Molecular Plant • Volume 4 • Number 5 • Pages 922–931 • September 2011 RESEARCH ARTICLE

Adaptive Cell Segmentation and Tracking for Volumetric Confocal Microscopy Images of a Developing Plant Meristem

Min Liu[a,2], Anirban Chakraborty[a,2], Damanpreet Singh[b], Ram Kishor Yadav[c], Gopi Meenakshisundaram[b], G. Venugopala Reddy[c,1] and **Amit Roy-Chowdhury[a,1]**

a Department of Electrical Engineering, University of California, Riverside, CA, USA
b Department of Computer Science, University of California, Irvine, CA, USA
c Department of Botany and Plant Sciences, and Center for Plant Cell Biology, University of California, Riverside, CA, USA

ABSTRACT Automated segmentation and tracking of cells in actively developing tissues can provide high-throughput and quantitative spatiotemporal measurements of a range of cell behaviors; cell expansion and cell-division kinetics leading to a better understanding of the underlying dynamics of morphogenesis. Here, we have studied the problem of constructing cell lineages in time-lapse volumetric image stacks obtained using Confocal Laser Scanning Microscopy (CLSM). The novel contribution of the work lies in its ability to segment and track cells in densely packed tissue, the shoot apical meristem (SAM), through the use of a close-loop, adaptive segmentation, and tracking approach. The tracking output acts as an indicator of the quality of segmentation and, in turn, the segmentation can be improved to obtain better tracking results. We construct an optimization function that minimizes the segmentation error, which is, in turn, estimated from the tracking results. This adaptive approach significantly improves both tracking and segmentation when compared to an open loop framework in which segmentation and tracking modules operate separately.

Key words: Shoot apical meristem; stem cells; cell tracking; cell segmentation; integrated segmentation and tracking.

INTRODUCTION

Pattern formation in developmental fields involves the precise spatial arrangement of different cell types in a dynamic landscape wherein cells exhibit a variety of behaviors such as cell division, cell expansion, and cell migration (Meyerowitz, 1997). Cell–cell communication between cells located in different cell layers of multilayered tissues guides specification of different cell types and also cell behavioral patterns. Therefore, a quantitative understanding of the dynamics of spatial and temporal organization of gene expression patterns within multilayered and actively growing developmental fields is crucial to the process of pattern formation.

As an example, the shoot apical meristem (SAM) stem-cell niche in plants is a dynamic multilayered structure consisting of about 500 cells, and it provides cells for all the above-ground plant structures (Steeves and Sussex, 1989). The SAM has been divided into distinct functional domains based on differential gene expression patterns and growth patterns (Figure 1). The cells in the outermost L1 layer and the sub-epidermal L2 layer divide in anticlinal orientation (perpendicular to the SAM surface), while the underlying corpus forms a multilayered structure in which cells divide in random orientations. Within this framework, the SAM is divided into functional zones each with distinct gene expression activities. The central zone (CZ) contains a set of stem cells and the progeny of stem cells differentiate within the adjacent peripheral zone (PZ). The CZ also supplies cells to the rib-meristem (RM) located beneath the CZ. Thus, cell fate specification within SAMs is a dynamic process intimately coupled to transient changes in gene activation/repression and also changes in growth patterns (Williams and Fletcher, 2005). Thus, the SAM represents a 3-D network of cells in which cell-growth patterns and differentiation are dynamically regulated.

Recently, novel live-imaging methods to record the gene expression dynamics and growth patterns have been developed (Dumais and Kwiatkowska, 2002; Grandjean et al., 2003;

[1] To whom correspondence should be addressed. E-mail amitrc@ee.ucr.edu, tel. 951 827 7886, fax 951 827 4437.

[2] These authors contributed equally to this work and both should be given the status of first author.

doi: 10.1093/mp/ssr071
Received 19 May 2011; accepted 20 July 2011

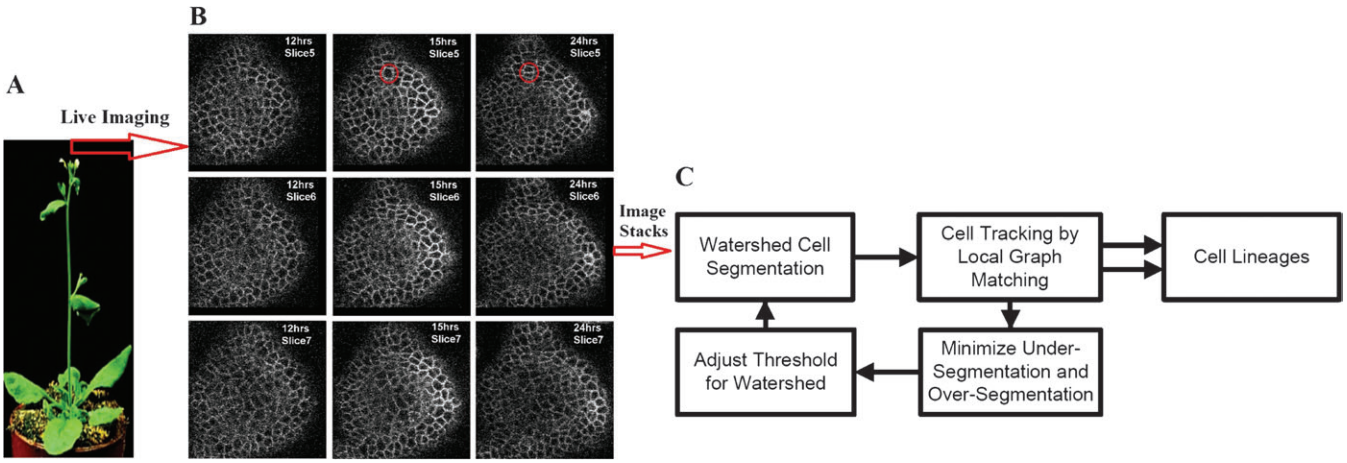

Figure 1. Overall Imaging and Image Analysis Framework
(A) SAM located at the tip *Arabidopsis* shoot.
(B) Time-lapse imagery of SAMs labeled with plasma membrane localized YFP. Examples of three image slices (vertical columns) at three time instants (horizontal columns). The red circles denote a cell division. Cross-sectional images of SAMs depicted in the vertical columns are separated by 1.5 μm, while the size of a single cell is about 5 μm in diameter. Therefore, each cell is represented in two or three consecutive slices.
(C) The diagram of the adaptive cell segmentation and tracking scheme.

Reddy et al., 2004; Heisler et al., 2005). The novel live-imaging method utilizes laser scanning confocal microscopy to acquire time series of the 3-D imagery of SAMs for up to 4–5 d. An example of one such 4-D dataset in which the SAM is labeled with plasma membrane localized Yellow Fluorescent Protein (YFP) to follow cell outlines of individual cells is shown in Figure 1B. The manual tracking of cells through successive cell divisions and gene expression patterns is beginning to yield new insights into the process of stem-cell homeostasis (Reddy and Meyerowitz, 2005). However, manual tracking is laborious and impossible as larger and larger amounts of microscope imagery are collected worldwide. More importantly, manual analysis will not provide quantitative information on cell behaviors, besides cell-cycle length and the number of cell divisions within a time period. There are significant challenges in automating the segmentation and tracking process. The SAM cells in a cluster appear very similar with few distinguishing features, cell-division event changes the relative locations of the cells, and the live images are inherently noisy.

There has been some work on automated tracking and segmentation of cells in time-lapse images, both plants and animals. One of the well-known approaches for segmenting and tracking cells is based on level-sets (Chan and Vese, 2001; Li and Kanade, 2007; Cunha et al., 2010). However, the level-set method is not suitable for tracking of SAM cells because the cells are in close contact with each other and share similar physical features. The Softassign method uses the information on point location to simultaneously solve both the problem of global correspondence as well as the problem of affine transformation between two time instants iteratively (Chui, 2000; Gor et al., 2005; Rangarajan et al., 2005). A recent paper (Fernandez et al., 2010) uses SAM images acquired from multiple angles to automate tracking and modeling. Since SAMs are imaged from multiple angles, it imposes a limitation on the temporal resolution. This precludes a finer understanding of spatial–temporal dynamics through dynamical modeling. In an earlier study, we have used level-set segmentation and local-graph matching method to find correspondence of cells across time points by using live imagery of plasma membrane labeled SAMs (Liu et al., 2010) imaged at 3-h intervals. However, this study did not make an attempt to integrate segmentation and tracking so as to minimize the segmentation and tracking errors (Figure 2B and 2C), which are major concerns in noisy live imagery. Here, we have combined the local-graph matching-based tracking methodology from Liu et al. (2010) with the watershed segmentation in an adaptive framework in which tracking output is integrated with the segmentation (Figure 1C).

RESULTS

We first provide a brief description about the building blocks of our system—the watershed segmentation method (Najman and Schmitt, 1994) and local-graph matching-based tracker (Liu et al., 2010), and then provide details about the integrated adaptive segmentation and tracking framework along with the results.

Watershed Segmentation

We used watershed transformation (Beucher and Lantuejoul, 1979; Najman and Schmitt, 1994) to segment cell boundaries. Watershed treats the input image as a continuous field of basins (low-intensity pixel regions) and barriers (high-intensity pixel regions), and outputs the barriers that represent cell boundaries. It has been used to segment cells of *Arabidopsis thaliana* root meristem (Marcuzzo et al., 2008). It outperforms the level-set method in two aspects. On one hand, it reaches

more accurate cell boundaries, which can be clearly noticed by comparing Figure 3B and 3C. On the other hand, it is faster, which provides the opportunity to implement it in an adaptive framework efficiently. However, the main drawback is that it results in both over-segmentation and under-segmentation of cells, especially those from deeper layers of SAMs that are noisy. So, prior to applying the watershed algorithm, the raw confocal microscopy images undergo H-minima transformation in which all the pixels below a certain threshold percentage *h* are discarded (Soille, 2003). The H-minima operator was used to suppress shallow minima, namely those whose depth is lower than or equal to the given *h*-value. The watershed segmentation after the H-minima operator with a proper threshold can produce much better segmentation results than level-set segmentation. As shown in Figure 3B, the segmented cells have much more accurate boundaries than the cells segmented by level-set algorithm (Figure 3C).

Since the H-minima threshold value *h* plays a very crucial role in the watershed algorithm, especially when the input images are noisy, it is extremely important to choose an appropriate threshold value such that only the correct cell boundaries are detected. Generally, a higher value of the threshold parameter *h* performs under-segmentation of the images (Figure 2C) and, conversely, a lower value over-segments the images (Figure 2B). This is also evident in Figure 4C, which shows that the number of over-segmented cells in a noisy image slice increases as we choose lower values for the H-minima threshold *h* and, on the other hand, a larger value of *h* produces more under-segmented cells. Since the cell size is fairly uniform for most cells of the SAM, the watershed should ideally produce a segmented image that contains similarly sized cells. Thus, a good starting threshold could be the value of *h* such that variance of cell areas in the segmented image is minimal. This optimal value of *h* is what we are trying to obtain, as will be explained later.

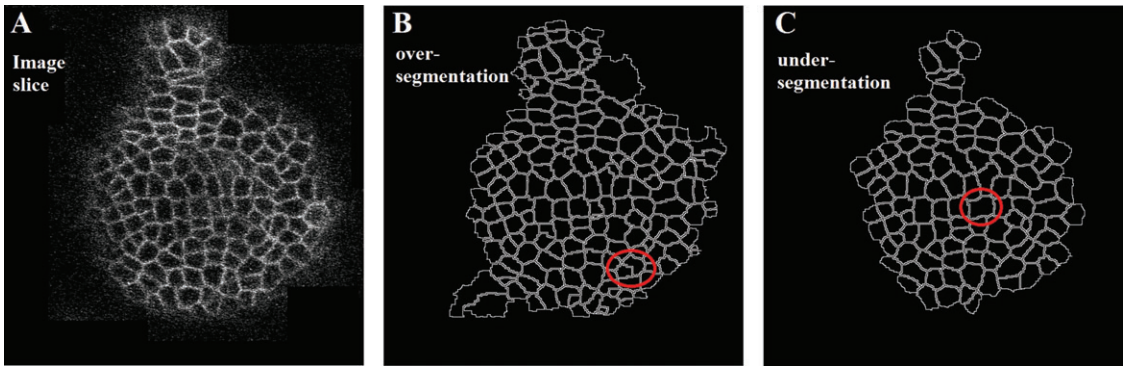

Figure 2. Illustration of Over-Segmentation and Under-Segmentation by Watershed Method.
(A) The input image slice.
(B) Over-segmented cell denoted by red circle.
(C) Under-segmented cell denoted by red circle.

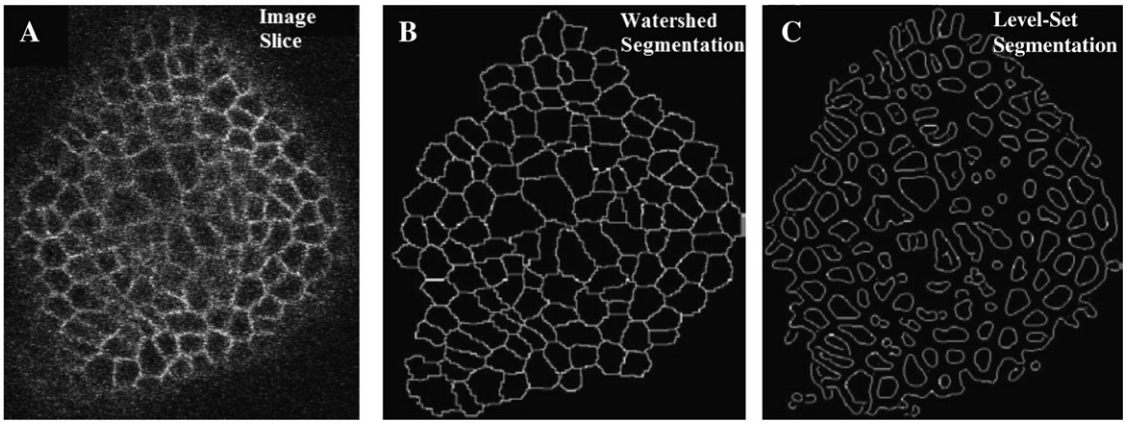

Figure 3. Improved Segmentation Results Obtained Using Watershed Transformation in Comparison to the Level-Set Segmentation.
(A) The input image slice.
(B) The segmented image by using watershed segmentation.
(C) The result of level-set segmentation.

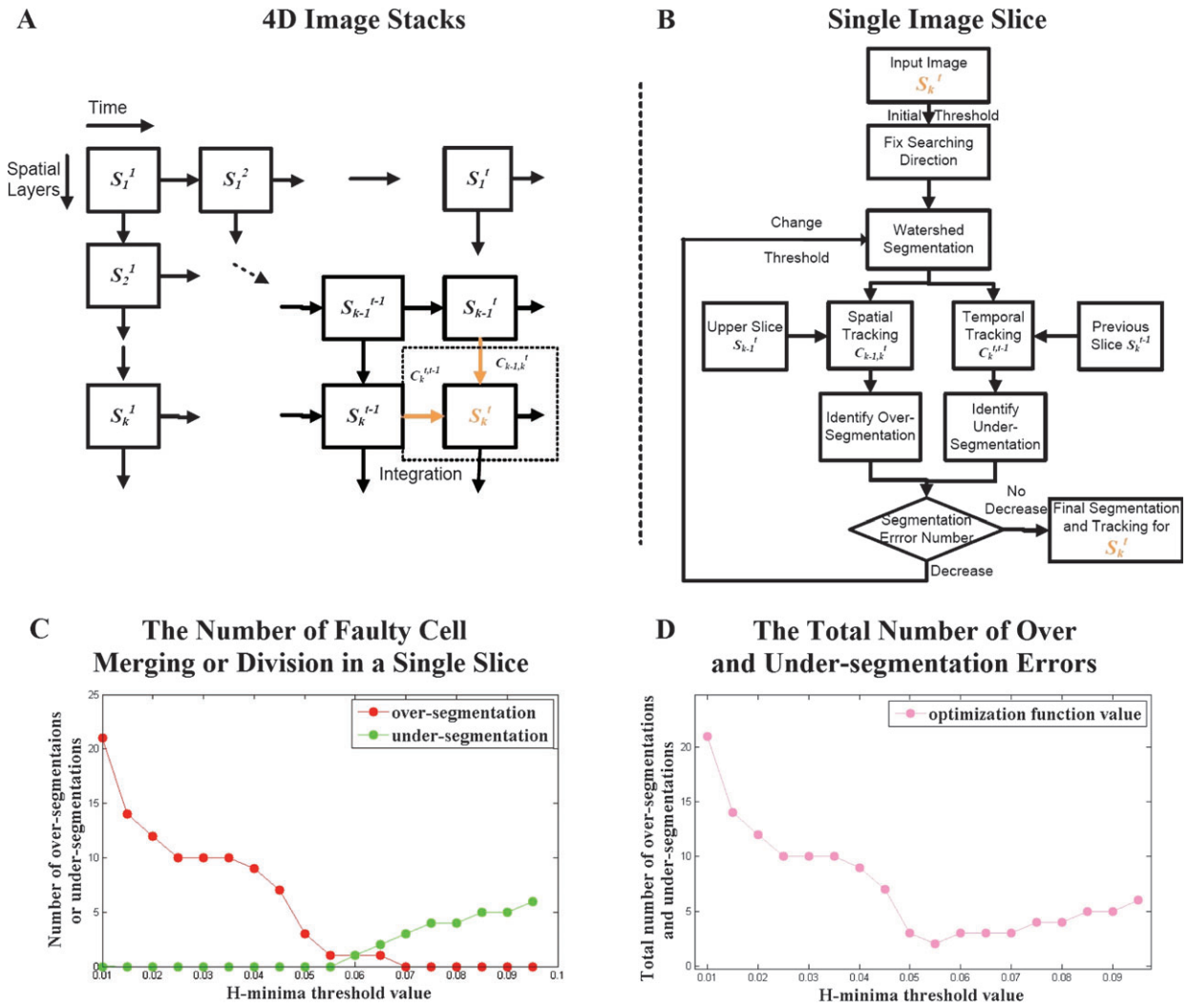

Figure 4. Optimization Scheme.

(A) Schematic showing how to integrate the spatial and temporal trackers for 4-D image stacks.
(B) Adaptive segmentation and tracking scheme for a certain image slices S_k^t (the k^{th} slice at the t time point).
(C) The illustration of the number of faulty cell-merging events (green plot) or spurious cell divisions (red plot) with respect to the changing of H-minima threshold h, in one noisy image slice.
(D) The total number of over-segmentation and under-segmentation errors with respect to the changing of H-minima threshold h.

Cell Tracking Using Local-Graph Matching

We build upon the local-graph-matching framework, in which a vertex in the graph represents every cell and neighboring vertices are connected by an edge (Liu et al., 2010). The graph structure automatically includes the relative position information of the cells, such as the relative distance between two neighboring cells (the edge length) and the edge orientation. As described in Liu et al. (2010), we can find the most similar cell pair (known as the 'Seed Pair') by matching the relative positions of cells with respect to their nearest neighbors through the local graph for any two consecutive time points. Starting from this seed pair, we grow the number of matched cells by computing the similarities between local regions in the graph. Cell divisions are detected by detecting changes in the topology of the graph.

The tracker performance depends heavily on the quality of the segmentation output. However, due to a low Signal-to-Noise Ratio (SNR) in the live cell imaging dataset, the cells are often over or under-segmented. Therefore, the segmentation and tracking have to be carried out in an integrated and adaptive fashion, where the tracking output for a particular slice acts as an indicator of the quality of segmentation and the segmentation can be improved so as to obtain the best tracking result.

Design of Integrated Optimization Function

Due to the rapid deterioration of image quality in deeper layers of the Z-stack, the existing segmentation algorithms tend to under-segment or over-segment image regions

(especially in the central part of the image slices). Even a manual segmentation of cells is not always guaranteed to be accurate if each slice in the deeper layers is considered separately due to very low SNR. In fact, in such cases, we consider the neighboring slices, which can provide additional contextual information to perform segmentation of the noisy slice in a way that provides the best correspondence for all the segmented cells within the neighborhood. The automated method of integrated segmentation and tracking proposed here involves correcting faulty segmentation of cells by integrating their spatial and temporal correspondences with the immediate neighbors as a feedback from the tracking to the segmentation module. In the next few paragraphs, we formalize this framework as a spatial and temporal optimization problem and elaborate the proposed iterative solution strategy that yields the best segmentation and tracking results for all the cell slices in the 4-D image stack.

The advantage of using watershed segmentation is that it can accurately find the cell boundaries, while its main drawback is over-segmentation and under-segmentation, which can be reduced by choosing the proper H-minima threshold. Due to the over-segmentation errors in regions of very low SNR, the watershed algorithm often tends to generate spurious edges through the cells. In cases in which a cell is imaged at multiple slices along the Z-stack and is over-segmented in one of the slices, the tracker can identify this over-segmentation error as a spurious 'spatial cell-division' event. Clearly, this is not legitimate and is a result of faulty segmentation. Additionally, cell merging in the temporal direction (again an impossible event) can arise from under-segmentation, where the watershed algorithm fails to detect a legitimate edge between two neighboring cells. The intuition behind the proposed method in this paper is to reduce the over-segmentation errors by minimizing the spurious spatial cell divisions and reduce the under-segmentation errors by minimizing the number of merged cells. Specifically, for frame, S_k^t the k^{th} image slice at time point t, we are going to minimize the number of spurious 'cell divisions' between it and its spatial neighbor S_{k-1}^t and the number of spurious cell-merging events in S_k^t from its temporal predecessor S_k^{t-1}, as shown in Figure 4. (Although it may be possible to identify that such spurious events have happened through simple rules disallowing a cell division in the Z-direction or a merging in the forward temporal direction, correcting for them is a lot harder, as the structure of the collection of cells needs to be maintained. Our approach will allow detection of not only such spurious events, but also their correction.)

The spatial and temporal correspondences between cell slices across neighboring confocal images can be represented through correspondence matrices. As an example, $C_{(k-1,k)}^t$ is the correspondence matrix from the slice S_{k-1}^t to S_k^t, and $C_k^{(t,t-1)}$ is the correspondence matrix from the frame S_k^t to S_k^{t-1}. Each element in correspondence matrix C denotes whether two cell i, j in two images (may be at different time instances or different slices at the same time) are the same cell or not, namely:

$$\begin{cases} C[i,j]=0, \text{if } i \text{ and } j \text{ are not matched,} \\ C[i,j]=1, \text{if } i \text{ and } j \text{ are matched.} \end{cases} \quad (1)$$

For a cell $i \in S_k^{t-1}$, if there exists a cell pair $\{j_1, j_2\} \in S_k^t$ such that $C[i,j_1]=C[i,j_2]=1$, then we can identify that there is a cell-division event in which the cell, i has divided into two daughter cells j_1 and j_2. On the other hand, if, for a cell $j \in S_k^t$, there exists a cell pair $\{i_1, i_2\} \in S_k^{t-1}$ such that $C[i_1,j]=C[i_2,j]=1$, then a cell-merging event is readily detected, in which cells $i_1$and i_2 from the previous slice S_k^{t-1} have merged into a single cell j in slice S_k^t. This is clearly the result of under-segmentation on the slice S_k^t that needs to be corrected. In a very similar way, the errors caused by over-segmentation can also be detected from the tracking output. For any two spatially consecutive slices S_{k-1}^t and S_k^t in the same Z-stack, if multiple cells in S_k^t correspond to one single cell in S_{k-1}^t (i.e. there exists a one-to-many correspondence between cells in spatially consecutive slices), an over-segmentation error in the slice S_k^t is detected.

The under-segmentation and the over-segmentation error can be quantitatively represented as the number of faulty 'cell-merging events' and the 'spurious cell divisions' as obtained from the spatio-temporal correspondences generated by the tracker. For example, the number of the illegal cell-merging events can be counted as:

$$\begin{cases} Q_k^{(t,t-1)} = \left\{ \text{cell } i \text{ in } S_k^t \middle| \exists \text{ a pair } (j_1,j_2) \text{ in } S_k^{t-1} \text{ for which } C_k^{(t,t-1)}[i,j_2] \right. \\ \left. = C_k^{(t,t-1)}[i,j_2]=1 \right\}, N_k^{(t,t-1)} = \text{ size of } \left(Q_k^{(t,t-1)}\right), \end{cases} \quad (2)$$

where $Q_k^{(t,t-1)}$ is the set of under-segmented cells in S_k^t and $N_k^{(t,t-1)}$ is the total number of such cells. Similarly, we can compute $N_{(k-1,k)}^t$ as the number of spurious cell divisions (as described previously) in the spatial tracking from S_{k-1}^t to S_k^t.

The optimization goal here is to minimize the number of spurious cell divisions $\left(N_{(k-1,k)}^t\right)$ for the frame S_k^t from its upper slice S_{k-1}^t and the number of cell-merging events $N_k^{(t,t-1)}$ in S_k^t from its previous slice . As can be seen in Figure 4C, the error caused by over-segmentation, $N_{(k-1,k)}^t$, monotonically decreases (red curve in Figure 4C) with the increment in threshold h, whereas the error due to under-segmentation, $N_k^{(t,t-1)}$, monotonically increases (green curve in Figure 4C) with h. Hence, the optimal segmentation result can be obtained through finding the value of h for which the summation $\left(N_k^{(t,t-1)}+N_{(k-1,k)}^t\right)$ attains a minimum (as the lowest point in the pink curve in Figure 4D). The cost function $\left(N_k^{(t,t-1)}+N_{(k-1,k)}^t\right)$ is essentially an indicator of the overall error in segmentation (combining both over and under-segmentation) and can be optimized by varying the H-minima threshold h for S_k^t. Formally, the optimal value h_k^t is found as a solution to the following optimization problem:

$$h_k^t = \min_h \left(N_k^{(t,t-1)}(h) + N_{(k-1,k)}^t(h)\right). \quad (3)$$

With the variation of h_k^t (by either increasing or decreasing), the cost function decreases to a minimum (ideally 0). The threshold h_k^t for which the cost function attains this minimum is the optimum value of the threshold for H-minima transformation.

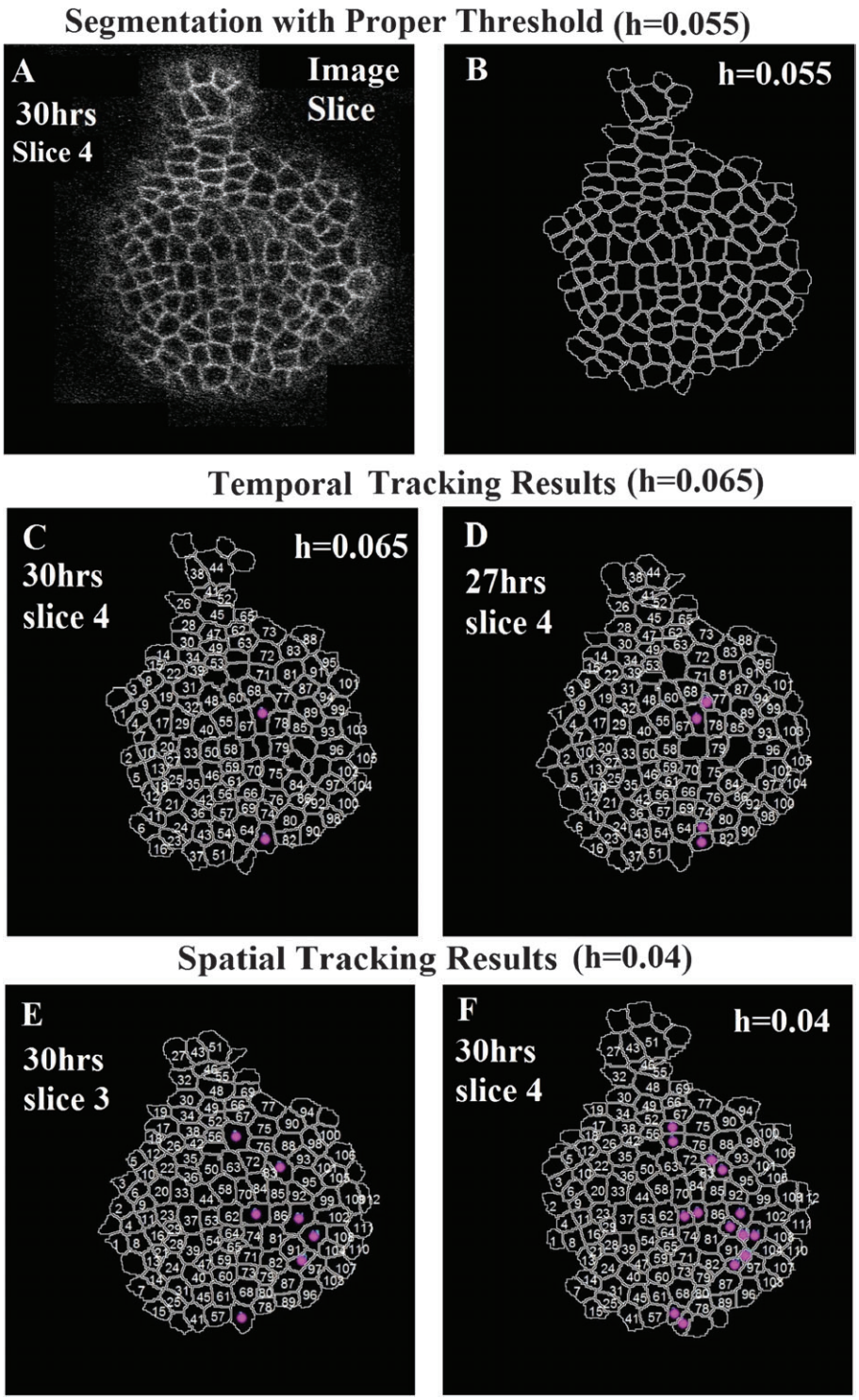

Figure 5. Illustration of Faulty Cell-Merging Events Caused by Under-Segmentation or Spurious Cell Divisions Caused by Over-Segmentation.
(A) The fourth slice of confocal Z-stack taken at the 30th-hour time point.
(B) The correct segmentation result with the optimal H-minima threshold value 0.05, which was found by the proposed adaptive method.
(C, D) The temporal tracking results (from the 30th-hour time point to the 27th-hour time point) with faulty cell-merging events (denoted by purple dots), which are caused by under-segmentation with H-minima threshold value 0.065.
(E, F) The spatial tracking results (from the third confocal slice to the fourth confocal slice along the z-scale) with spurious cell divisions (denoted by purple dots) and this is caused by over-segmentation. The same number denotes two corresponding slices along the Z-direction.

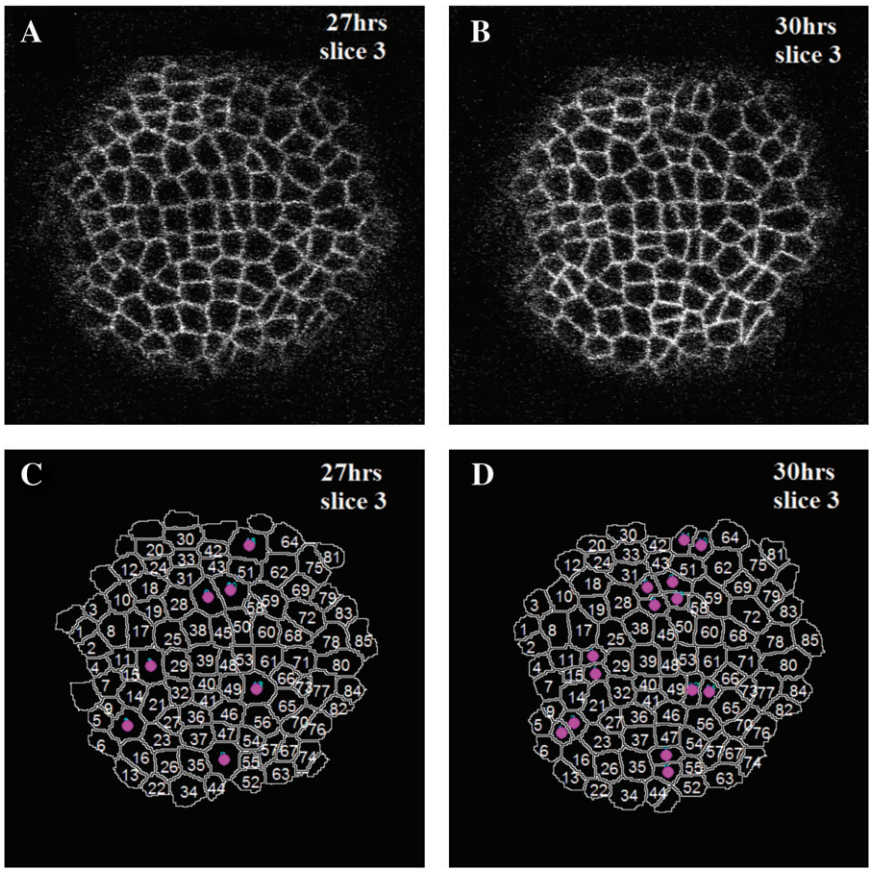

Figure 6. An Example of Adaptive Segmentation and Tracking Output.
The same number denotes corresponding cells across time points and the purple dots denote cell divisions.
(A, B) The original image slices at two time instances: 27th and 30th hours.
(C, D) The segmentation and tracking results of (A) and (B) using adaptive method.

Note that, given a value of h, we can compute N, although an analytical representation relating the two is unknown.

Optimization Scheme

Given 4-D image stack series $\{S_k^t, k{=}1:t{=}1:T\}$ as shown in Figure 4A (where k is the index for depth and t is for time), the algorithm proceeds as follows:

(1) We first estimate for $h_{k(init)}^t$ every image slice that minimizes the variances in cell areas in each of those slices and perform watershed segmentation with these estimated thresholds. These act as the initial estimates of the segmentation thresholds in our algorithm.

(2) For image slices $\{S_k^t, k{=}2:t{=}1\}$, we compute the optimum watershed thresholds $h_k^t \in [h_{\min}, h_{\max}]$ by minimizing the number of cell divisions $N_{(k-1,k)}^1$ in each of these slices. We repeat this process for slices $\{S_k^t, k{=}1:t{=}2:T\}$ and obtain optimum thresholds by minimizing $N_1^{(t,t-1)}$.

(3) For other image slices $\{S_k^t, k{=}2:k, t{=}2:T\}$, we vary the thresholds h_k^t in the range $[h_{\min}, h_{\max}]$ starting from the initial estimate $h_{k(init)}^t$. The search direction, namely either of $h_k^t > h_{k(init)}^t$ or $h_k^t < h_{k(init)}^t$, is chosen such that there is a decrease in the cost function $F_k^t = \left(N_k^{(t,t-1)} + N_{(k-1,k)}^t\right)$ with a change in h_k^t along the search direction. This can be done using a simple exhaustive search in a local neighborhood around $h_{k(init)}^t$. Once the search direction is fixed, we keep on varying h_k^t along the search direction until we encounter an increment in the cost function. At this value of h_k^t, beyond which the cost function increases, a minimum of the cost function is reached. We stop the search at this point and set h_k^t as the optimum watershed segmentation threshold for S_k^t (as shown in Figure 4B). We segment S_k^t with this threshold and obtain optimum tracking result while tracked from its immediate neighbors.

The search method described above is guaranteed to converge to a minimum of the cost function. As we have observed throughout our experiments, this minimum is generally the global minimum, although it cannot be shown analytically that a global minimum is always ensured. If run time of the algorithm is not an issue, an exhaustive search method over the entire range $[h_{\min}, h_{\max}]$ can also be employed to guarantee that a global minimum is always attained.

We have tested our proposed adaptive cell segmentation and lineage construction algorithm on two SAM datasets.

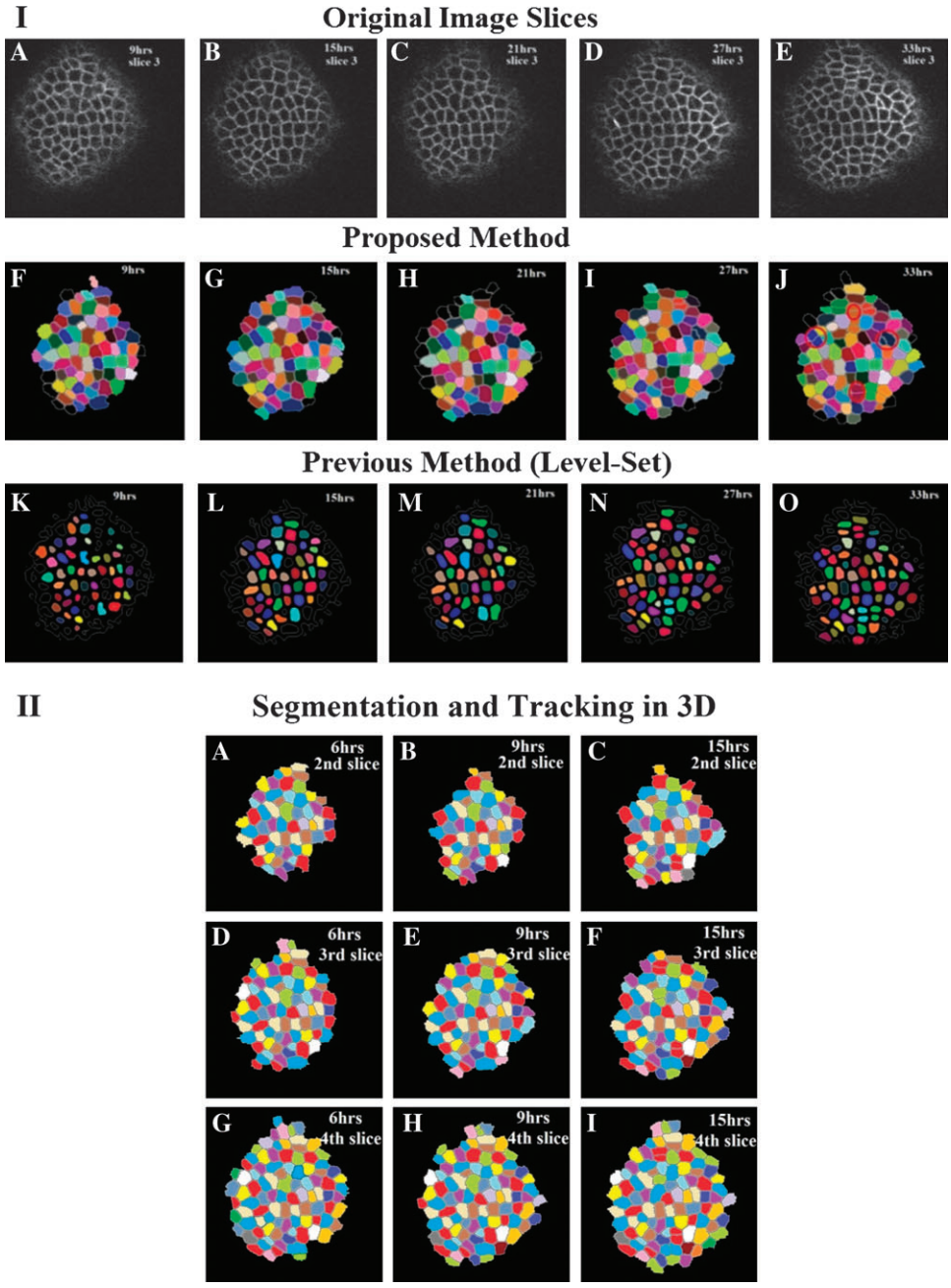

Figure 7. Segmentation and Tracking Results.

(I) An example of the estimated cell lineages. The first row is the original image slice series at the 9th, 15th, 21st, 27th and 33rd hours. The second row shows the segmentated cells and the estimated cell lineages using the proposed algorithm. The third row is the result from Liu et al., 2010) in which cell segmentation and tracking are not integrated. The same color in consecutive frames denotes the same cell. Cell-division examples are denoted by red circles in the second row.

(II) The segmentation and tracking results in 3-D stacks at selected time instances. The segmented cells shown in same color across consecutive slices (second, third, and fourth slices) represent same cells.

Datasets consist of 3-D images stacks taken at 3-h intervals for a total of 72 h (24 data points). Each 3-D stack consists of 30 slices in one stack, so the size of the 4-D image stack is 888 $\times$ 888 $\times$ 30 $\times$ 24 pixels. Registration is performed using the alignment method of Maximization of Mutual Information (Viola and Wells, 1995). We used the local-graph matching-based algorithm to track cells, the detailed information about which can be found in Liu et al. (2010). We demonstrate that, by integrating this within the proposed adaptive scheme, we are able to obtain significantly better results. The adaptive segmentation and tracking method is run on every two consecutive images in both the spatial

Table 1. The Number of Cells Correctly Tracked Using the Algorithm in Liu et al. (2010).

Dataset Method	Percent of correctly tracked cells			Average time over which lineages could be tracked (hours)		
	Method (Liu et al., 2010)	No adaption	This method	Method (Liu et al., 2010)	No adaptation	This method
Dataset1	71	90	98	25	50	56
ataset2	36	83	95	6	22	33

The watershed segmentation without integration and the method proposed here (this method) are shown in the left column. The average time period over which the lineages could be tracked is shown in the right column.

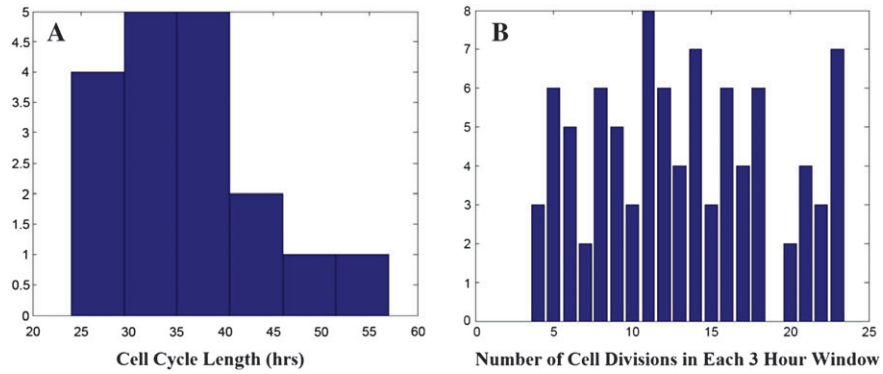

Figure 8. Statistical Analysis Results Based on the Segmentation and Tracking Results Using our Proposed Adaptive Method.
(A) The histogram showing cell-cycle length. **(B)** The number of cell divisions across each 3-h interval.

and temporal directions. The thresholds of H-minima for images in the given 4-D image stack are determined sequentially along the direction of the arrows shown in Figure 4A. In the segmentation, we normalized the image intensities in the range of [0 1] and set the searching range for the optimal H-minima threshold h in [0.005 0.09]. The step size used in the search was ± 0.005, and the sign depends on the search direction ($h_k^t > h_{k(init)}^t$ or $h_k^t < h_{k(init)}^t$). We manually verified the accuracy of the cell lineages obtained by the proposed algorithm.

Using the adaptive segmentation and tracking, we can find the optimal threshold for watershed segmentation with minimal over-segmentation and under-segmentation. The whole idea can be illustrated through an example in Figure 5. Figure 5A is the original image at 30 h and in the 4th slice, and Figure 5B is the segmentation result by our proposed adaptive method with the optimal H-minima threshold $h = 0.055$, which is found by minimizing the spatial faulty cell divisions (caused by over-segmentation) and temporal faulty cell-merging events (caused by under-segmentation). Using watershed segmentation with other H-minima thresholds, there will be either spatial over-segmentation or temporal under-segmentation. For example, there are two faulty cell-merging events in the temporal tracking (as shown in Figure 5C and 5D) that indicate under-segmentation caused by too high a H-minima threshold ($h = 0.065$), while the spurious cell divisions in the spatial tracking (as shown in Figure 5E and 5F) indicate over-segmentation caused by too low a H-minima threshold ($h = 0.04$).

Figure 6 is a typical example of the segmentation result and tracking result using our proposed adaptive method, with seven simultaneously dividing cells being detected. Figure 7(I) is an example of the estimated cell lineages across 24 time points (72 h) with five specific time points shown. The first row is the original image series. The second row shows the cell lineage using the proposed algorithm while the third row is the result from Liu et al. (2010). The same color in consecutive frames denotes the same cell at different time points. Some cell-division examples are highlighted by red circles. Figure 7(I) demonstrates the difference in segmentation and cell lineages achieved using the method in Liu et al. (2010) and our proposed algorithm. In Figure 7(II), the segmentation and tracking results in selected 3-D image stacks along three time instances (6, 9, and 15 h) are shown. The tracker is able to compute both spatial and temporal correspondences with high accuracy as a result of improved segmentation.

In order to verify the overall improvement of the proposed algorithm, we compared the cell-tracking statistics from the proposed method with that in Liu et al. (2010). The number of correctly tracked cells obtained by tracking images in consecutive time points with a 72-h period is compared in Table 1. The comparison is done on different datasets and we can see significant increase in the percentage of tracked cells obtained from the proposed method. Other important data to obtain are the cells' lineage lengths. Table 1 also shows the comparison of the average length of cell lineages between the proposed method and the algorithm in Liu et al. (2010), and confirms that the proposed method greatly improves the tracking results, especially

in the deeper central regions of the SAM. There are 27 more cells (21 of them are in the deeper central region) that maintain their lineages over a complete 72 h by our proposed method compared to the method in Liu et al. (2010).

DISCUSSION

The main challenge in segmenting and tracking cells of SAMs is that the cells are tightly packed in a multilayered structure and they share very similar features. We have addressed this problem by using an adaptive watershed segmentation and tracking method that exploits the geometric structure and topology of the relative positions of cells. Building upon our previous work on local-graph matching-based cell tracking, we show this adaptive scheme significantly outperforms the earlier approach. The closed-loop segmentation and tracking approach minimizes over-segmentation and under-segmentation errors by adapting the segmentation parameters based on the tracking output. This improvement in cell segmentation and tracking has enabled us to obtain a set of sufficiently accurate statistics that could be very useful in plant cell quantitative growth analysis and modeling. With the greatly improved tracking results, we can quantify cellular dynamics such as cell-cycle length and number of cell-division events within a given interval (Figure 8A and 8B). The statistics obtained through the proposed automated framework show very close similarity to the nature of cell growth and division statistics obtained in the earlier work (Reddy et al., 2004). Such statistical information is extremely useful in modeling the cell- growth dynamics.

FUNDING

This work is partially funded by National Science Foundation grants (IIS-0712253) to A.R.C. and (IOS-0718046) to G.V.R.

ACKNOWLEDGMENTS

We thank the microscopy core facility of the Center for Plant Cell Biology (CEPCEB) and the Institute of Integrative Genome Biology (IIGB), University of California, Riverside. No conflict of interest declared.

REFERENCES

Beucher, S., and Lantuejoul, C. (1979). Use of watersheds in contour detection. International Workshop on Image Processing: Real-Time Edge and Motion Detection/Estimation (France: Rennes).

Chan, T., and Vese, L. (2001). Active contours without edges. IEEE Transactions on Image Processing. **10,** 266–277.

Chui, H. (2000). A new algorithm for non-rigid point matching. IEEE Computer Society Conference on Computer Vision and Pattern Recognition. **2,** 44–51.

Cunha, A.L., Roeder, A.H.K., and Meyerowitz, E.M. (2010). Segmenting the sepal and shoot apical meristem of *Arabidopsis thaliana*. 2010 Annual International Conference of the IEEE Engineering in Medicine and Biology Society .

Dumais, J., and Kwiatkowska, D. (2002). Analysis of surface growth in shoot apices. Plant J. **31,** 229–241.

Fernandez, R., Das, P., Mirabet, V., Moscardi, E., Traas, J., Verdeil, J.L., Malandain, G., and Godin, C. (2010). Imaging plant growth in 4D: robust tissue reconstruction and lineaging at cell resolution. Nature Methods.**7,** 547–553.

Gor, V., Elowitz, M., Bacarian, T., and Mjolsness, E. (2005). Tracking cell signals in fluorescent images. IEEE Workshop on Computer Vision Methods for Bioinformatics. 142–150.

Grandjean, O., Vernoux, T., Laufs, P., Belcram, K., Mizukami, Y., and Traas, J. (2003). *In vivo* analysis of cell division, cell growth, and differentiation at the shoot apical meristem in *Arabidopsis*. Plant Cell. **16,** 74–87.

Heisler, M.G., Ohno, C., Das, P., Sieber, P., Reddy, G.V., Long, J.A., and Meyerowitz, E.M. (2005). Auxin transport dynamics and gene expression patterns during primordium development in the *Arabidopsis* Inflorescence Meristem. Current Biol. **15,** 1899–1911.

Li, K., and Kanade, T. (2007). Cell population tracking and lineage construction using multiple-model dynamics filters and spatio-temporal optimization. The 2nd International Workshop on Microscopic Image Analysis with Applications in Biology.

Liu, M., Yadav, R.K., Roy-Chowdhury, A., and Reddy, G.V. (2010). Automated tracking of stem cell lineages of *Arabidopsis* shoot apex using local graph matching. Plant J. **62,** 135–147.

Marcuzzo, M., Quelhas, P., Campilho, A., Mendonca, A.M., and Campilho, A.C. (2008). Automatic cell segmentation from confocal microscopy images of the *Arabidopsis* root. IEEE International Symposium on Biomedical Imaging. 712–715.

Meyerowitz, E.M. (1997). Genetic control review of cell division patterns in developing plants. Cell. **88,** 299–308.

Najman, L., and Schmitt, M. (1994). Watershed of a continuous function. Signal Processing. **38,** 99–112.

Rangarajan, A., Chui, H., and Bookstein, F.L. (2005). The softassign procrustes matching algorithm. Information Processing in Medical Imaging. 29–42.

Reddy, G.V., and Meyerowitz, E.M. (2005). Stem-cell homeostasis and growth dynamics can be uncoupled in the *Arabidopsis* shoot apex. Science. **310,** 663–667.

Reddy, G.V., Heisler, M.G., Ehrhardt, D.W., and Meyerowitz, E.M. (2004). Real-time lineage analysis reveals oriented cell divisions associated with morphogenesis at the shoot apex of *Arabidopsis thaliana*. Development. **131,** 4225–4237.

Soille, P. (2003). Morphological Image Analysis: Principles and Applications, 2nd edn (Secaucus, NJ, USA: Springer-Verlag New York, Inc.).

Steeves, T.A., and Sussex, I.M. (1989). Patterns in Plant Development: Shoot Apical Meristem Mutants of *Arabidopsis thaliana*.

Viola, P., and Wells, M.W.III (1995). Alignment by maximization of mutual information. Fifth International Conference on Computer Vision. 16–23.

Williams, L., and Fletcher, J.C. (2005). Stem cell regulation in the *Arabidopsis* shoot apical meristem. Curr. Opin. Plant Biol. **8,** 582–586.