Segmentation of Overlapping Plants in Multi-plant Image Time Series

Jorge Gutierrez[a], S. Elizabeth Castillo[a], Malia A. Gehan[a], and Noah Fahlgren[a]

[a]Donald Danforth Plant Science Center, 975 N Warson Road, St. Louis, MO 63132, USA

ABSTRACT

Multi-plant imaging using arrays of low-cost cameras is a successful strategy for capturing affordable high-throughput plant phenotyping data. An imaging platform of this type can enable simultaneous imaging of hundreds to thousands of plants. The resulting datasets enable analysis of dynamic plant growth, development, and environmental responses at high temporal resolution. Full analysis of these datasets requires the identification of individual plants for measurement, but computational separation of individual plants becomes challenging when neighboring plants overlap. Here, we introduce the use of the watershed transform to segment moderately overlapping plants in multi-plant time series datasets. Rather than focusing on segmenting plants in individual images, we utilize information encoded in the entire time series to propagate plant labels from an early time point when individual plants are separate to later time points. In preliminary studies, using this method allowed us increase the analyzable size of the dataset by 28%.

Keywords: Plant segmentation, Watershed, Image time series

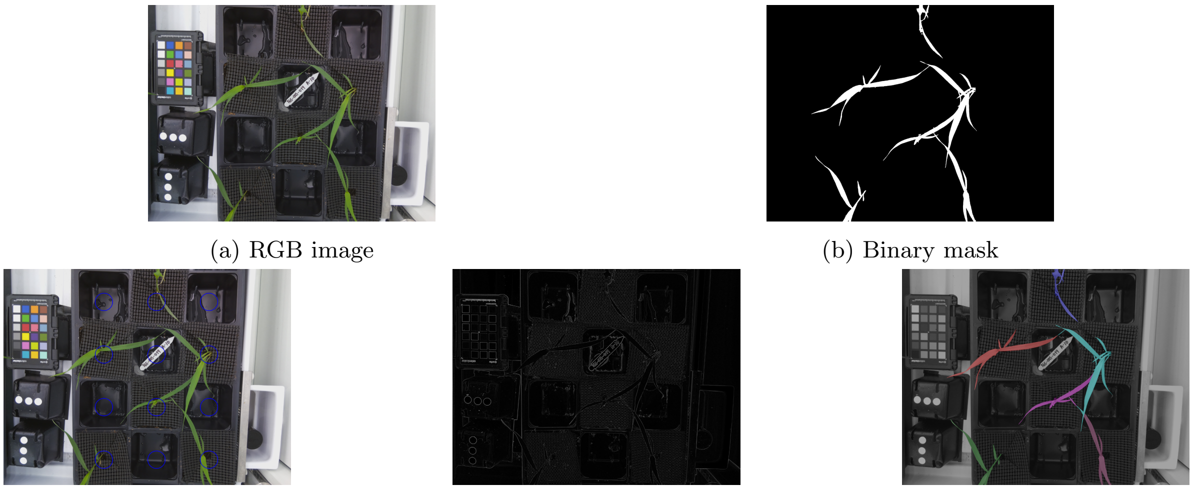
1. INTRODUCTION

Plant phenotyping aims to measure characteristics of plants or plant products for the purpose of research, breeding, production, or post-harvest assessment.[1] Traditional phenotyping approaches can suffer from being qualitative, labor intensive, or destructive, and image-based methods aim to improve the throughput and reduce the cost of plant phenotyping, as well as expand the range of potential traits that can be measured.[2] The ability to non-destructively measure phenotypes over time is a particularly useful application of image-based phenotyping due to the potential for improved insights into dynamic plant responses in research and applications for real-time monitoring and data-driven decision making in agriculture.[3]

Image-based plant phenotyping utilizes computer vision and machine learning methods to extract biological measurements from image data, and the community has developed a broad set of tools for a wide variety of applications.[4] A major component of deriving measurements from images involves algorithms that identify features of interest (e.g. plants, cells, leaves, etc.) and ignore irrelevant information (e.g. background), frequently involving segmentation of image data into two or more sets of features. Segmenting green plants from the background (of different color) can be as simple as using a threshold in one grayscale channel of a given color space. In well-controlled experimental environments, satisfactory segmentation using a threshold and some basic mathematical morphology operations such as erosion and dilation is often achievable.

In fully-automated systems where individual plants are moved to the camera sensors for imaging, segmentation is usually simplified since a single plant is captured in each image.[1] In contrast, imaging in high-density plant growth facilities and field plots often collects data on multiple plants per image.[1,5] In a multi-plant image setting, a problem arises when two or more plants overlap in an image. Segmentation of individual plants in a connected region is more difficult than segmentation from the background because the colors of the plants are usually similar and the shape of the plants is irregular. After a given time, it can be difficult even for a human to tell plants apart.

Further author information: (Send correspondence to Jorge Gutierrez)
Jorge Gutierrez: E-mail: JGutierrez@danforthcenter.org, ORCID: 0000-0002-8656-1567
S. Elizabeth Castillo: ecastillo@danforthcenter.org
Malia A. Gehan: E-mail: mgehan@danforthcenter.org
Noah Fahlgren: E-mail: nfahlgren@danforthcenter.org



(a) RGB image (b) Binary mask

(c) ROIs/Markers (d) Magnitude of the gradient (e) Segmentation result

Figure 1: Images produced at different steps of using the watershed transform to segment overlapping plants in a single image. (a) the original image, (b) mask that discerns the plants from the background, (c) manually entered regions of interest (ROI) to be used as markers (d) magnitude of the gradient computed over the grayscale version of the original image (d) illustration of the segmentation result assigning a different color to each plant.

In this article we propose a method of using the watershed transform to segment moderately overlapping plants in multi-plant image time series. It exploits the spatial regularity of the plants over time (small movement and growth) and uses the watershed transform to guide the propagation of the labels in the spatial dimensions and across frames in time. Section 2 summarizes the watershed segmentation method and how it can be used to segment overlapping plants in a single image. In Section 3 we explain how we can view the image time series as a 3D array of data and how we can sequentially apply the watershed segmentation. Section 4 shows the results of segmenting a time series of *Setaria viridis* (L.) P.Beauv. plants.

2. WATERSHED SEGMENTATION

The watershed transform[6] is a popular method for image segmentation. The basic idea of the watershed transform is to interpret a grayscale image as a topographical map where the height at each point is given by the pixel intensity. Segmentation is achieved by devising divide lines to separate the water from different basins in a theoretical flood of the terrain.

There have been several variations and improvements to the original algorithm resulting in highly efficient implementations. Nevertheless, an essential aspect driving the segmentation result is the grayscale representation of the input image. Originally, Beucher (1992) lists examples of applying the watershed transform to the gradient, the distance function, and the inverse distance function, depending on the nature of the elements in the image to be segmented.[7] In relevant examples, Gehan et al. (2017) and Acosta-Gamboa et al. (2016) use the distance transform of the grayscale image to segment the leaves of a plant with no other parametrization.[8,9]

In the setting of segmenting overlapping plants we are interested in obtaining dividing lines corresponding to the visible edge between the overlapping leaves. Given a successful segmentation of the plants versus the background, and the appropriate markers, applying the watershed transform to the magnitude of the gradient (only on plant pixels) could be enough to obtain a coherent divide line between overlapping leaves. Figure 1 shows the result of using the watershed transform on two overlapping plants given manually generated markers.

Watershed segmentation using markers has the advantage of resulting in exactly the number of expected objects, reducing the problem of oversegmentation (more objects than expected). Obtaining divide lines precisely at the edge of overlapping leaves might be difficult if markers are far from that edge or if the edge is surrounded by high intensity pixels that form a barrier to that catchment basin.

Manually generating markers for multiple plants in the multiple frames of an image time series would be time consuming. Additionally, reusing the same markers for all the frames might result in badly positioned markers for some, leading to unsuccessful segmentation.

3. IMAGE TIME SERIES SEGMENTATION

3.1 Time series as a 3D array

We assume that the plant movement is minor between two consecutive frames. Thus, pixels in a past frame belonging to a given plant, will most likely belong to the same plant in a consecutive frame. Since the plants grow as time passes, then background pixels near the edges of the plants are likely to become plant pixels in future frames.

By stacking consecutive frames, we can consider a whole image time series as a 3D block of data (images in grayscale). We exploit this regularity of the images through time to propagate the labels from some initial markers in a frame where individual plants have not yet starting overlapping toward the rest of the frames. This strategy can be seen as using the whole segmentation results of the past frame as markers for the segmentation of the next one. The result is that a small set of pixels needs to be labeled and a likelihood that where two plants start touching, the pixels in both sides of the edge are already labeled correctly.

3.2 Time series segmentation

The watershed transform can be applied to 3D data by considering the connectivity between neighboring pixels in the 3rd (time) dimension. Using the implementation included in the Python package scikit-image[10,11] we can directly apply the watershed segmentation to the whole image time series. In addition to a 3D block of grayscale images, we need an accompanying 3D block of binary masks separating the plants from the background, a set of markers, and to compute the magnitude of the 3D gradient from the grayscale stack of images. The masking data indicate the pixels to take into account for the segmentation (the ones that are not background) and the gradient gives the pixel intensity where we apply the watershed transform.

This solution does not differentiate between propagating labels one direction or the other (past or future) in the 3rd dimension. The propagation of labels in watershed depends only on the pixel intensities and their connectivity. Bidirectional propagation may result in label propagation from *future* frames to *past* ones. Thus, a mislabeled pixel in a frame where two plants overlap can propagate the wrong label to pixels in the neighboring plant in previous frames where the plants do not yet overlap.

In order to mitigate this behaviour we enforce some direction by sequentially applying the watershed segmentation to small blocks of stacked frames in time order. This does not prevent propagation in the wrong direction inside the sub-blocks, but it prevents the propagation from frames further in the future. We apply this strategy to blocks of 3-5 frames. The method is summarized in Algorithm 1 and will be available in the next major release of the Python package PlantCV[9] as the function `segment_image_series`. A tutorial can be found in https://github.com/danforthcenter/plantcv-tutorial-segment-image-series.git.[12]

Algorithm 1: Segment Image Series

Data: *images*, *masks*, *rois*, k
Result: *output_labels*
output_labels ← initialization with zeros ;
$output\_labels_0$ ← markers from *rois* ;
for *each frame n* **do**
  img_stack ← grayscale block from $n - \lfloor \frac{k}{2} \rfloor$ to $n + \lfloor \frac{k}{2} \rfloor$ of *images*;
  mask_stack ← block from $n - \lfloor \frac{k}{2} \rfloor$ to $n + \lfloor \frac{k}{2} \rfloor$ of *masks*;
  markers ← block from $n - \lfloor \frac{k}{2} \rfloor$ to $n + \lfloor \frac{k}{2} \rfloor$ of *output_labels*;
  edges ← gradient magnitude of *img_stack* ;
  stack_labels ← watershed transform using *edges*, *markers* and *mask_stack* ;
  $output\_labels_n$ ← center frame of *stack_labels*
end

4. RESULTS AND DISCUSSION

The dataset consists of multi-plant pictures of *S. viridis* with six evenly spaced plants per tray. The pictures were taken every hour from 7AM to 10PM (included). We consider the last picture of one day and the first picture on the next to be contiguous in time. A subset of the dataset is available with the tutorial.[12] Figure 2 shows the results of using `segment_image_series` 1 in the above dataset. The eight frames correspond to four consecutive days late in the experiment where there is moderate overlap between the plants. Once the overlap between the plants becomes too severe, the method becomes less successful. In this dataset separating the overlapping plants allowed us to analyze around 38 more frames (over two days) compared to stopping the analysis once plants overlap. This amounts to around 28% more data.

In addition to *S. viridis*, the image series segmentation method was successfully used for *Arabidopsis thaliana* (L.) Heynh. and *Oryza sativa* (L.) (data not shown). These results indicate that an advantage of using watershed segmentation with 3D time series datasets is the ability to utilize the method with multiple plant species without the need for training or extensive optimization. Additionally, the method presented here can be implemented with existing tools in PlantCV and the Python scientific community and does not require specialized computing. Deep learning-based methods would be an obvious alternative approach, but these methods would require the collection of labeled training data, potentially for each species.

In future work we plan to present segmentation results from multiple plant species and evaluate the quality of segmentation using ground truth datasets.

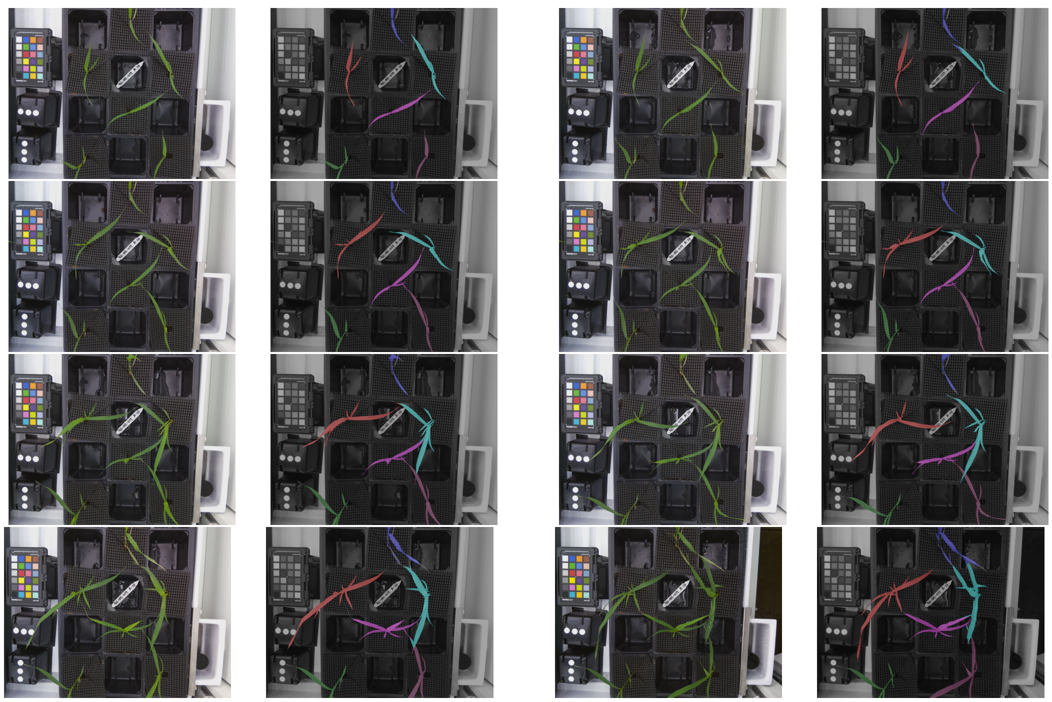


Figure 2: Segmentation results of the *S. viridis* image time series. Eight images corresponding to four consecutive days where the plants overlap moderately are presented as two columns. The original RGB images are shown on the left side of each column. The visualization images on the right side show the segmentation results assigning a different color (label) to each plant.

ACKNOWLEDGMENTS

This work was supported by grants from the National Science Foundation (1921724) and the United States Department of Agriculture National Institute Food and Agriculture (2019-67021-29926).

REFERENCES

[1] Casto, A. L., Schuhl, H., Tovar, J. C., Wang, Q., Bart, R. S., Fahlgren, N., and Gehan, M. A., "Picturing the future of food," *Plant phenome j.* **4** (Jan. 2021).

[2] Dhondt, S., Wuyts, N., and Inzé, D., "Cell to whole-plant phenotyping: the best is yet to come," *Trends Plant Sci.* **18**, 428–439 (Aug. 2013).

[3] Harfouche, A. L., Jacobson, D. A., Kainer, D., Romero, J. C., Harfouche, A. H., Scarascia Mugnozza, G., Moshelion, M., Tuskan, G. A., Keurentjes, J. J. B., and Altman, A., "Accelerating climate resilient plant breeding by applying Next-Generation artificial intelligence," *Trends Biotechnol.* **37**, 1217–1235 (Nov. 2019).

[4] Lobet, G., Draye, X., and Périlleux, C., "An online database for plant image analysis software tools," *Plant Methods* **9**, 38 (Oct. 2013).

[5] Tovar, J. C., Hoyer, J. S., Lin, A., Tielking, A., Callen, S. T., Elizabeth Castillo, S., Miller, M., Tessman, M., Fahlgren, N., Carrington, J. C., Nusinow, D. A., and Gehan, M. A., "Raspberry pi-powered imaging for plant phenotyping," *Appl. Plant Sci.* **6**, e1031 (Mar. 2018).

[6] Beucher, S., "Watersheds of functions and picture segmentation," *IEEE International Conference on Acoustics, Speech, and Signal Processing* **7**, 1928–1931 (1982).

[7] Beucher, S., "The watershed transformation applied to image segmentation," *Scanning Microscopy* **1992**(6), 28 (1992).

[8] Acosta-Gamboa, L. M., Liu, S., Langley, E., Campbell, Z., Castro-Guerrero, N., Mendoza-Cozatl, D., and Lorence, A., "Moderate to severe water limitation differentially affects the phenome and ionome of arabidopsis," *Functional Plant Biology* **44**(1), 94–106 (2016).

[9] Gehan, M. A., Fahlgren, N., Abbasi, A., Berry, J. C., Callen, S. T., Chavez, L., Doust, A. N., Feldman, M. J., Gilbert, K. B., Hodge, J. G., et al., "Plantcv v2: Image analysis software for high-throughput plant phenotyping," *PeerJ* **5**, e4088 (2017).

[10] Van der Walt, S., Schönberger, J. L., Nunez-Iglesias, J., Boulogne, F., Warner, J. D., Yager, N., Gouillart, E., and Yu, T., "scikit-image: image processing in python," *PeerJ* **2**, e453 (2014).

[11] Neubert, P. and Protzel, P., "Compact watershed and preemptive slic: On improving trade-offs of superpixel segmentation algorithms," *International Conference on Pattern Recognition* , 996–1001, IEEE (2014).

[12] Gutierrez, J., "PlantCV tutorial - segment image series v1.0," doi.org/10.5281/zenodo.5573116 (Oct. 2021).