3].

REFERENCES

1. Geman, S. & Geman, D. Stochastic Relaxation, Gibbs Distributions,

and the Bayesian Restoration of Images. *Ieee T Pattern Anal* **6**, 721-741 (1984).

2. Frey, B.J. & Jojic, N. A comparison of algorithms for inference and learning in probabilistic graphical models. *IEEE Trans Pattern Anal Mach Intell* **27**, 1392-1416 (2005).
3. Dauwels, J., Korl, S. & Loeliger, H.A. Particle methods as message passing. *2006 IEEE International Symposium on Information Theory, Vols 1-6, Proceedings*, 2052-2056 (2006).

Supplementary Note 2. **Computing the Medial Section.**

Foreground Segmentation. The foreground root was separated from the background by applying edge detection and active contour methods to the propidium iodide dye outlining the cells. For each slice S in a confocal image z stack, we took all points with a gradient magnitude $|\nabla S|$ larger than an adjustable threshold as interior points of the root. For the particular images in our data set, we empirically determined reasonable thresholds between 1 and 2 percent of the maximum image contrast. The interior points were locally connected through morphological closing of the binary result. Various edge detection and linking methods could be substituted to meet different needs of speed and sensitivity. We took the union of all points found by edge detection and points separated from the image boundary by edge detection to be interior points of the root and therefore part of the foreground. Because we wished to eliminate root hairs, we performed a morphological opening, which has the additional benefit of eliminating speckle noise from the background.

As each of the z-planes of the image was binarized to represent root and non-root pixels, the individual planes were stacked and bit-packed into a 3D binary image. We ran an active contour on a projection of this image to determine the foreground boundary in the x and y directions. We minimized the energy functional

$$E = \int_0^1 \frac{1}{2}(a|\dot{\mathbf{x}}(s)|^2 + \beta|\ddot{\mathbf{x}}(s)|^2 + E_{\mathrm{ext}}(\mathbf{x}(s))\,ds \qquad \text{Eq 1}$$

using a gradient descent method based on Gradient Vector Flow (GVF) [1](Matlab code available at http://www.iacl.ece.jhu.edu/static/gvf/). We used a modified version of GVF and Gradient Vector Convolution (GVC) [2] so that we could handle large image sizes. The GVC is defined by a convolution of the image gradient with a two parameter kernel, $K = (x^2 + y^2 + h)^{-n/2}$, where the parameters n and h control the capture range and the spatial smoothing of the external force.

The choice of the active contour parameters α and β in Equation 1, along with the gradient descent parameters viscosity and external force, determines a particular smoothness to the contour. The appropriate smoothness of the root boundary was different in the meristem than in the differentiated part of the root, and active contours with different parameters were applied separately to these two regions. All of the parameters are dimensionless and it is their relative values that are important. We held α and β constant at 0.1, the viscosity constant at 1.0, and varied only the external image force in the gradient descent step. In the differentiated part of the root we used an external force weight of 0.1, and in the meristem we used values between 1.0 and 1.5. The relative sizes of the parameters created a higher curvature boundary in the meristem, where we expected to see a high curvature at the root tip.

Computing a 3D Medial Axis. Locating a representative longitudinal medial surface was done in two parts. First a 2D medial axis for the foreground shape from the previous step was calculated, and then the depth of the medial axis was determined by modeling the root as a curved cylinder. The 2D medial axis is similar to a skeletonization of the foreground shape [3], but we wanted to: i) obtain a continuous curve rather than an image, ii) eliminate spurious branches, and iii) extend the medial axis to the high curvature tip of the root. These features were accomplished by the shape axis (SA) transformation with an appropriate branching cost [4]. The SA transformation required a choice of similarity criteria; we found that the differentiated parts of the root were best described by a parallelism criterion and the meristem was best described by a co-circularity criterion.

The smoothness of the boundary guaranteed by the active contour allowed the use of a stricter version of the SA transform for the parallel portion of the root that naturally separates it from the co-circular part. The SA transform paired every point on the boundary with a parallel point on the boundary (or with itself) by globally minimizing the energy function $\sum_{i\sim j}\left|\tau_i \cdot \tau_j\right|$, where τ is the normalized tangent vector to the boundary and $i \sim j$ indicates the i^{th} and j^{th}

points are paired. Our approach avoided the challenge of a global minimization by first pairing only those points for which the energy function is zero. This can be accomplished with a complexity on the order of a binary search on the boundary and effectively reduces the problem to matching boundary points over several smaller segments of the boundary. Then the SA transform can be applied on the remaining boundary fragments requiring the energy function to be minimized on each fragment rather than globally. We have found that filling the gaps with appropriately smooth functions instead worked well in practice. We chose Bezier splines to join the matched boundary points. The splines have 4 control points, the two endpoints of the spline that are halfway between the paired boundary points and two additional points displaced from the endpoints in a direction tangent to the boundary at the matched point at a distance one third of the Euclidean distance between the endpoints.

In addition to being more computationally efficient than the full transform, the strict approach guaranteed a single axis with two endpoints and no branching. This axis did not extend into the meristem of the root and ended at the boundary between the parallel and co-circular parts of the root. The axis was extended to the tip of the root using the following co-circularity criterion

$$\sum_{i\sim j}\frac{\left|(x_i - x_j)(\tau_i - \tau_j)\right|}{\left|x_i - x_j\right|\left|\tau_i - \tau_j\right|} \qquad \text{Eq 2}$$

where ***x*** is a point on the boundary and ***τ*** is the tangent vector. Tangents were computed in opposite directions, vertical bars indicate Euclidean norms, and $i \sim j$ indicates the i^{th} and j^{th} points are paired. The point at the tip of the root was the self matched point when this criterion was minimized.

Once the location of the medial axis in the *x* and *y* directions was computed, the depth was determined by considering the root as a curved cylinder in 3D. At each point along the two-dimensional medial axis, we extended a perpendicular line to the edge of the foreground root found in the first step. This provided the diameter of the cylinder at that location. The location of the top surface of the root was taken from the 3D binary image computed earlier. The depth of the medial axis was then simply half the diameter deeper than the top surface. In order to overcome the low resolution in the *z*-direction, the depth was smoothed by a Gaussian convolution. The width of the Gaussian was chosen such that its frequency counterpart had a width of 2/3rds the Nyquist frequency of the medial axis resolution. Because the medial axis depth was not periodic, we reflected the medial axis about its endpoints. We performed the convolution for a single reflection about the endpoint

$$R(s) = \begin{cases} D(s) & \text{for } s>0 \\ D(0)-D(-s) & \text{for } s<0 \end{cases} \qquad \text{Eq 3}$$

where R is the reflection, D is the original depth and s is the distance along the axis and ranges from zero to the length of the axis. For a double reflection about both the endpoint and the endpoint depth:

$$R(s) = \begin{cases} D(s) & \text{for } s>0 \\ D(0)-D(-s) & \text{for } s<0 \end{cases} \qquad \text{Eq 4}$$

We chose the smoothed version with minimal square difference from the original. Combining the depth information with the 2D information resulted in a medial axis curve for the root in the 3D image.

Reconstructing a Medial Longitudinal Section. A representative medial longitudinal section is a surface that intersects the medial axis and extends perpendicular to it (Figure 3). There are many such sections and we chose one such that the depth of the surface perpendicular to the medial axis was constant. A set of points on the surface with the same spacing as the x-y resolution of the original image was computed, and the quasi 3D image stack was interpolated at these points to create an image of the medial longitudinal section of the root. A straightened image of the medial section was computed with x-y resolution in the same way as the original image. The pixel value at a distance *s* from the tip and *r* from the medial axis of the straightened image was computed by interpolation of the original image at a distance *s* from the tip along the medial axis and a distance *r* from the medial axis in a direction perpendicular to the medial axis in the plane of the medial section surface. The straightened image was registered to an atlas based on the most prominent geometric features of the root, the tip and the longitudinal distance where the root width becomes nearly constant.

REFERENCES

1. Xu, C.Y. & Prince, J.L. Gradient vector flow: A new external force for snakes. *Proc Cvpr Ieee*, 66-71 (1997).
2. Wang, Y.Q. & Jia, Y.D. External Force for Active Contours: Gradient Vector Convolution. *Lect Notes Artif Int* **5351**, 466-472 (2008).
3. Serra, J.P. & Cressie, N.A.C. Image analysis and mathematical morphology. (Academic Press, 1982).
4. Geiger, D., Liu, T.L. & Kohn, R.V. Representation and self-similarity of shapes. *Ieee T Pattern Anal* **25**, 86-99 (2003).